Deep Player Behavior Modeling

Johannes Pfau

Supervised by Prof. Dr. Rainer Malaka

Second reviewer: Prof. Dr. Magy Seif El-Nasr

Digital Media Lab

Faculty 3: Mathematics / Computer Science

University of Bremen

Submitted 30st November 2020
Defended 17th May 2021

A dissertation submitted in partial fulfilment of the requirements for the degree of Doctor of Engineering (Dr.-Ing.)

Universität Bremen

WWW.UNI-BREMEN.DE

First edition, June 10, 2021

Universität Bremen

Declaration by Postgraduate Students

Authenticity of Dissertation

I hereby declare that I am the legitimate author of this Dissertation and that it is my original work.
No portion of this work has been submitted in support of an application for another degree or qualification of this or any other university or institution of higher education.
I hold the University of Bremen harmless against any third party claims with regard to copyright violation, breach of confidentiality, defamation and any other third party right infringement.

Faculty/Institute/Centre/School	Faculty 3: Mathematics / Computer Science
Degree	Doctor of Engineering (Dr.-Ing.)
Title	Deep Player Behavior Modeling
Candidate (Id.)	Johannes Pfau (3116523)

Signature of Student ______________________________

Date June 10, 2021

Für Molto.

“Wenn es funktioniert, ist es veraltet.”

Acknowledgements

I would like to thank my supervisors Prof. Dr. Rainer Malaka and Prof. Dr. Magy Seif El-Nasr and my advisor Dr.-Ing. Jan D. Smeddinck for inspiration, guidance and empowerment, as well as Prof. Dr. Gordon Pipa and Dr. Tarek R. Besold for assisting me throughout the writing of my Master's thesis that led to the formation of this dissertation.

Thanks to Prof. Georgios N. Yannakakis and Antonios Liapis from L-Università ta' Malta for hosting my research exchange, great collaboration and pushing my understandings of game AI.

Thanks to all of my co-authors and colleagues, not limited to Georg Volkmar, Ioannis Bikas, Dmitry Alexandrovsky, Sebastian Höffner, Mehrdad Bahrini, Thomas Münder & Anke Reinschlüssel, Robert Porzel, Susanne Putze, Michael Bonfert, Nima Zargham, Supara Grudpan, Thomas Fröhlich, Tanja Döring, Dirk & Nina Wenig, Irmgard Laumann, Svenja Voß and Insa Warms-Cangalovic, for the close collaboration, mutual inspiration, critique, exchange of ideas, company and the occasional, but most welcome distraction from everything that is ought to be important.

Thank you for making the doctoral period an unforgettable chapter of my life.

I am grateful to the University of Bremen, the Technologie-Zentrum Informatik und Informationstechnik (TZI) and the Digital Media Lab for giving me the opportunities for integrated as well as independent research within the research projects Adaptify and EASE, funded by the Bundesministerium für Bildung und Forschung (BMBF) and Deutsche Forschungsgesellschaft (DFG), respectively. Thanks to all project partners and the private server *Beyond Aion* that enabled situated field studies and of course thanks to all participants, testers and players taking part in all of the evaluations that eventually make up this thesis.

Extraordinary thanks to my friends and family for never-ending support, enablement, concern and backup. Thanks for giving me more than I ever needed. And thanks for everything I never asked for. Thank you Molto, Tine, Lina and Sophie. Thank you Freya, Misery, Ori and Nala. Thank you Ewald, Tristan, Marco, Philipp, Kai and Eva and thanks to the rest of the fools.

Abstract

Video games have become the entertainment industry's leading branch, with revenues that surpass even TV, cinema or music. This rapid development goes along with equally skyrocketing consumer demands and expectations, not limited to the continuous production of content, the validation against flaws and gameplay bugs and the preservation of real-time online functionalities – in ever-growing systems and applications. While efforts to overcome these issues primarily involve distinct expenses of intensive manual labor, automated and/or artificial intelligence-driven approaches as procedural content generation, dynamic difficulty adjustment or autonomous testing aim at lifting the burden from the developers' shoulders. For the simulation of artificial behavior, human-likeness or believability is considered to be one of the main quality criteria, yet most industrial as well as academic approaches focus on **generally** believable behavior for these purposes. This dissertation introduces the concept, architecture, implementation and evaluation of Deep Player Behavior Modeling, which assesses the atomic decision making of particular players and generates **individual** behavior representations to be implemented in artificial agents. After the examination through multiple field studies in different games and genres, these agents proved to be able to convincingly display individual strategies and preferences, represent in-game proficiency accurately and became indistinguishable from their original human player. Together with an extensive literature review and expert interviews that point out the case for usable AI in video games, this thesis contributes to the fields of game user research, game AI, machine learning and player modeling within both academia and industry and illustrates significant advances in the application fields of dynamic difficulty adjustment, player substitution, automated game testing and serious games.

Zusammenfassung

Videospiele haben sich zum führenden Zweig der Unterhaltungsindustrie entwickelt, deren Umsätze inzwischen selbst Fernseh-, Kino- oder Musikwirtschaft übertreffen. Diese rasante Entwicklung geht einher mit ebenso zunehmenden Ansprüchen und Erwartungen der Verbraucher in Hinsicht auf unter anderem das kontinuierliche Angebot von Inhalten, Fehlervermeidung und -behebung und Erhaltung von Echtzeit-Online-Funktionalitäten – in stetig wachsenden Systemen und Anwendungen. Während die Bemühungen, diese Probleme zu überwinden, in erster Linie mit deutlichem Aufwand intensiver manueller Arbeit verbunden sind, zielen automatisierte und/oder durch künstliche Intelligenz gesteuerte Ansätze wie prozedurale Generierung, dynamische Schwierigkeitsanpassung oder autonome Testläufe darauf ab, die Last von den Schultern der Entwickler zu mindern. Für die Simulation von künstlichem Verhalten gilt Menschenähnlichkeit oder Glaubwürdigkeit als eines der Hauptqualitätskriterien, dennoch konzentrieren sich die meisten industriellen wie auch akademischen Ansätze für diese Zwecke auf **allgemein** glaubwürdiges Verhalten. Diese Dissertation stellt das Konzept, die Architektur, die Implementierung und die Evaluierung von Deep Player Behavior Modeling vor, das die atomare Entscheidungsfindung einzelner Spieler abbildet und **individuelle** Verhaltensrepräsentationen generiert, die anschließend künstliche Agenten steuern können. Nach der Evaluation durch mehrere Feldstudien in verschiedenen Spielen und Genres haben diese Agenten bewiesen, dass sie in der Lage sind, individuelle Strategien und Präferenzen überzeugend darzustellen, das Fertigkeitsniveau im Spiel akkurat zu repräsentieren und letztendlich von ihrem ursprünglichen menschlichen Spieler nicht mehr zu unterscheiden sind. Zusammen mit einer ausführlichen Literaturrecherche und Experteninterviews, die die Möglichkeiten von benutzbarer KI in Videospielen hervorheben, leistet diese Arbeit Beiträge zu den Bereichen der Spielnutzerforschung, Spiel-KI, maschinellem Lernen und Spielermodellierung sowohl in der Wissenschaft als auch in der Industrie und demonstriert bedeutende Fortschritte in den Anwendungsbereichen der dynamischen Schwierigkeitsanpassung, Spielersubstitution, automatisierten Spieltests und Serious Games.

Contents

1 Introduction 1
1.1 Deep Player Behavior Modeling 2
1.2 Document Structure 4

2 Background & Literature Overview 9
2.1 AI in Games Research 9
2.2 Player Modeling 10
2.2.1 Player Modeling Criteria 10
2.2.2 Classification & Discussion 15
2.2.3 Approach Contextualization 18
2.3 Industrial Background Extension 19
2.4 Application fields 22
2.4.1 Dynamic Difficulty Adjustment 22
2.4.2 Player Substitution 24
2.4.3 Automated Game Testing 25
2.4.4 Cheating/Botting Detection 27
2.4.5 Serious Games 28

3 Studies & Developments 29
3.1 Research Agenda 29
3.2 Methods 30
3.3 Engineering Deep Player Behavior Modeling 31
3.3.1 Initial Benchmark 31
3.3.2 Initial DPBM Module 41
3.3.3 Advanced Benchmark 42

3.3.4 Summary 51
3.4 Dynamic Difficulty Adjustment 53
3.4.1 Comparison to Heuristic Opponent Behavior 53
3.4.2 Long-Term Commitment 56
3.5 Player Substitution 58
3.6 Automated Game Testing 61
3.6.1 Autonomous Solving 61
3.6.2 Automated Balancing 63
3.7 Serious Games 66

4 Contributions Towards Research Questions 69

5 Limitations & Future Work 77

6 Conclusion 81

7 Publications 83
7.1 Foundational Publications 83
7.2 Supportive Publications 86
7.3 Additional Related Publications 88

References 245

List of Figures

2.1 Two players facing *embodiments* in *Divinity: Original Sin II* (Larian Studios, 2017), exemplary for the popular type of imitation opponents in video games. Mimicking outer appearance, equipment choices, underlying character stats and learned action skills they seem to provide a balanced encounter, yet lacking the competence of approximating the players' decision making but relying on heuristics or random actions. 23

3.1 Screenshot of *Lineage II*, illustrating a player using the *Elemental Crash (Fire)* skill on an enemy [$F1$]. The corresponding state-action record is depicted in Table 3.3. 34

3.2 HMM skill transition graph of a single player [$F1$]. White percentages and the skill icon sizes display the relative usage of the respective skill. The width of the transition arrows is proportional to the transition probability from one skill to another. The blue arrow shows the most likely skill to begin attacking each enemy, while red transition arrows depict the main rotation (transition probabilities are labeled black). Skills used in less than 3% of encounters are included in the calculations of the model but excluded from the visualization due to visibility reasons. 35

3.3 Example network for an individual player [$F1$]. Real valued variables are mapped to the range from 0 to 1, previous skills are one-hot encoded in the input layer. Hidden layer and neuron count varied after optimizing for prediction accuracy. 37

3.4 Movement data of a single player [$F1$]. Blue lines visualize the user's trajectory (starting at green, ending in the red spot), blue dots indicate skill usage. A yellow line encloses the travelled area. The thick black line shows the approximation of the *global* movement behavior via a B-spline. 39
3.5 Prediction accuracies for random guessing (RND), a baseline only relying on action frequencies (BL), HMMs, DTs and MLPs; heuristic filtering included. . 40
3.6 Behavior flowchart of an agent driven by the initial DPBM module. Given a certain situation, the action selection module (MLP) approximates the most likely action of the original individual player, filters it for feasibility and executes it, effectively ending up in a succeeding situation. 41
3.7 MLP architecture mapping game state (information about player, opponent and skill history) to action (skill usage) probabilities [$F3$]. Sizes of input and output layers varied depending on the player's class and skill usage. 43
3.8 Testing prediction accuracy heatmap between hidden layers and nodes. . . . 44
3.9 Average training and testing prediction accuracies over several iterations of IxI^4xO MLP modeling aggregated player behavior data [$F3$], with respect to the amount of training epochs. The orange line indicates overfitting based on a peak in testing accuracy. 45
3.10 Change in training and testing prediction accuracy of IxI^4xO MLP after 1000 training epochs with increasing amount of data points. The orange window approximates the session size, where accuracy tends to converge from 2-3 sessions on. 46
3.11 Training and testing prediction accuracy between only *RD*, only *MR* and the original combined (C) input vector. 47
3.12 Average prediction accuracies of $(t * I)xI^4xO$ MLPs after 1,000 and 10,000 training epochs, where t is the amount of time steps included in retrospection. 48
3.13 Top-1 to Top-10 training and testing prediction accuracies of a IxI^4xO MLP after 1000 training epochs. 49
3.14 RFC testing prediction accuracy heatmap between the number of used DTs and the maximum depth within. 50
3.15 Comparison of average Top-1 testing accuracies and training times from the advanced benchmark methods 51
3.16 In-game screenshot of *Korona:Nemesis* used in multiple studies [$S2, F2$]. The player on the left utilizes **Water** to counter a **Fire** projectile. 53
3.17 Elements in *Korona:Nemesis* and their respective interactions. Arrows illustrate when an element hits another critically, yielding symmetrical, extended Rock-Paper-Scissors dynamics. 54

3.18 Results of the Intrinsic Motivation Inventory subscales for [$S2$]. 54

3.19 DPBM architecture utilized for each individual player in *Korona:Nemesis* [$S2$, $F2$], mapping game state (information about player and closest target) to action probabilities. 55

3.20 Four opponent types in *AION* ([$F3$]), where α, β and γ employed traditional parameter tuning DDA (*rubber-banding*). 56

3.21 Exemplary arrangement of a subset of skills available to the Sorcerer class in Aion. Additionally, context-dependent skills (when the player or a target opponent is in a particular condition) and a multitude of items can be activated. 56

3.22 Daily number of unique players entering *Eternal Challenge* compared to all other available instances during the study period [$F3$]. 57

3.23 Results of the Intrinsic Motivation Inventory subscales for the compared traditional DDA variables and DPBM [$F3$]. 58

3.24 Study sequence for each match [$F2$]. 58

3.25 Boxplot illustrating the results of the custom awareness scale between players that detected *(d)* a bot and players unaware *(u)* of substitution [$F2$]. 59

3.26 Scene from *Anna's Quest*. Circles represent objects (blue) or characters (red), which can be viewed, used or combined with one of the items (upper left corner). Additionally, items can be viewed, used or combined with other items. 61

3.27 Reward map visualization of a single game state in *ICARUS* ([$S1$]). 62

3.28 In-game screenshot of the Player versus Environment (PvE) benchmark in *Aion* (NCsoft, 2008). DPBM-driven player replicas encounter 100 heuristic opponents with increasing difficulty in one-on-one situations (attack horizontally, maxHP vertically). Depending on the game state between the agent and its target, emerging behavioral patterns for action preferences and sequences can be observed. 63

3.29 Proficiency heatmaps of the best, average and worst player replication of the benchmark. The horizontal axis denotes the increasing attack value of the heuristic opponent while the vertical axis describes the increasing HP value of it (+25% per step, respectively). 64

3.30 Proficiency results of player replicas across different classes from the PvE evaluation (left) and after parameter regulation (right). Includes means (indicated by **x**), medians (–) as well as the proficiency of a generalized model of the class (◊) and random play (O). 65

3.31 Screenshot excerpts from *Kitchen Clash* [*A*4]. Within a generative, natural language processing procedure, task descriptions for everyday activities are derived from written online corpora and have to be completed in a Virtual Reality (VR) game environment. Subsequently, compound executions are graded in terms of quantitative efficiency and qualitative assessments. While general object constraints marked requirements for completing the tasks, the actual action sequence and specific object usage could be utilized to solve underspecified roles and estimate human-like preferences distributions. . . . 66

3.32 Screenshot from the mobile game *Tool Feud* [*A*5]. Task descriptions for everyday activities are presented with a constantly changing set of tool choices, where players have to choose the most appropriate. DPBM's preference distribution of the whole population are utilized to model the correctness of the respective answers, yet the most popular choice(s) are dynamically removed to filter out obvious answers and acquire alternative solutions. 67

3.33 Screenshot from the online multi-player game *Elevator Empire*. While players seek to optimize and expand their own hotel enterprise, they have to train and manage their constantly increasing staff. The behavior of these employees is manifested by DPBM and manipulated by demonstration, leading to a successively accumulating optimization problem that adaptively challenges the player while simultaneously aggregating world knowledge about everyday activities. 68

4.1 Measured proficiency values (blue) versus predicted proficiency (green), using a $1572x1572^4x1$ DPBM network that mapped game state variables and current and preceding actions to a single proficiency target value (ranging from 0 = worst to 1 = optimal proficiency). 76

5.1 Behavior flowchart of an agent employing the succeeding DPBM module version. 78

List of Tables

2.1 Player modeling taxonomy based on four independent facets, according to Smith et al. (2011). 11
2.2 Literature review containing articles addressing “player modeling”, “opponent modeling”, “imitation learning” or “player behavior” in “video games”. 15
2.3 Classified research objectives of the related work depicted in Table 2.2. 17
2.4 Classification of DPBM within the player modeling criteria 18

3.2 Excerpt from a **player** entry recorded in a complete match session of *Dota2*, provided by OpenDota (https://docs.opendota.com/). Further attributes that do not contribute to the purpose of player modeling are omitted for visibility. 32
3.3 Sample state-action entry recorded in *Towards Deep Player Behavior Models in MMORPGs* [*F*1], as seen in Figure 3.1. In comparison to 3.2, this behavior information is recorded at every action (i.e. skill) a player executes and incorporates surrounding game state variables. **Bold** fields are used in the behavior model computation, others serve visualization or movement analysis purposes. 33
3.4 Percentages (and absolute numbers in parentheses) of bot detection estimates, according to the responses to the in-game bot detection survey [*F*2]. 59

List of Abbreviations

AI Artificial Intelligence . . . 1

ANN Artificial Neural Network . . . 12

BL Baseline . . . 39

CNN Convolutional Neural Network . . . 12

CTF Capture The Flag . . . 27

DDA Dynamic Difficulty Adjustment . . . 2

DPBM Deep Player Behavior Modeling . . . 2

DT Decision Tree . . . 36

EC Eternal Challenge . . . 42

FPS First-Person Shooter . . . 14

FSA Finite State Automaton . . . 19

GWAP game with a purpose . . . 66

HCI Human-Computer Interaction . . . 4

HMM Hidden Markov Model . . . 34

HP Health Points . . . 36

ICARUS Intelligent Completion of Adventure Riddles via Unsupervised Solving . . . 61

IMI Intrinsic Motivation Inventory . . . 30

LSTM Long Short-Term Memory network . . . 12

MCTS Monte-Carlo Tree Search . . . 12

MLP Multi-Layer Perceptron . . . 36

MMORPG Massively Multiplayer Online Role Play Game . . . 26

MOBA Multiplayer Online Battle Arena . . . 24

MP Mana Points . . . 38

MR	Main Rotation	47
NPC	Non-Player Character	19
PCG	Procedural Content Generation	12
PENS	Player Experience of Need Satisfaction	30
PvE	Player versus Environment	xvii
PvP	Player versus Player	65
RD	Responsive Decision	47
RFC	Random Forest Classification	12
RL	Reinforcement Learning	9
RNN	Recurrent Neural Network	12
RTS	Real-Time Strategy game	12
SDT	Self-Determination Theory	12
SOM	Self-Organizing Map	14
SVM	Support Vector Machine	12
TB	Test Bed	12
TD	Tower Defense	26

1

Introduction

Due to the steady growth of popularity and accessibility, the video game industry has evolved into a multi-billion dollar branch that surpassed all other entertainment lines such as TV, cinema or music [1]. Along with this development, player demands for content and mechanics are ramping up to extents that even large companies struggle to manage (Washburn Jr et al., 2016). Next to core content production, issues include software execution or gameplay bugs that go undetected (e.g., 80% of the 50 most popular games on the major distribution platform *Steam*[2] need critical updates after launch (Lin et al., 2017), players facing imbalanced challenges (Adams, 2002), connectivity issues with large-scale online systems (Kaiser et al., 2009) or the handling of cheating or other unethical behavior (Doherty et al., 2014). This dissertation investigates techniques for substituting or automating aspects of these challenges to anchor contributions in increasing player experiences, streamlining development and maintenance processes and cost-savings. Since the public release of video games, the field of scientific Artificial Intelligence (AI) examined them in order to establish agents capable of applied problem solving, outperform human proficiency and approach issues not limited to the previously mentioned cases, yet industrial development sticks to *"simple rule-based finite and fuzzy-state machines for nearly all their AI needs"* (Woodcock, 2001) in the majority of cases. Exceptions apply, predominantly in games where the AI itself constitutes the game's mechanics (Yannakakis, 2012), such as in the reinforcement learning of the companion animal in *Black and White* (Lionhead Studios, 2001), the dynamic difficulty adjustment features in *Halo* (Bungie, 2001) or *Left 4 Dead* (Valve, 2008) or the imitation learning (*Drivatar*) of *Forza Motorsport* (Turn 10 Studios, 2005). One of the major causes of this

[1] https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/

[2] https://store.steampowered.com/

disparity might be the significantly differing definitions between scientific and industrial AI, which in the former can be expressed as *"the theory and development of computer systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages"* [3], whereas within the context of video games, *"artificial intelligence consists of emulating the behavior of other players or the entities [...] they represent. The key concept is that the behavior is simulated. In other words, AI for games is more artificial and less intelligence. The system can be as simple as a rules-based system or as complex as a system designed to challenge a player as the commander of an opposing army"* (Kehoe, 2009). After the following section represents the scientific background as well as statements of the industry that elaborate on requirements of game AI (**plausibility/believability**, **computational performance** and **ease of implementation**), the main contribution of this dissertation is introduced by the design, development and evaluation of Deep Player Behavior Modeling (DPBM) that aims at overcoming the aforementioned issues.

1.1 | Deep Player Behavior Modeling

Approaching the closing of multiple unsolved gaps in the aforementioned areas of imbalanced challenges, online connectivity breakdowns and the inestimable error potential emerging from vast game state spaces, this dissertation introduces Deep Player Behavior Modeling (DPBM) that establishes implicit Dynamic Difficulty Adjustment (DDA), enables online player substitution and augments automated game testing while considering the previously identified design guidelines. In contrast to recent advancements in game playing through deep learning (e.g. Deep Q-Learning (Mnih et al., 2015), AlphaGo (Silver et al., 2016a) or AlphaZero (Silver et al., 2017), DPBM does not optimize for in-game performance or proficiency, but for the proximity to player-specific behavior. In the following sections, DPBM will be explicated as a generative and individual approach of computational player modeling that targets a close replication of particular players, enabling the applications of appropriately challenging opponents (cf. Section 3.4, $[F3, S2]$), temporary substituting disconnected players while keeping an approximate proficiency (cf. Section 3.5, $[F2]$) or incorporating particular player behavior into autonomous testing routines (cf. Section 3.6, $[S1]$). In order to model individual characteristics of play, every atomic executed action is recorded together with the contextual game state (cf. Table 3.3, $[F1]$). Once a sufficiently representative amount of data is provided, a model mapping game states to actions can be established via machine learning

[3] https://en.oxforddictionaries.com/definition/artificial_intelligence.

(cf. Section 3.3.1) that can drive computer controlled agents by generating behavior approximating the original player. By application of this technique, the following underlying research question is approached, further divided into sub-questions Q1-Q4 for granularity.

How can generative player modeling be realized in order to substitute individual human-like decision making in a representative, fair and convincing manner?

Q1. Can generative player modeling be utilized to reproduce individual player behavior with measurably similar decision making?

Q2. Can generative player modeling convince players that it imitates individual behavior believably?

Q3. Can challenging artificial agents that employ the player's individual decision making lead to a motivating experience?

Q4. Can generative player modeling contribute added value to unresolved issues within dynamic difficulty adjustment, online disruptions and playtesting in ecologically valid game scenarios?

I hypothesize that the implementation of DPBM introduced through this thesis and its corresponding publications is capable of representing and generating individual behavior that comes close to the original player, convinces this and fellow players of the similarity, makes up for motivating challenges and expedites progress within the enumerated application fields. Consequential to consistent results approving the hypothesis, this thesis represents a considerable advancement of research on the applicability of machine learning methods for applied problems in video game development. Unprecedented implementation and evaluation setups, short- to long-term field studies and transparent narrations amount to novel, unique and valid theoretical, technical and empirical contributions to the fields of machine learning, games user research and player modeling within both academia and industry.

1.2 | Document Structure

In the following, this thesis is structured into multiple sections and begins with reciting the foundational, supportive and additional related publications that constitute this dissertation. The background chapter summarizes the history of classic scientific game AI, comprehensively reviews and classifies the recent literature concerning player modeling, contextualizes this approach within these and elaborates on the application fields of DDA, player substitution, automated game testing and undesirable behavior detection accompanied by related work in these areas. Within Studies & Developments, the underlying research question as well as proportioned sub-questions and methods to answer these are displayed, following technical developments of DPBM (benchmarks and architecture) as well as results of evaluations in the aforementioned fields (from a Human-Computer Interaction (HCI) perspective). Section 4 joins the advances from these particular studies and answers both sub-questions as well as the overarching research question. Eventually, limitations of the used approach are revealed, discussed and criticized, before solutions and further investigations are proposed through future work. After a concluding statement, complete versions of the contained publications are presented together with their contribution towards this thesis and the personal contribution of the author.

	Foundational Publications
[*F*1]	Pfau et al. (2018a): Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. **Towards Deep Player Behavior Models in MMORPGs.** In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*. ACM, 2018.
[*F*2]	Pfau et al. (2020b): Johannes Pfau, Jan David Smeddinck, Ioannis Bikas, and Rainer Malaka. **Bot or not? User Perceptions of Player Substitution with Deep Player Behavior Models.** In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, 2020b.
[*F*3]	Pfau et al. (2020c): Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. **Enemy Within: Long-term Motivation Effects of Deep Player Behavior Models for Dynamic Difficulty Adjustment.** *In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, 2020c.
[*F*4]	Pfau et al. (2020a): Johannes Pfau, Antonios Liapis, Georg Volkmar, Georgios Yannakakis and Rainer Malaka. **Dungeons & Replicants: Automated Game Balancing via Deep Player Behavior Modeling.** *In Proceedings of the 2020 IEEE Conference on Games (CoG)*. IEEE, 2020a.

	Supportive Publications
[*S*1]	Pfau et al. (2017): Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. **Automated Game Testing with ICARUS: Intelligent Completion of Adventure Riddles via Unsupervised Solving**. In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play*. ACM, 2017.
[*S*2]	Pfau et al. (2019b): Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. **Deep Player Behavior Models: Evaluating a Novel Take on Dynamic Difficulty Adjustment.** In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, 2019b.

[*S*3] Pfau et al. (2020d):
Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. **The Case for Usable AI: What Industry Professionals Make of Academic AI in Video Games**. In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play*. ACM, 2020.

Additional Related Publications

[*A*1] Pfau et al. (2018b):
Johannes Pfau, Jan David Smeddinck, Georg Volkmar, Nina Wenig, and Rainer Malaka. **Do You Think This is a Game?** In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 2018b.

[*A*2] Pfau and Malaka (2019):
Johannes Pfau and Rainer Malaka. **Can You Rely on Human Computation?: A Large-scale Analysis of Disruptive Behavior in Games with a Purpose.** In *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*. ACM, 2019.

[*A*3] Volkmar et al. (2019):
Georg Volkmar, Johannes Pfau, Rudolf Teise, and Rainer Malaka. **Player Types and Achievements – Using Adaptive Game Design to Foster Intrinsic Motivation.** In *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*. ACM, 2019.

[*A*4] Pfau et al. (2019a):
Johannes Pfau, Robert Porzel, Mihai Pomarlan, Vanja Sophie Cangalovic, Supara Grudpan, Sebastian Höffner, John Bateman, and Rainer Malaka. **Give Meanings to Robots with Kitchen Clash: A VR Human Computation Serious Game for World Knowledge Accumulation.** In *Joint International Conference on Entertainment Computing and Serious Games*. Springer, 2019a.

[*A*5] Pfau and Malaka (2020):
Johannes Pfau and Rainer Malaka. **We Asked 100 People: How Would You Train Our Robot?** In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play*. ACM, 2020.

[A6] Bahrini et al. (2020b):
Mehrdad Bahrini, Nima Zargham, Johannes Pfau, Stella Lemke, Karsten Sohr and Rainer Malaka. **Enhancing Game-Based Learning Through Infographics in the Context of Smart Home Security.** In *Joint International Conference on Entertainment Computing and Serious Games*. Springer, 2020b.

[A7] Bahrini et al. (2020a):
Mehrdad Bahrini, Nima Zargham, Johannes Pfau, Stella Lemke, Karsten Sohr and Rainer Malaka. **Good Vs. Evil: Investigating the Effect of Game Premise in a Smart Home Security Educational Game.** In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play.* ACM, 2020a.

[A8] Zargham et al. (2020):
Nima Zargham, Johannes Pfau, Tobias Schnackenberg and Rainer Malaka. **Handle With Care: Exploring Recognition Error Handling Methodologies for Speech-Based Systems.** In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. Under Review.* ACM, 2021.

[A9] Porzel et al. (2020):
Robert Porzel, Vanja Cangalovic, Mihai Pomarlan, Sebastian Höffner, Johannes Pfau, John Bateman and Rainer Malaka. **Understanding Instructions All the Way: A Simulation-based Approach.** In *Proceedings of the 28th International Conference on Computational Linguistics. Under Review.* 2020.

2

Background & Literature Overview

2.1 | AI in Games Research

In contrast to its industrial counterpart, AI in games research most prominently formulates game playing as problem solving and aims to find optimal, efficient and/or dominant solutions within highly dimensional game state spaces. Beginning with purely symbolic games that could be described within a limited set of logical rules (such as the *Nim* game (Bouton, 1901), effectively solved by the first computer dedicated only to video games, *Nimatron* (Condon, 1942), the field quickly started to tackle more complex board games that are regarded as requiring high levels of human intelligence (Newell et al., 1972). Most of these were eventually beaten by AI, using mostly classical search approaches combined with heuristics, such as in the backgammon solver BKG 9.8 (Berliner, 1980), Chinook (Schaeffer, 1989) that reached a perfect algorithm in playing checkers (Schaeffer et al., 2007) or most popularly DeepBlue (Campbell et al., 2002) that received global attention in beating the world chess champion Kasparov in 1997. Recent advances include but are not limited to natural language processing (e.g. IBM's Watson that outperformed world champions of Jeopardy (High, 2012) or deep learning (e.g. AlphaGo (Silver et al., 2016b) or AlphaZero (Silver et al., 2017), excelling in chess, shōgi and go). Above that, they surpassed purely symbolic games and started to master subsymbolic, indeterministic and real-time video games such as ATARI games via deep Q-learning (Mnih et al., 2015), Super Mario World via neuroevolution (SethBling, 2015), Quake III via population-based Reinforcement Learning (RL) (Jaderberg et al., 2019) or StarCraft II via multi-agent RL (Vinyals et al., 2019).

Despite the great successes of these approaches, the main evaluation criterion remains winning performance, driving the overall objective to overcome human playing capabilities and competing algorithms. While this might yield novel and formidable challenges to top-tier professionals or world champions, near-optimal opponents rarely provide engaging matches for the vast majority of players. In this respect, player modeling addresses the individualization of game experience by adapting game content to the psychological type, proficiency or emotional state of the player or establishes agents that maximize individual similarity, human-likeness or believability instead. Within the following sections, a systematic literature review will be described that depicts the history and current state of player modeling. After highlighting and classifying by methodical differences, the integration of this dissertation's approach is presented and accomplished advancements are emphasized.

2.2 | Player Modeling

In order to achieve and constitute an informative representation about the state of the art of scientific player modeling, I conducted a systematic literature review. Within the databases of ACM Digital Library, IEEE Xplore, Springer Link, Elsevier Science Direct, Semantic Scholar, AAAI Digital Library, CiteSeerX and arXiv, 1.777 articles could be identified that address approaches referred to as "player modeling", "opponent modeling", "imitation learning" or assessing "player behavior" in "video games". After removing duplicate entries, exclusion based on titles, exclusion based on abstract content and final selection based on inclusion criteria (technical or theoretical contribution to the field, publication within the last 20 years, english language), 65 publications remained to be examined in detail (cf. Table 2.2). Within these, utilized games and genres were noted and player modeling approaches distinguished via the criteria outlined in the following section.

2.2.1 | Player Modeling Criteria

Depending on the research focus, application field and scientific background, player modeling can be defined and realized in various distinctive ways. Smith et al. (2011) differentiate between the dimensions **Domain**, **Purpose**, **Scope** and **Source** of player modeling (cf. Table 2.1) that break down the most crucial features of particular approaches. Above that, Yannakakis et al. (2013) constitute a taxonomy that is able to express the technical implementation and architectural details of computational player modeling between model-based (top-down/framework-driven), model-free (bottom-up/data-driven) and hybrids. Within this taxonomy, they also discern the type of data used for the player model construction into atomic player actions and preferences, objective input in terms of physiological data and game context data, referring to the global game state. To contrast the approach of DPBM from the existing related work, Table 2.2 discerns prior work within the literature based on the most essential criteria from the mentioned taxonomies:

- **Scope: Individual** (Smith et al., 2011)
  ✓: If the approach is able to represent behavior specific to a single individual player
- **Purpose: Generative** (Smith et al., 2011)
  ✓: If the approach is able to generate artificial agent behavior based on the model
- **Data structure** (Yannakakis et al., 2013)
  A: Atomic data (state-action tuples for atomic actions)
  H: High-level behavior (metadata such as kills/deaths/level/time needed for tasks)
  M: Movement data (e.g. trajectories/point sequences)
  P: Physiological data (e.g. ECG, GSR or EEG)
  V: Visual representations (e.g. pixels)
- **Player Experience**
  ✓: If the approach evaluates player experience of applied player modeling

Domain	**Purpose**
Game actions *details recorded inside of the game's rule system*	Generative *literally produces details in place of a human player*
Human reactions *details observable in the player as a result of play*	Descriptive *conveys a high-level description, usually visually or linguistically*
Scope	**Source**
Individual *applicable only to one player*	Induced *learned/fit/recorded by algorithmic means*
Class *applicable to a sub-population*	Interpreted *concluded via fuzzy/subjective reasoning from records*
Universal *applicable to all players*	Analytic *derived purely from the game's rules and related models*
Hypothetical *unlikely to be applicable to any players, but interesting nonetheless*	Synthetic *justified by reference to an internal belief or external theory*

Table 2.1: Player modeling taxonomy based on four independent facets, according to Smith et al. (2011).

Reference	**Title** *(description)*	Game	Scope: Individual	Purpose: Generative	Data structure	Player Experience
Partlan et al. (2019)	Player Imitation for Build Actions in a Real-Time Strategy Game *(Learning buildings and unit preferences in a Real-Time Strategy game (RTS) via Random Forest Classification (RFC) and Recurrent Neural Networks (RNNs))*	StarCraft	✓	✓	A	
Ahmad et al. (2019)	Modeling Individual and Team Behavior through Spatio-temporal Analysis *(Extracting state graphs from classified movement behavior)*	DOTA2	✓		M	
Melhart et al. (2019)	Your Gameplay Says It All: Modelling Motivation in Tom Clancy's The Division *(Mapping in-game preferences to Self-Determination Theory (SDT) questionnaire items via Support Vector Machines (SVMs)*	Tom Clancy's The Division	✓		H	
Goulart et al. (2019)	Learning How to Play Bomberman with Deep Reinforcement and Imitation Learning *(Long Short-Term Memory networks (LSTMs) & behavioral cloning)*	Bomberman		✓	A	
Hester et al. (2018)	Deep Q-Learning from Demonstrations *(Increasing Deep RL performance by human data)*	Hero, Pitfall, Road Runner		✓	V	
Mehrasa et al. (2018)	Deep Learning of Player Trajectory Representations for Team Activity Analysis *(Team classification and event recognition in sports via Convolutional Neural Networks (CNNs))*	Hockey, Basketball	✓		M	
de Lima et al. (2018)	Player behavior and personality modeling for interactive storytelling in games *(Mapping player choices to Big-Five personalities for adaptive narratives via Artificial Neural Networks (ANNs))*	Test Bed (TB) adventure game	✓		H	
Liao et al. (2017)	Deep Convolutional Player Modeling on Log and Level Data *(Experience prediction based on atomic game state logs)*	Infinite Mario, Gwario	✓		A	✓
Camilleri et al. (2017)	Towards general models of player affect *(Affect estimation by biofeedback)*	TB Shooter, Puzzle, Horror			P	✓
Chen and Yi (2017)	The Game Imitation: Deep Supervised Convolutional Networks for Quick Video Game AI *(Visual imitation via Deep CNNs)*	Super Smash Bros, Mario Tennis		✓	V	
Bindewald et al. (2017)	Clustering-Based Online Player Modeling *(Clustering player behavior, then individualize by weighting parameters)*	TB arcade game	✓	✓	A	
Holmgård et al. (2016)	Evolving models of player decision making: Personas versus clones *(Automated game testing by evolving believable player models)*	TB dungeon crawler	✓	✓	A	
He et al. (2016)	Opponent Modeling in Deep Reinforcement Learning *(Deep Q-Learning)*	Soccer, Quiz Bowl	✓		H	
Holmgård et al. (2015)	Monte-Carlo Tree Search for Persona Based Player Modeling *(Player modeling without players via Monte-Carlo Tree Search (MCTS))*	TB dungeon crawler		✓	A	
Burelli and Yannakakis (2015)	Adapting virtual camera behaviour through player modelling *(Camera point-of-view preference adaptation)*	TB puzzle game	✓	✓	H	✓
Liapis et al. (2015)	Procedural Personas as Critics for Dungeon Generation *(Playability assessment of Procedural Content Generation (PCG) dungeons via archetypical player models)*	TB dungeon crawler	✓	✓	A	
Holmgård et al. (2014b)	Personas versus Clones for Player Decision Modeling *(Q-Learning & neuroevolution for categorical player models)*	TB dungeon crawler	✓	✓	A	
Holmgård et al. (2014a)	Evolving personas for player decision modeling *(Q-Learning & neuroevolution for categorical player models)*	TB dungeon crawler	✓	✓	A	
Holmgård et al. (2014)	Generative Agents for Player Decision Modeling in Games *(Q-Learning & neuroevolution for categorical player models)*	TB dungeon crawler	✓	✓	A	

Reference	**Title** *(description)*	Game	Scope: Individual	Purpose: Generative	Data structure	Player Experience
Lee et al. (2014)	Learning a Super Mario controller from examples of human play *(Finding unique behavior traits via inverse RL)*	Super Mario Bros	✓	✓	A	✓
Oh et al. (2014)	Imitation Learning for Combat System in RTS Games with Application to StarCraft *(Replaying human StarCraft sessions)*	StarCraft	✓	✓	A	
Yannakakis et al. (2013)	Player Modeling *(Taxonomy)*					
Ortega et al. (2013)	Imitating human playing styles in super mario bros *(Neuroevolution for playing similar than human players)*	Infinite Mario Bros	✓	✓	A	✓
Tekofsky et al. (2013)	Psyops: Personality assessment through gaming behavior *(Mapping from game behavior to personality types)*	Battlefield 3	✓		H	
Togelius et al. (2013)	Active Player Modelling *(Selection methods for supervised player modeling)*					
Liapis et al. (2013)	Designer Modeling for Personalized Game Content Creation Tools *(Preference learning (goal detection) in CAD software)*					
Holmgård et al. (2013)	Decision Making Styles as Deviation from Rational Action: A Super Mario Case Study *(Player modeling as difference to rational agent (A*) decision making)*	Super Mario Bros	✓		A	
Martinez et al. (2013)	Learning deep physiological models of affect *(Affect estimation by biofeedback)*	TB arcade game	✓		P	✓
Tence et al. (2013)	Stable growing neural gas: A topology learning algorithm based on player tracking in video games *(Movement modeling for PCG of worlds using neural gas)*	Unreal Tournament 2004		✓	M	
Drachen et al. (2012)	Guns, swords and data: Clustering of player behavior in computer games in the wild *(K-means/simplex volume maximization clustering)*	TERA, BF: BC2			H	
Karpov et al. (2012)	Believable bot navigation via playback of human traces *(Merging player movement traces)*	Unreal Tournament 2004		✓	M	✓
Gemine et al. (2012)	Imitative Learning for Real-Time Strategy Games *(Supervised learning of heuristic bot behavior)*	StarCraft II	✓	✓	A	
Smith et al. (2011)	An inclusive view of player modeling *(Taxonomy)*					
van Lankveld et al. (2011)	Games as personality profiling tools *(Mapping from game behavior to personality types)*	Neverwinter Nights	✓		H,M	✓
Mahlmann et al. (2010)	Predicting Player Behavior in Tomb Raider: Underworld *(Predicting churn rate)*	Tomb Raider: Underworld	✓		H	
Tognetti et al. (2010)	Modeling enjoyment preference from physiological responses in a car racing game *(Mapping from biofeedback to preferences)*	The Open Racing Car Simulator			H	✓
Pedersen et al. (2010)	Modeling Player Experience for Content Creation *(Experience modeling via preference learning (ANNs))*	Infinite Mario Bros	✓		H	✓
Zhang et al. (2010)	Playing Tetris using Learning by Imitation *(Learning player strategies for Tetris via SVMs)*	Tetris		✓	A	

Reference	Title *(description)*	Game	Scope: Individual	Purpose: Generative	Data structure	Player Experience
Tencé et al. (2010)	The Challenge of Believability in Video Games: Definitions, Agents Models and Imitation Learning *(Believability criteria within behavior modeling)*					
Drachen et al. (2009)	Player Modeling using Self-Organization in Tomb Raider: Underworld *(Self-Organizing Maps (SOMs) for player type categorization)*	Tomb Raider: Underworld	✓		H,M	
Weber and Mateas (2009)	A Data Mining Approach to Strategy Prediction *(Classification of strategy types based on gameplay features)*	StarCraft	✓		H	
Canossa and Drachen (2009)	Play-Personas: Behaviours and Belief Systems in User-Centred Game Design *(Framework for personalized game parametrization)*	Tomb Raider Underworld	✓		H	
Yannakakis (2009)	Preference learning for affective modeling *(Affect estimation by biofeedback (Bayesian Learning, ANNs))*				P	
Yannakakis et al. (2009)	Preference Learning for Cognitive Modeling: A Case Study on Entertainment Preferences *(Player entertainment modeling)*	TB exergame	✓		P	✓
Lueangrueangroj and Kotrajaras (2009)	Real-Time Imitation Based Learning for Commercial Fighting Games *(Frequentist modeling/dynamic scripting of player actions)*	Street Fighter Alpha 3	✓	✓	A	✓
Bakkes et al. (2009)	Opponent modelling for case-based adaptive game AI *(DDA & Nearest-Neighbor-Clustering)*	SPRING		✓	H	
Missura and Gärtner (2009)	Player modeling for intelligent difficulty adjustment *(DDA & SVM Clustering)*	TB 2D Shooter		✓	H	
Yannakakis and Hallam (2008)	Entertainment modeling through physiology in physical play *(Affect estimation by biofeedback)*	TB exergame	✓		P	✓
Togelius et al. (2007)	Towards automatic personalised content creation for racing games *(Combination of player modeling and neuroevolution to generate content)*	TB racing game	✓	✓	A	
Sharma et al. (2007)	Player modeling evaluation for interactive fiction *(Player modeling for adapting narrative content)*	TB adventure game	✓	✓	H	✓
Thue et al. (2007b)	Learning Player Preferences to Inform Delayed Authoring *(Player modeling for adapting narrative content)*	TB adventure game	✓	✓	H	✓
Thue et al. (2007a)	Interactive storytelling: A player modelling approach *(Player modeling for adapting narrative content)*	TB adventure game	✓	✓	H	✓
Bauckhage et al. (2007)	Learning Human Behavior from Analyzing Activities in Virtual Environments *(Bayesian Motion Modeling, Believability Testing)*	Quake II	✓	✓	M	✓
Schadd et al. (2007)	Opponent Modeling in Real-Time Strategy Games. *(Classification of RTS strategies by fuzzy models)*	SPRING	✓		H	
Baker and Cowling (2007)	Bayesian Opponent Modeling in a Simple Poker Environment *(Classification into 4 playing styles by bayesian modeling)*	Poker	✓		A	
Thawonmas et al. (2006)	Clustering of Online Game Users Based on Their Trails Using Self-organizing Map *(SOMs for player movement clustering)*	TB game			M	
Gorman et al. (2006)	Believability Testing and Bayesian Imitation in Interactive Computer Games *(Player modeling of First-Person Shooter (FPS) behavior, believability framework)*	Quake II	✓	✓	A,M	✓

Reference	Title *(description)*	Game	Scope: Individual	Purpose: Generative	Data structure	Player Experience
Yannakakis and Maragoudakis (2005)	Player modeling impact on player's entertainment in computer games *(Maximizing interestingness by bayesian learning opponents)*	Pacman	✓	✓	H	
Thurau et al. (2007)	Bayesian Imitation Learning in Game Characters *(Modeling movement characteristics of players)*	Quake II		✓	M	
Thurau et al. (2005)	Is Bayesian imitation learning the route to believable gamebots? *(Bayesian imitation learning of FPS movement)*	Quake II	✓	✓	M	
Charles and Black (2004)	Dynamic Player Modelling: A Framework for Player-centred Digital Games. *(General framework)*					
Houle (2006)	Player modeling for adaptive games *(Adapting manually defined player traits to individual players)*		✓		H	
Thurau et al. (2004)	Imitation learning at all levels of game-AI *(Imitation learning of movement, strategy actions, reactive behavior)*	Quake II	✓	✓	A,M	
Bauckhage et al. (2003)	Learning Human-Like Opponent Behavior for Interactive Computer Games *(Waypoint learning with neural gas)*	Quake II	✓	✓	M	
Davidson et al. (2000)	Improved Opponent Modeling in Poker *(Action prediction by ANNs)*	Poker	✓		A	
Billings et al. (1998)	Opponent Modeling in Poker *(Predicting and adapting to player behavior by weighting parameters)*	Poker	✓		A	

Table 2.2: Literature review containing articles addressing "player modeling", "opponent modeling", "imitation learning" or "player behavior" in "video games".

2.2.2 | Classification & Discussion

Out of the included 65 publications, 66.2% approached the modeling of *individual* behavior and 52.3% presented a method for *generating* artificial behavior from their models. 35.4% utilized **a**tomic behavior, whereas 32.4% relied on **h**igh-level information, 16.9% characterize **m**ovement, 6.2% interpreted **p**hysiological data and 3.1% **v**isual input. 27.7% focused on or added an additional player experience evaluation that assesses the quality of artificially generated behavior from the particular model. Interpreting these features, the literature can be classified into general research objectives (cf. Table 2.3). In this regard, *theoretical foundation* considers the construction of explanations, frameworks (Charles and Black, 2004) or taxonomies (Smith et al., 2011). *Analysis of general player behavior* approaches point out methodologies of visualization, classification (Drachen et al., 2012) or imitation (Holmgård et al., 2015) of game-typical, non-individual player behavior. *Movement dynamics* process mostly trajectories or spatial sequences of individual players (Bauckhage et al., 2003) or teams (Mehrasa et al., 2018). *Estimation of player experience* does not explicitly model behavior, but underlying motivation or other psychological factors to estimate affect (Yannakakis, 2009), personality (Tekofsky et al., 2013) or entertainment (Yannakakis and Hallam, 2008). In *adaptation* techniques, individual player models are constructed (mainly through high-level or physiological data) in order to adjust game parameters such as difficulty (Missura and Gärtner, 2009), narrative (Thue et al., 2007a) or visual configurations (Burelli and Yannakakis, 2015). *Opponent modeling* usually utilizes atomic decision making actions, but not to generate artificial game agents that act like individual players, but to inform agents about the preferences (Billings et al., 1998), strategies (Schadd et al., 2007) and probable behavior of their opponents. Only *replication* approaches aim at completely imitating individual player behavior based on atomic data, to generate artificial agents resembling player-specific strategies, preferences and play styles for various motivations. Partlan et al. (2019), Oh et al. (2014) and Gemine et al. (2012) approximated player behavior in terms of building or unit selection within the RTS games StarCraft and StarCraft II, Lee et al. (2014) and Ortega et al. (2013) imitated playthroughs of the Jump'n'Run Super Mario Bros and Gorman et al. (2006) and Thurau et al. (2004) modeled distinctive properties within the FPS Quake II. In order to augment automated game testing, Togelius et al. (2007) trained player model agents evaluating procedurally generated racing tracks and Holmgård et al. (2014) and Liapis et al. (2015) blended human decision making styles with pre-categorized personas in order to assess differences in behavior of players traversing a testbed dungeon crawler. Of all the 15 publications in the *replication* category, only 4 evaluated their approach using a player experience assessment. All of these focused on human likeness or believability, as well as objective or subjective performance of the agent (Gorman et al., 2006; Lee et al., 2014; Lueangrueangroj and Kotrajaras, 2009; Ortega et al., 2013).

Research objective	References	Count
Theoretical foundation (e.g. framework, taxonomy)	Yannakakis et al. (2013), Togelius et al. (2013), Smith et al. (2011), Tencé et al. (2010), Charles and Black (2004)	5
Analysis of general (non-individual) player behavior	Goulart et al. (2019), Hester et al. (2018), Chen and Yi (2017), Holmgård et al. (2015), Drachen et al. (2012), Mahlmann et al. (2010), Zhang et al. (2010)	7
Movement dynamics description	Ahmad et al. (2019), Mehrasa et al. (2018), Tence et al. (2013), Karpov et al. (2012), Bauckhage et al. (2007), Thawonmas et al. (2006), Thurau et al. (2007), Thurau et al. (2005), Bauckhage et al. (2003)	9
Estimation of player experience/affect/personality	Melhart et al. (2019), de Lima et al. (2018), Liao et al. (2017), Camilleri et al. (2017), Tekofsky et al. (2013), Holmgård et al. (2013), Martinez et al. (2013), van Lankveld et al. (2011), Tognetti et al. (2010), Drachen et al. (2009), Yannakakis (2009), Yannakakis et al. (2009), Yannakakis and Hallam (2008), Houle (2006)	14
Adaptation based on player model	Burelli and Yannakakis (2015), Liapis et al. (2013), Pedersen et al. (2010), Canossa and Drachen (2009), Missura and Gärtner (2009), Sharma et al. (2007), Thue et al. (2007b), Thue et al. (2007a), Yannakakis and Maragoudakis (2005)	9
Opponent/Strategy modeling	He et al. (2016), Weber and Mateas (2009), Bakkes et al. (2009), Schadd et al. (2007), Baker and Cowling (2007), Davidson et al. (2000), Billings et al. (1998)	7
Replication of individual player behavior	Partlan et al. (2019), Bindewald et al. (2017), Holmgård et al. (2016), Liapis et al. (2015), Holmgård et al. (2014b), Holmgård et al. (2014a), Holmgård et al. (2014), Lee et al. (2014), Oh et al. (2014), Ortega et al. (2013), Gemine et al. (2012), Lueangrueangroj and Kotrajaras (2009), Togelius et al. (2007), Gorman et al. (2006), Thurau et al. (2004)	15

Table 2.3: Classified research objectives of the related work depicted in Table 2.2.

2.2.3 | Approach Contextualization

DPBM aims at replicating individual player behavior on an atomic level to populate and evaluate artificial agent behavior (cf. Table 2.4). According to the taxonomy of Smith et al. (2011), DPBM directly utilizes *game actions* **(domain)** to *generate* **(purpose)** *individually* **(scope)** modeled behavior by means of *induced* **(source)** training of machine learning techniques. As per Yannakakis et al. (2013), it can be described as a *model-free* (bottom-up) player modeling technique that maps *gameplay data* to actions mainly via *classification*.

Besides believability or human likeness, little to no research has been done on motivational or engagement aspects of playing with or against player model agents, especially not when facing own individual behavior. Within believability assessments, related work is limited to recorded video comparisons from observers excluded from the original game play and assessing *general* human likeness, i.e. how likely it is for an agent to be human, but not how likely it is to be a particular individual player. Thus, this dissertation is not limited to the implementation and benchmarking of the underlying player modeling approach [F1, F2, F3, S2], but emphasizes the player experience evaluation with respect to motivation [F3, S2], awareness and individual believability [F2] within the application fields of dynamic difficulty adjustment, player substitution and automated game testing (cf. section 2.4 Application fields). For the sake of ecological validity and real-world versatility, it includes and builds upon long-term [F3] field studies [F2, F3, S2] within published online games [F1, F2, F3, S2].

Reference	**Title** *(description)*	**Game**	**Scope: Individual**	**Purpose: Generative**	**Data structure**	**Player Experience**
[F4] Pfau et al. (2020a)	Replication Army: Enhancing Automated Game Balancing by Deep Player Behavior Modeling	AION	✓	✓	A	
[F3] Pfau et al. (2020c)	Enemy Within: Long-term Motivation Effects of Deep Player Behavior Models for Dynamic Difficulty Adjustment	AION	✓	✓	A	✓
[F2] Pfau et al. (2020b)	Bot or not? User Perceptions of Player Substitution with Deep Player Behavior Models	TB Fighting game	✓	✓	A	✓
[S2] Pfau et al. (2019b)	Deep Player Behavior Models: Evaluating a Novel Take on Dynamic Difficulty Adjustments	TB Fighting game	✓	✓	A	✓
[F1] Pfau et al. (2018a)	Towards Deep Player Behavior Models in MMORPGs	Lineage II	✓	✓	A	

Table 2.4: Classification of DPBM within the player modeling criteria

2.3 | Industrial Background Extension

In order to get a representative impression of the differences between industrial game AI (where already finite state automata, heuristic decision making or pathfinding algorithms are referred to) and research game AI (e.g. reasoning, planning, learning, (player) modeling, knowledge bases or multi-agent interaction), 105 of the currently most successful game companies were contacted for qualitative semi-structured interviews within *The Case for Usable AI* [*S*3]. After a period of six weeks and two additional reminders, ($n = 9$) responses could be collected that served for an outcome-oriented structuring content analysis (Mayring, 2010).

All of the surveyed participants agreed on the successful integration of pathfinding in more or less every modern video game since algorithms like A* (Hart et al., 1968) are cheap in computation, reliable and compelling, which make up the necessary conditions for consumer environments. Above that, compared to many of the other fields of AI, pathfinding is absolutely essential for video game opponents to prevent totally idiosyncratic behavior, which led to a very early establishment in the industry. Another often mentioned technique is the Finite State Automaton (FSA), for its robustness and observability, despite lacking any higher level capability of reasoning. Developers state that they use them for *"Movement state machines, etc." (P6)*, *"Character action sequences and combat" (P4)* or *"a lot of tasks not considered AI, like managing states of User Interface widgets" (P3)*, fulfilling predictable tasks far from more elaborate AI approaches. Dynamic difficulty adjustment is reportedly roughly applied with heuristics like *"[opponents] will start to miss more after managing to hit the player too rapidly" (P6)*, while the same holds also for reasoning systems, which are mostly reduced to frugal decision making about movement (*"e.g. to find out what a good position to shoot from will be, considering things like line-of-fire, distance to target, minimal distance from current position, closeness to allies, etc" (P7)*, *"Most of our AI is still reactive, but we have systems that 'sample' positions in the world for things like: get good attack position, cover spot, etc" (P6)*. Knowledge bases for Non-Player Characters (NPCs) are elementary but common, incorporating known versus unknown facts, e.g. in *"computer player's knowledge of the game state (where other units are on the map)" (P3)*. PCG has found it's place in the game industry, not least because of games that are completely centered around it (e.g. *Minecraft* (Mojang, 2011), *Spore* (Maxis, 2008) or *No Man's Sky* (Hello Games, 2016) but also in regular games that are not completely focused on PCG, mostly for *"Worldbuilding" (P2)* or *"[generating] in-game content, like making trees at design time" (P3)*. Multi-agent interaction is stated to be a discipline that can improve game quality in a thoroughly manner, which is why many companies try to come up with good solutions, e.g. *“NPCs can decide to perform a complex attack together” (P4)*, *“One AI charges a player, while the team members give covering fire” (P6)*, albeit drawing on FSA for these decisions. The reasons for the sparse and conservative use of scientific AI are straightforward and shared among the industry:

> *“So far, our AI systems are mostly reactive and driven by behavior trees that get signals from events that happen in the world. The reason for this is that we need to model explicit rules in their behaviors to make the AI readable and “fun” for the player. Also, we need to do this using limiting CPU bandwidth and in a way that these systems are debuggable” (P6).*

When asked about their personal position with respect to scientific AI, they agreed that it bears a considerable potential of interestingness (for both developers and players) and capabilities of making the environment more believable, yet it comes with a notable implementation and configuration effort that actually make the industry focus on heuristic workarounds. The underlying mindset prevalent in contemporary video game industry is best resumed by referring to their own words:

> *“What we call “AI” in games is vastly different than what's used in academia, or in business/engineering/apps/... Due to specific requirements like suspension of disbelief, games need a tighter control of possible outcomes and cannot afford the situation to be wildly misinterpreted. [...] Using decision trees, goal oriented action planning, and similar is found in some games, but we still largely rely on hand-tuned conditions controlled by hard-coded ifs, state machines etc. If you care more about “plausibility” than “intelligence”, experience shows that hand-tuned solutions go a long way further than emergent ones. Also, consider the fact that performance budget is severely limited especially if there's a large number of actors. E.g we once experimented with a very elaborate goal-oriented action planning algorithm heuristic for gunfight tactics (choosing cover, targets, ...) where things like e.g. flanking were emergent results of the simple base logic resting on data like cover positions, precision estimation, etc... The results were impressive, but way too expensive. And could still produce unexpected results in some cases. When you consider that most games in that genre do away with prescripted actions for each possible scene, saving an order of magnitude on performance - and guaranteeing no unexpected behavior, you realize that there's still a long way to go for “real AI” in games.” (P1)*

> *"In order to make AI a noticeable feature where towns are full of interacting NPCs or where enemies are executing complex strategy, a company has to dedicate probably a dozen or more programmers/designers for over a year to set it all up, which is very expensive. Also, the more complex the AI, the more bugs that are created which reduces the polish of the game. We would love to have awesome villager AI with life like daily routines, but it's just too cost prohibitive." (P4)*

> *"As game AI is focused on creating entertainment rather than primarily solve problems (which academic AI typically does), and usually has much stricter constraints on performance than academic AI, it is often faster to custom make solutions rather than use academic approaches. It also appears to be largely cheaper to produce a solution that fits the game and is "correct enough" than actually implement a method that produces a correct result. I think for most game AI developers, the interest in using academically developed AI goes as far as it can improve specifics in AI behaviour reliably and within budget (both development resources as well as CPU and memory)." (P5)*

> *"I think there are some opportunities to do more "advanced" AI in video games, but, it probably means that these games needs to be build and designed "around" these systems to make them really shine." (P6)*

Summarized, these statements inform the development of the techniques examined in this dissertation and scientific AI in general by providing design guidelines that expect **plausibility/believability**, **computational performance** and **ease of implementation** to be applicable and recognized by the industry.

2.4 | Application fields

Given a fully functional player modeling implementation that satisfies the aforementioned criteria (representing individual player behavior based on atomic game actions that can be generated at runtime), a multitude of opportunities within various application fields emerge. This section comprises the most crucial fields for this dissertation, namely DDA (offering adaptation beyond parameter tuning; training players by exposing them to own strengths and weaknesses), player substitution (bridging online match disruption due to dropouts; providing more individually representative agents), automated game testing (enhancing the estimation of balancing issues by incorporating realistic human player behavior, relieving human testers) and cheating/botting detection (revealing behavior that is more likely to stem from undesirable third-party bots rather than players; yielding objective evidence based on behavior in cases of identity theft). Advancements and evaluations utilizing DPBM in these fields ($[F1-4, S2]$) are explicated in the course of the following section 3: Studies & Developments.

2.4.1 | Dynamic Difficulty Adjustment

Providing and balancing an accurate level of difficulty is critical for keeping players constantly engaged (Adams, 2002). Disparities can ultimately lead to boredom/underload or frustration/overload, which make for two of the main causes why players stop playing games (Debeauvais, 2016). Since individual skill and its progression are hard to foresee throughout potentially large player bases and difficulty and it's progression can not be defined or programmed precisely, the field of DDA attempts to regulate emergent mismatches dynamically. To estimate imbalanced challenge-proficiency-discrepancies, various assessment techniques have been researched, such as success probability estimation (Hunicke, 2005; Spronck et al., 2004), psychological evaluation (Van Lankveld et al., 2008) or biofeedback (Hristova, 2017; Liu et al., 2009; Nogueira et al., 2016; Rani et al., 2005). Alternatively, various machine learning techniques have been deployed to classify player expertise, such as nearest neighbor clustering (Bakkes et al., 2009), Support Vector Machines (SVMs) (Missura and Gärtner, 2009), neuroevolution (Olesen et al., 2008), Reinforcement Learning (RL) (Andrade et al., 2005) or Monte-Carlo Tree Search (MCTS) (Demediuk et al., 2019). However, when it comes to adjusting this difficulty, most approaches focus on heuristic parameter tuning, even in the most recent advancements (Ang and Mitchell, 2017, 2019; Constant and Levieux, 2019; Fernandes and Levieux, 2019; Frommel et al., 2018).

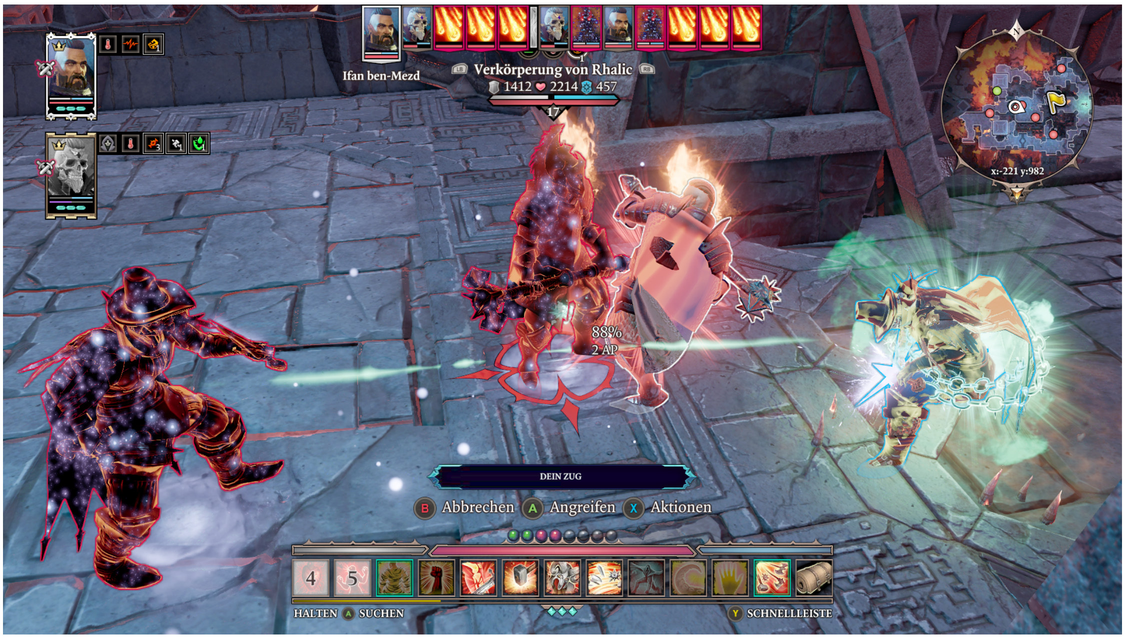


Figure 2.1: Two players facing *embodiments* in *Divinity: Original Sin II* (Larian Studios, 2017), exemplary for the popular type of imitation opponents in video games. Mimicking outer appearance, equipment choices, underlying character stats and learned action skills they seem to provide a balanced encounter, yet lacking the competence of approximating the players' decision making but relying on heuristics or random actions.

On the other hand, opponents that imitate the player character exist in numerous commercial games, perhaps most notably the recurring Dark Link in the The Legend of Zelda series (Nintendo EAD, 1987), Guild Wars' Doppelganger (ArenaNet, 2005), Renegade Shepard from Mass Effect 3 (BioWare, 2012), SA-X in Metroid Fusion (Nintendo R&D1, 2002) or Embodiments in Divinity: Original Sin II (Larian Studios, 2017), cf. Figure 2.1. These encounters are perceived as some of the most interesting challenges that artificial opponents can offer, since the strengths and weaknesses of the own player character have to be acknowledged and exploited in a seemingly balanced battle. Yet, so far these have only been realized as crude approximations of the original player, as they mimic appearance, equipment, basic moves and/or skill sets but rely on heuristic, strategically rigid decision making.

Combining this paradigm with generative player modeling, this dissertation introduces a distinct adaptation module that incorporates player proficiency implicitly instead of explicitly and represents and generates game proficiency on a multi-dimensional level, allowing for complex emergent dynamics. I hypothesize that an agent that keeps up with the progress of the player, displays similar strengths and weaknesses and challenges players to constantly improve or rethink strategies will yield a novel and captivating take on DDA.

2.4.2 | Player Substitution

Match disruption in online games is one of the major causes for frustration reported by players and makes for a frequent occurrence given varying network quality depending on location and over time (Cecin et al., 2004; Kaiser et al., 2009). Recent results of data mining *Dota2* ascertain that at least one player disconnects in 11.7% of over 50 million online matches (Michael, 2019). Designing and deploying scalable online games that avoid interruptions remains an important challenge (Guo et al., 2012).

Network stability and connection maintaining are under steady improvement, both in terms of progress on physical connections, as well as through the development of architectures and protocols for tackling discontinuity issues (Mildner et al., 2011; Plumb et al., 2018; Yahyavi and Kemme, 2013) or prediction of traffic anomalies to counteract bandwidth- or connectivity-loss before it becomes critical (Gu et al., 2011; Horovitz and Dolev, 2009). Yet, online games are still vulnerable to connectivity disruptions, since they can arise from a large variety of potential error sources, ranging from fast-paced real-time mechanics over massively large amounts of simultaneous players to vast connection distance differences that can span continents. In combination, these issues are improbable to be overcome completely and can significantly impact the motivation of affected players and of other players in the same play-session. Disconnected players in cooperative team fights for example have to be compensated for by allies which – depending on the game and genre – is unlikely to be manageable beyond short durations (Guthrie et al., 2014).

Apart from unintended cut-offs, disconnecting on purpose can also occur due to a range of reasons, such as *escaping*, in which players avoid their loss to be recorded, resentful behavior *("rage-quitting")*, in which players seek to deprive their opponent(s) of victory or intentionally hurt their own team in collaboratively competitive games, as well as forced disconnects of opponents via glitches or third-party tools (Moeller et al., 2009; Mørch, 2003; Yan and Randell, 2005, 2009; Yee et al., 2006). To counteract purposely caused interruptions, some games record them as losses or penalize them, which can lead to even higher frustration for non-self-inflicted disconnects (Robles et al., 2008). Other examples of successful commercial games substitute disconnected players by heuristic, computer-controlled bots that continue playing, e.g. Left 4 Dead (Valve, 2008), a FPS), Heroes of the Storm (Blizzard Entertainment, 2015), a Multiplayer Online Battle Arena (MOBA)), Super Smash Bros. 4 (Sora Ltd., 2014), a Beat'em up), Mario Kart 8 (Nintendo EAD, 2014), a racing game), Civilization V (Firaxis Games, 2010), a turn-based strategy game), Company of Heroes 2 (Relic Entertainment, 2013), an RTS), or Rocket League (Psyonix, 2015), a sports game). However, such substitution is frequently criticized, since the replacing bot is usually under-performing and not able to compete with human players. While modern machine learning approaches have proven to master a variety of games by continual improvement through simulated play (Mnih et al., 2015; Silver et al., 2016a; Tesauro, 1994), over-performing bots would also miss the point of adequate, representative substitutions, since they would yield an obvious and considerable potential for abuse.

In computer generated behavior in general, human likeness or believability has been established as one of the most important metrics to facilitate engaging game play (Acampora et al., 2012; Holmgård et al., 2014a; Khalifa et al., 2016; Miranda et al., 2016; Ortega et al., 2013; Turing, 1950; Umarov and Mozgovoy, 2014). However, these approaches have focused on producing a general closeness to human behavior so far, not explicitly on representing behavior from specific individual players within the same game session. Although player disconnects pose long-standing challenges, substituting disconnected players by means of player modeling bots has not been approached in openly published materials before, neither academically nor in the games industry, and – to the best of our knowledge – there is no prior scientific research on alternative temporary replacements. Thus, this dissertation will present, explicate and evaluate a novel method on bridging online match disruption by replacing dropout players with DPBM-fueled substitutes.

2.4.3 | Automated Game Testing

Automatic simulations of video game play have proven to be usable in situations where human testing is too tedious or not exhaustive enough for the purpose of finding bugs and glitches, parameter tuning, and assuring solvability.

The majority of scientific approaches focuses on detecting logical bugs or game crashes, such as Radomski and Neubacher (2015) or Varvaressos et al. (2017) who identified violations of manually defined constraints via simulated play. Buhl and Gareeboo (2012) highlight the utility of autonomous testing routines in everyday continuous integration and continuous delivery pipelines by contrasting the amount of encountered bugs against previous developments without them. Zheng et al. (2019) designed a game playing agent utilizing deep reinforcement learning, while Chan et al. (2004) made use of a neuroevolution approach that on top of playing was able to report on the constellation and sequence of actions that lead to game malfunctions. Furthermore, Bécares et al. (2017) mapped human tester playthrough records to semantic re-play models using Petri nets and Iftikhar et al. (2015) and Schaefer et al. (2013) introduced frameworks for autonomously testing generic games of the platformer or puzzle genre, respectively.

Other work tackles solvability (such as Powley et al. (2016) or Shaker et al. (2013) that aided level design of physics-based puzzle games by assuring potential solutions are feasible, or Schatten et al. (2017a,b) that simulated large-scale dynamic agent systems to test quest solvability in Massively Multiplayer Online Role Play Games (MMORPGs)), as well as performance and network load monitoring, as approached by Ostrowski and Aroudj (2013). Van Kreveld et al. (2015) and Southey et al. (2005) assessed difficulty or interestingness approximations of levels or mechanics by machine learning of descriptive in-game metrics.

Eventually, one of the most difficult and time-consuming phases of the game design process remains the balancing of different in-game units, character classes, factions or roles between which players are able to choose. Following the definition of Sirlin (2009), a multiplayer game is *"balanced if a reasonably large number of options available to the player are viable"* (where viability sets the requirement of having many meaningful choices throughout a game), while *"players of equal skill should have an equal chance at winning"*. Together with frequently desired asymmetrical configuration possibilities of these options, this inherently leads to combinatorial explosions, which can become hazardous for the enjoyability of the game and the satisfaction of its players. Even worse, Hullett et al. (2012) highlight that balancing issues most of the time *"only become apparent after many months of play"* and the trouble with these issues (in comparison to straightforward fixable bugs, glitches and solvability aspects) is that they do not only appear during the launch of a newly published game. Instead, balancing is an ongoing and repeating task that is heavily influenced by the perceptions of the player community (as per Lewis and Wardrip-Fruin (2010), *"after each patch, often the discussion begins again, factoring in new balancing or abilities for each class"*). In the games industry, this is most often approached through long-term expert analysis, excessive human play-testing, and persistent debates with the community.

In this regard, scientific approaches often build on simulations that iteratively assess balance criteria and dynamically tune in-game parameters based on the former. Jaffe et al. (2012), García-Sánchez et al. (2018) and de Mesentier Silva et al. (2017) applied this paradigm to board or card games, which was amplified by Mahlmann et al. (2012) by introducing procedurally generated cards on top of these simulations. In other genres, Beau and Bakkes (2016) utilized MCTS for balancing units of Tower Defense (TD) games, Morosan and Poli (2017) tweaked difficulty specifications in RTS and Arcade games after neuroevolution agents assessed these, Zook et al. (2019) deployed active learning to a 2D shooter within a number of iterations with varying parameters and

Leigh et al. (2008) dynamically balanced strategies by coevolution of two competing agents playing a Capture The Flag (CTF) game.

Closely related to the approach outlined in this dissertation, Holmgard et al. (2018) conflated atomic player behavior into procedural personas to simulate and test different play styles in a Dungeon Crawler game and Gudmundsson et al. (2018) utilized atomic choices in order to predict the difficulty of various levels of a Match-3-Puzzle game. Nonetheless, even if some approaches process some kind of human player input, incorporating actual information about individual and atomic player behavior has not been tackled yet. Generative player modeling has the potential to unite automatic simulation methods with behavioral information. This gives developers the opportunity to receive practically immediate insights on which player strategies are popular, dominant and/or may require rework, how parameter tuning will likely alter the outcome of strategies before presenting it to the community and how to automatically balance game mechanics after large-scale permutations of classes, setups, parameters and behavior – in all stages of development.

2.4.4 | Cheating/Botting Detection

One of the major classification paradigms in which player behavior has successfully been studied so far is the detection of unwanted automated software (botting) in online games, based on the players' traffic (Chen et al., 2008; Hilaire et al., 2010), social interactions (Oh et al., 2013) or action frequencies (Kim et al., 2005; Mishima et al., 2013; Thawonmas et al., 2008). Malicious bot software has no or little access to the actual game variables and objects and is thus usually based on heuristic or predefined decision making. Above that, botting is used mostly in worthwhile in-game areas and thus typically makes use of fixed paths, leading to rigid movement behavior. As such, differences between the classes of bot and human player can be identified quite accurately given the aforementioned techniques. A less investigated problem in online games is the act of identity theft, where criminals gain unwanted access to user accounts. Existing approaches tackle the issue through different means of automatic detection utilizing temporally structured metadata (Oh et al., 2012) or malicious action classification (Woo et al., 2012). These approaches can presumably be extended by employing in-depth player behavior models for the classification between real human account/character owners and imposters. Finally, competitive games are always prone to cheating or hacking. In such cases, DPBM can be utilized to improve play-style analytics in order to classify suspicious or technically impossible behavior.

2.4.5 | Serious Games

Apart from the aforementioned application fields that can immediately affect gameplay and player experience, player behavior and atomic decisions within serious games, human computation and/or games with a purpose can have additional importance for the respective scientific fields. One of the major challenges within educational serious games or exergames is the requirement for consistent continuation over longer terms, which is highly dependent on the game's potential to provide sustained intrinsic motivation (Wouters et al., 2013). Similar to the successful approaches of introducing AI methods (Johnson et al., 2005) or the integration of DDA into this genre (Hocine et al., 2014), the personalized challenges produced by DPBM might be able to even extend these motivational capabilities. When it comes to human computation games, large amounts of in-game player actions are recorded and interpreted to aggregate knowledge for solving real-world problems. Prominent examples as *Foldit* mine unfolding strategies of complex proteins (Curtis, 2015) or improve computer vision by incentivizing players to locate objects in images (*Peekaboom*) (Von Ahn et al., 2006). Since the accumulated results often affect real-world problems, disruptive players that produce malicious behavior should be detected and removed from the result set. Within this dissertation, an evaluation about the magnitude of this malicious behavior will be conducted, following with an implementation of a human computation serious game that utilizes DPBM for all of the aforementioned applications: facilitating long-term motivation by populating large-scale agent behavior with individual player strategies; substituting fellow players to ensure non-interrupted gameplay even when playing asynchronously; detecting malicious behavior by contrasting valuable versus malicious behavior; and directly interpret behavioral player traces to infer real-world knowledge for the purpose of robotic assistance within the context of everyday activities.

3

Studies & Developments

The following chapter elaborates on the underlying research questions and provides evidence raised in the foundational and supportive publications of this thesis. For each of those contributions, research rationales, results and discussions are outlined, which can be fully accessed within the respective publications (cf. $[F1-4]$, $[S1-3]$, $[A1-9]$). These publications correspond to the larger theoretical and empirical context of this dissertation and concern the underlying research agenda.

3.1 | Research Agenda

Due to the ongoing rise of complexity, popularity and content production cost of video game development, industrial production and maintenance, especially for flagship productions, is reaching the limits of what even large companies can sustain. Following the demand of players, games grow more complex in terms of content and mechanics, where the action spaces become nearly endless, greatly increasing the number of things that could potentially go wrong. This includes players facing unbalanced challenges, software execution or gameplay bugs that go undetected, connectivity issues with large-scale systems, and cheating or other unethical behavior. Usually, these issues are tackled by time-consuming parameter tuning, interminable testing routines prior and posterior to public launches, refinement of hardware architectures and software compensation protocols or persistent manual supervision, respectively. Even if these approaches are unlikely to be entirely replaced by alternative strategies, it remains even less likely that they will be able to abandon mentioned issues completely. Yet, applying a technique well-researched in other application fields might close the gap that traditional methods struggle to overcome: player modeling. Studying the capabilities of contemporary machine learning regarding modeling and generating individual player behavior on a representative level, the research agenda centralizes on the following general research question:

How can generative player modeling be realized in order to substitute individual human-like decision making in a representative, fair and convincing manner?

In order to keep interfering biases between the most crucial components of this general research question at a minimum, it was further on separated, evaluated and answered through the following sub-questions:

Q1. Can generative player modeling be utilized to reproduce individual player behavior with measurably similar decision making?

Q2. Can generative player modeling convince players that it imitates individual behavior believably?

Q3. Can challenging artificial agents that employ the player's individual decision making lead to a motivating experience?

Q4. Can generative player modeling contribute added value to unresolved issues within dynamic difficulty adjustment, online disruptions and playtesting in ecologically valid game scenarios?

3.2 | Methods

Since the interdisciplinary nature of the general research question is inherently divided into technical implementability (Q1) and player experience (Q2,3,4), the following section will first establish, describe and benchmark the underlying machine learning strategies that lead to the development of DPBM, before illustrating evidence for advancements within the mentioned application fields based on the results of the several conducted HCI/player experience evaluations. These mainly consist of self-determination theory (SDT) (Rigby and Ryan, 2011) questionnaires established in games user research, such as the Player Experience of Need Satisfaction (PENS) model (Ryan et al., 2006), the Intrinsic Motivation Inventory (IMI) (Ryan, 1982) or flow (Csikszentmihalyi et al., 1990), as well as self-constructed quantitative and qualitative assessments.

3.3 | Engineering Deep Player Behavior Modeling

The following chapter will document the most crucial steps carried out in order to construct a feasible implementation of DPBM. Before being able to make informed decisions about underlying machine learning techniques, let alone the integration into artificial agent decision making, a dataset had to be aggregated that would make for a fertile testing ground.

3.3.1 | Initial Benchmark

Publicly accessible datasets that comprise vast proportions of recorded real-world player information are found in several instances, such as OpenDota [1], an open-source platform offering extensive data about players, teams and hero characters within millions of match recordings of the popular MOBA Dota2 (Valve, 2013). Tracker Network [2] states to offer gameplay data for over 100 million players from prominent online games such as Fortnite (Epic Games, 2017), Counter Strike: Global Offensive (Valve, 2012) or Overwatch (Blizzard Entertainment, 2016) and PandaScore [3] provides real-time as well as recorded eSports data. Nevertheless, all of these third-party data providers rely on and offer only publicly available statistical data that fall into the category of high-level behavioral data (cf. Section 2.2.1), such as win/lose rates, kill/death/assist scores, final quantities of action uses, damage dealt, items bought, gold collected, experience points gathered and similar records (cf. Table 3.2). While this might yield significant insights about overall versatility of playable characters, balance and match analysis, the high-level meta data structure renders it insufficient for the desired atomic decision making modeling. Even with the information about which actions are used frequently, no knowledge about the contextual game state during these action decisions are contained, which limits the expressiveness of the eventual player modeling agent to action selection merely based on proportional frequencies. To give empirical evidence about this insufficiency, modeling based on mere action frequencies serves as a baseline for both the initial (cf. Section 3.3.1) as well as the advanced benchmark (cf. Section 3.3.4).

[1] https://www.opendota.com/
[2] https://tracker.gg/
[3] https://pandascore.co/

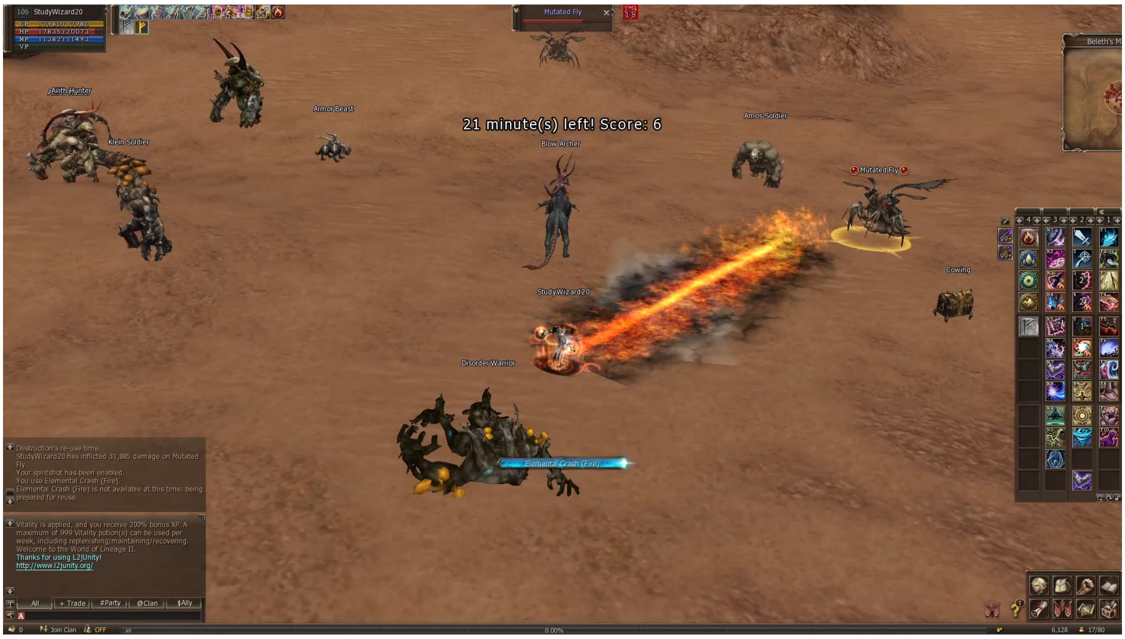
Attribute	Type	Description
ability_upgrades_arr	(int)	An array describing how abilities were upgraded
ability_uses	(object)	Object containing information on how many times the played used their abilities
ability_targets	(object)	Object containing information on who the player used their abilities on
damage_targets	(object)	Object containing information on how and how much damage the player dealt to other heroes
actions	(object)	Object containing information on how many and what type of actions the player issued to their hero
assists	(int)	Number of assists the player had
damage	(object)	Object containing information about damage dealt by the player to different units
damage_inflictor	(object)	Object containing information about about the sources of this player's damage to heroes
damage_inflictor_received	(object)	Object containing information about the sources of damage received by this player from heroes
damage_taken	(object)	Object containing information about from whom the player took damage
deaths	(int)	Number of deaths
gold_spent	(int)	How much gold the player spent
hero_damage	(int)	Hero Damage Dealt
hero_healing	(int)	Hero Healing Done
hero_hits	(object)	Object containing information on how many ticks of damages the hero inflicted with different spells and damage inflictors
item_uses	(object)	Object containing information about how many times a player used items
killed	(object)	Object containing information about what units the player killed
killed_by	(object)	Object containing information about who killed the player
kills	(int)	Number of kills
level	(int)	Level at the end of the game
multi_kills	(object)	Object with information on the number of the number of multikills the player had
permanent_buffs	(object)	Array describing permanent buffs the player had at the end of the game.
purchase	(object)	Object containing information on the items the player purchased
runes	(object)	Object with information about which runes the player picked up
stuns	(float)	Total stun duration of all stuns by the player
radiant_win	(boolean)	Boolean indicating whether Radiant won the match
duration	(int)	Duration of the game in seconds
isRadiant	(boolean)	Boolean for whether or not the player is on Radiant
win	(int)	Binary integer representing whether or not the player won
lose	(int)	Binary integer representing whether or not the player lost
total_gold	(int)	Total gold at the end of the game
total_xp	(int)	Total experience at the end of the game
neutral_kills	(int)	Total number of neutral creeps killed
tower_kills	(int)	Total number of tower kills the player had
courier_kills	(int)	Total number of courier kills the player had
lane_kills	(int)	Total number of lane creeps killed by the player
hero_kills	(int)	Total number of heroes killed by the player
observer_kills	(int)	Total number of observer wards killed by the player
sentry_kills	(int)	Total number of sentry wards killed by the player
roshan_kills	(int)	Total number of roshan kills (last hit on roshan) the player had
necronomicon_kills	(int)	Total number of Necronomicon creeps killed by the player
ancient_kills	(int)	Total number of Ancient creeps killed by the player
observer_uses	(int)	Number of observer wards used
sentry_uses	(int)	Number of sentry wards used
purchase_tpscroll	(object)	Total number of TP scrolls purchased by the player
actions_per_min	(int)	Actions per minute
rank_tier	(int)	The rank tier of the player. Tens place indicates rank, ones place indicates stars.

Table 3.2: Excerpt from a **player** entry recorded in a complete match session of *Dota2*, provided by OpenDota (https://docs.opendota.com/). Further attributes that do not contribute to the purpose of player modeling are omitted for visibility.

Thus, in order to assess the feasibility of the desired contextual state-action architecture, an initial study was conducted [*F*1] that asked players to perform a common task in the MMORPG *Lineage II* (NCSoft, 2003) and tracked all meaningful decisions made by the players within the situational context represented by a set of game state variables (cf. Table 3.3). Within a 30-minute time limit, participants with considerable prior experience of the game were asked to defeat as many NPC enemies as possible in a controlled single-player environment, using their preferred strategy of skill (action) selection and order. Afterwards, they were asked to explain their decision making in own words and highlight which actions responded to particular situations. Utilizing this data set, the following machine learning benchmark between hidden markov models, decision trees and multi-layer perceptrons was carried out.

Attribute	**Sample Value**	**Description**
time	10.02.2018 11:34:23	The time at which the skill was executed.
timeReuseAvailable	10.02.2018 11:34:33	The time from which on the skill will be available again.
skillID	11017	The unique ID to identify the skill.
skillName	Elemental Crash (Fire)	The skill's name.
casterID	268492421	The unique ID of the player carrying out the skill.
casterName	StudyWizard20	The player's name.
casterClassID	182	The unique ID of the player's character class.
casterClassName	Wizard (Feoh Storm Screamer)	The name of the character class.
casterHPpercentage	88.85	The current health points of the player before skill execution.
locX	-12856	The absolute x value of the player's location.
locY	386275	The absolute y value of the player's location.
locZ	-2958	The absolute z value of the player's location.
targetID	23324	The unique ID of the type of the target.
targetName	Mutated Fly	The target's name.
targetClassID	-1	The unique ID of the target's character class.
targetClassName	NPC	The name of the target's character class.
targetHPpercentage	68.1	The current health points of the target before skill execution.
ai_intention	AI_INTENTION_ATTACK	The current state of the target, if NPC (ATTACK, CAST, IDLE).
distance	370.75	The relative distance between player and target.
locXtarget	-12523	The absolute x value of the target's location.
locYtarget	386112	The absolute y value of the target's location.
locZtarget	-2944	The absolute z value of the target's location.
zone	Hellbound (Study)	The current area the player is located.
score	6	The current amount of enemies killed within the study.

Table 3.3: Sample state-action entry recorded in *Towards Deep Player Behavior Models in MMORPGs* [*F*1], as seen in Figure 3.1. In comparison to 3.2, this behavior information is recorded at every action (i.e. skill) a player executes and incorporates surrounding game state variables. **Bold** fields are used in the behavior model computation, others serve visualization or movement analysis purposes.



106 StudyWizard20
CP 17980/17980
HP 17835/20073
MP 11382/11493
VP

21 minute(s) left! Score: 6

Destruction's re-use time.
StudyWizard20 has inflicted 31,885 damage on Mutated Fly.
Your spiritshot has been enabled.
You use Elemental Crash (Fire).
Elemental Crash (Fire) is not available at this time: being prepared for reuse.

Vitality is applied, and you receive 200% bonus XP. A maximum of 999 Vitality potion(s) can be used per week, including replenishing/maintaining/recovering.
Welcome to the World of Lineage II.
Thanks for using L2jUnity!
http://www.l2junity.org/

Figure 3.1: Screenshot of *Lineage II*, illustrating a player using the *Elemental Crash (Fire)* skill on an enemy [F1]. The corresponding state-action record is depicted in Table 3.3.

Hidden Markov Models. As reported in the post-test questionnaire of *Towards Deep Player Behavior Models in MMORPGs* [F1], one major behavioral criterion seems to be the adherence to individual skill rotations (i.e. repeating sequences). In Hidden Markov Models (HMMs) (Baum and Petrie, 1966), sequences among states can be expressed via interconnected state-transition probabilities. This transition neglects most of the underlying (hidden) variables, but only depends on predecessor states. Albeit reducing the predictive power to a single dimension (action sequence), I formulated the estimation of the main rotation as a Markov chain with the respective previous skills as observable variables, while the complex behavior strategy stayed hidden. Consequential, they proved to be a suitable starting point for player modeling for their capability of generating intuitive illustrations. As seen in Figure 3.2, they can directly expressed as skill transition graphs, while the probability of a player executing one skill is approximated by the probability given a previously known predecessor. The most used skill together with its most probable successors constitute the individual main rotation, which differs from player to player.

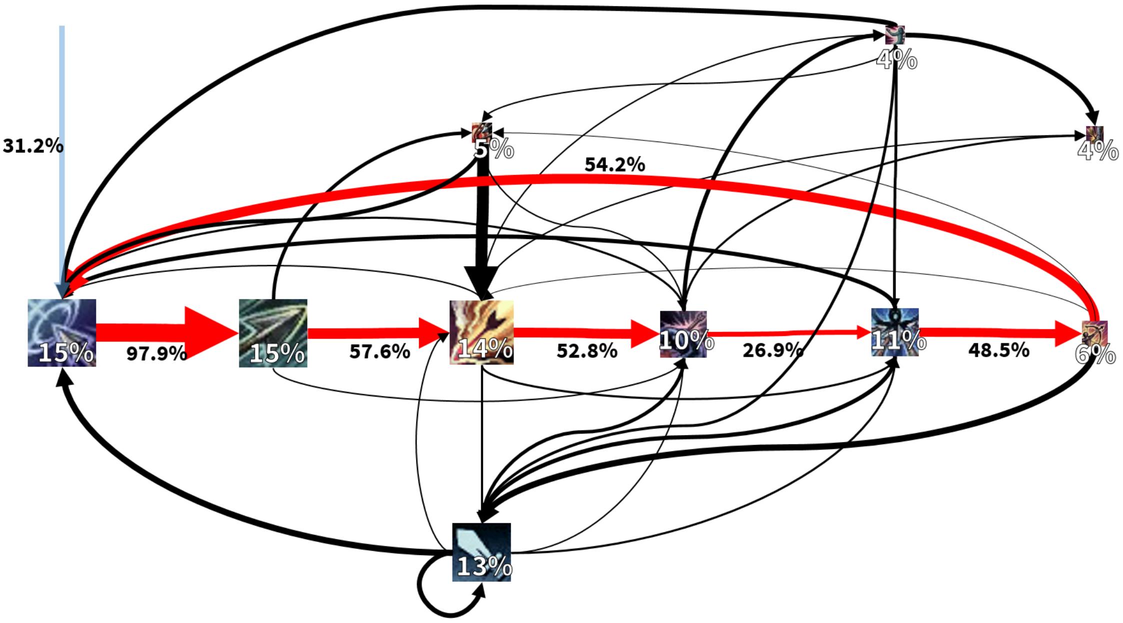


Figure 3.2: HMM skill transition graph of a single player [F1]. White percentages and the skill icon sizes display the relative usage of the respective skill. The width of the transition arrows is proportional to the transition probability from one skill to another. The blue arrow shows the most likely skill to begin attacking each enemy, while red transition arrows depict the main rotation (transition probabilities are labeled black). Skills used in less than 3% of encounters are included in the calculations of the model but excluded from the visualization due to visibility reasons.

Decision Trees. While HMMs struggle with incorporating larger numbers of dimensions, Decision Trees (DTs) (Breiman et al., 1983) can break data down following the most discriminatory variables by spanning trees of binary decisions. This allows for pinning down decisive factors for skill usage accurately from a selection of many contextual game state factors that are potentially relevant. Information was included regarding the enemy's intention (IDLE, ATTACK or CAST), the previously used skill and binary choices whether the player's Health Points (HP) percentage, the enemy's HP percentage or the distance between them is above or below the respective mean of the current player's data. Discriminativeness was calculated via Shannon entropy. Resultant, this approach did not only yield a higher accuracy in predicting skill usage compared to HMMs, but was even capable of "explaining" the intention of situationally used skills. By reversing the tree and collecting all paths ending in a particular skill leaf, the situational context of this skill can be assessed and compared to the qualitative statements of the post-study questionnaire. For example, when asked for skills that were situationally used, one participant stated that he activated a skill (*"Death Lord"*) whenever his HP dropped to a low level, in order to transfer some of the enemy's HP to his own. Reversing the tree returns **"NOT HP above mean"** (95.2% accuracy) as the top criterion for this skill, followed by **"Target HP above mean"** (90.5%). The player's contextual usage of this skill is thus accurately described by *"having low HP while the enemy has high HP"*. Furthermore, one player stated to use *"Bow Strike"* whenever an enemy gets too close, and thus knocking the enemy back. At this point, the tree returned **"NOT distance above mean"** and **"target HP above mean"** as top criteria, explaining even more than the uttered statement (since an approaching low HP target could be defeated quickly, but only approaching high HP targets are countered with the knock-back skill). This process of reversing DTs produces a ranked set of meaningful variables in which particular skills are used and is capable of delivering clearly understandable insights to developers.

Multi-Layer Perceptrons. Multi-Layer Perceptrons (MLPs) (Rosenblatt, 1958) add further predictive performance since the learning process does not rely on manually defined discrimination criteria and all variables contribute their real values instead of binary decisions, as is the case with DTs. As a drawback, neither can the trained model be easily visualized nor can the process be reversed in order to describe situations in which particular skills are used.

Above that, the computational and temporal effort of training and retrieving is considerably higher than for the former techniques. Nevertheless, the scalability of outputs beyond situations explicitly provided in the training data and high accuracy in predic-

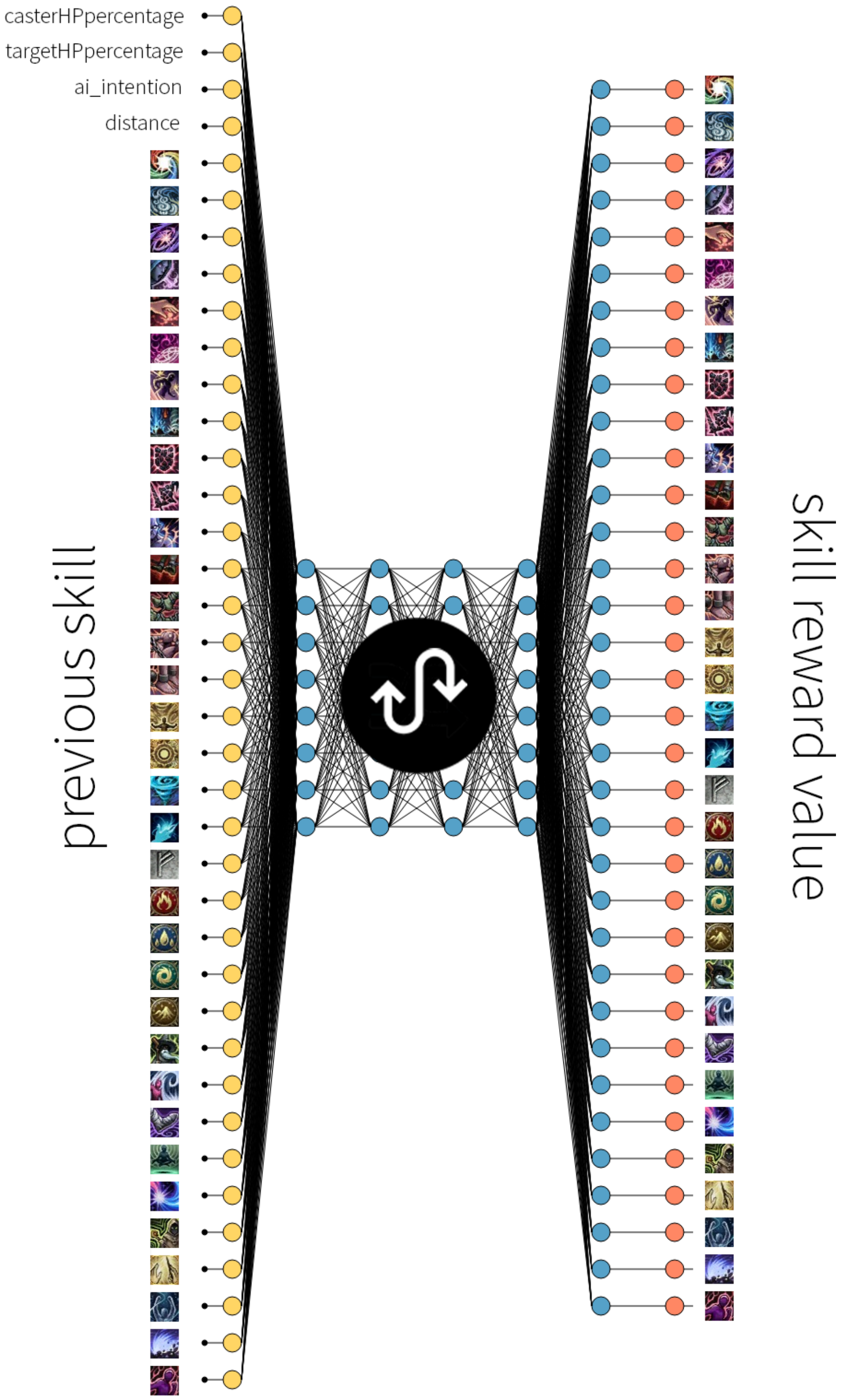


Figure 3.3: Example network for an individual player [F1]. Real valued variables are mapped to the range from 0 to 1, previous skills are one-hot encoded in the input layer. Hidden layer and neuron count varied after optimizing for prediction accuracy.

tion render it a viable candidate for behavior generation. To establish general applicability, one MLP with backpropagation and a logistic sigmoid activation function was trained for each participant individually, where input and output array sizes varied due to different numbers of skills used / available. Each network consisted of up to 38 input and up to 34 output nodes representing particular skill IDs after one-hot encoding (cf. Figure 3.3), where feasible values for the amount of hidden layers, nodes within and training epochs were examined afterwards. Target values were constituted by the (binary) use of particular skills given the situation defined from the input array. For the eventual prediction, the computed output array was translated to a density function from which the guessed skill is picked probabilistically.

Heuristics. In order to increase the eventual prediction accuracy, certain manually defined heuristics can be applied that filter, weigh or rearrange the resulting skill probability distribution produced by all of the mentioned techniques. Yet, to not superpose correct predictions with false assumptions, only conservative transformations of the distribution were applied, i.e. filtering out probabilities for actions that are technically impossible to execute. In this case, these only appeared in the form of cooldown time (skills that were not available to be used again), insufficient HP/Mana Points (MP) conditions or unfulfilled distance requirements.

Movement. When modeling movement, it might be beneficial to distinguish between *local* and *global* movement, where *local* decisions consider the momentary motion relative to a current opponent (e.g. approaching the enemy, keeping a certain distance or fleeing) and *global* movement describes higher-level goals (e.g. traversing the map in a particular pattern). While *local* movement was inherently implemented by integrating the relative distance between player and target as an input parameter within the modeling techniques, *global* movement decisions showed to be less assessable. Figure 3.4 illustrates the recorded behavior of a single participant of *Towards Deep Player Behavior Models in MMORPGs* [*F*1] and the computed model of *global* motion utilizing B-splines (Piegl, 1993). While those showed a crude approximation of the overall movement of a session, they suffered from a tremendous temporal computation effort and the necessity to manually define model parameters (dimensions) that rendered it insufficient for fully automatic approaches.

Summary. Within the pilot evaluation of DPBM [*F*1], the individual prediction accuracies of HMMs, DTs, and MLPs were compared against each other as well as random guessing and a Baseline (BL) condition that only took into account action frequencies (cf. Figure 3.5). Significant differences ($p < 0.05$) between the baseline and all machine learning techniques demonstrate that recorded data in the form of absolute frequencies

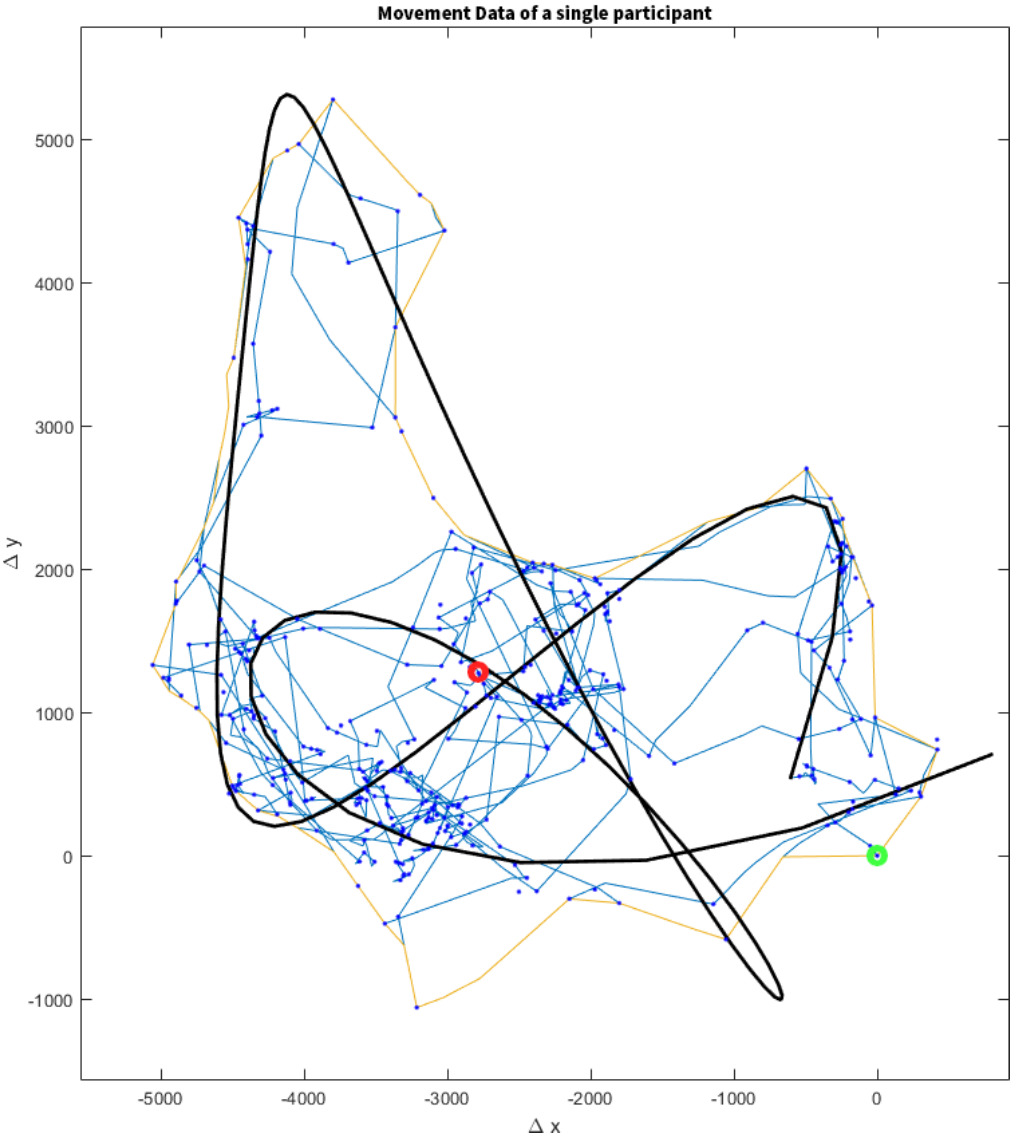


Figure 3.4: Movement data of a single player [*F*1]. Blue lines visualize the user's trajectory (starting at green, ending in the red spot), blue dots indicate skill usage. A yellow line encloses the travelled area. The thick black line shows the approximation of the *global* movement behavior via a B-spline.

(as in 3.2 from OpenDota) are infeasible for individual generative player modeling and comprehensive state-action data (as in 3.3) is considerably more expressive. Random guessing performed foreseeably abysmal given the high-dimensional action space. Furthermore, the study outlines different strengths of the deployed machine learning techniques [$F1$], with HMMs yielding immediately visualizable extractions of main rotations, DTs offering explanations for situationally used skills and MLPs presenting high prediction accuracy of skill usage. As the subsequent part of this thesis focuses on generation and replication instead of analysis, MLPs led to the construction of the first DPBM architecture (cf. Section 3.3.2).

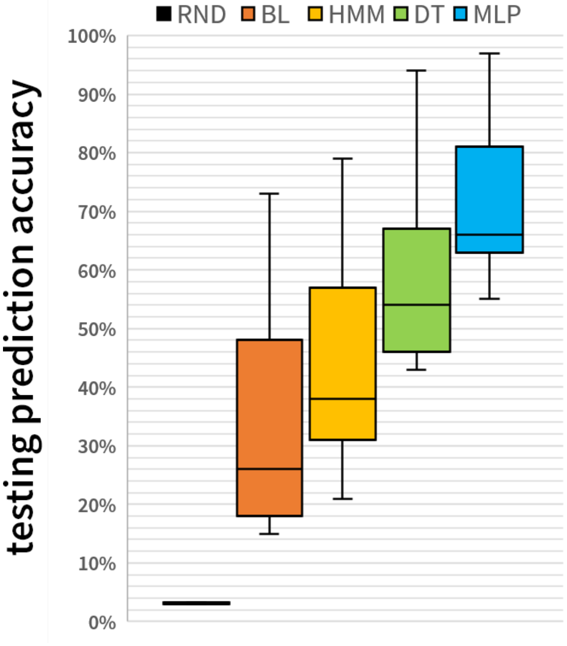


Figure 3.5: Prediction accuracies for random guessing (RND), a baseline only relying on action frequencies (BL), HMMs, DTs and MLPs; heuristic filtering included.

3.3.2 | Initial DPBM Module

Given the insights of this pilot evaluation $[F1]$, the first approach for an architecture describing a DPBM-fueled agent was formed. In this minimal setup (cf. Figure 3.6), state-action decision modeling makes up for the core module of replicating player behavior, which is only constrained by a heuristic filter handling infeasible conditions (e.g. insufficient HP/MP, distance or cooldown for particular actions). Utilizing the high prediction accuracy of deep learning, a number of successive studies $[S2, F2-4]$ deploy MLPs to constitute DPBM opponents within various evaluations. In these cases, modeling movement behavior was limited to approximating *local* motion by striving for the situationally favored distance to the target character. *Global* movement was neglected for the time being due to particularly confined spatial in-game areas within the studies and the lack of an efficient approach.

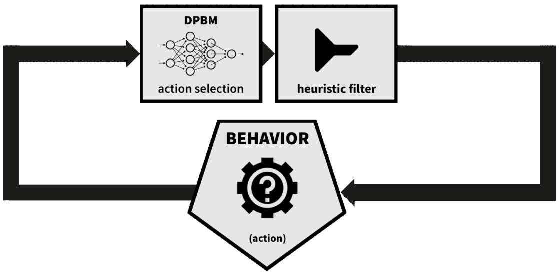


Figure 3.6: Behavior flowchart of an agent driven by the initial DPBM module. Given a certain situation, the action selection module (MLP) approximates the most likely action of the original individual player, filters it for feasibility and executes it, effectively ending up in a succeeding situation.

Notably, the observed behavior utilized for the initial benchmark originated from recording an in-game activity aiming at the elimination of a multitude of enemies significantly weaker than the player, differing drastically from the activities of later studies $[S2, F2, F3]$, where players challenge opponents at approximately eye level. This benchmark turned out to be helpful to estimate the capabilities of the used machine learning models, but to not disregard the presumably differing play styles in these situations, the following advanced benchmark encompasses data aggregated after completion of *Enemy Within* $[F3]$.

3.3.3 | Advanced Benchmark

In *Enemy Within* [F3], players with considerable prior experience of the MMORPG *Aion* (NCsoft, 2008) were introduced to the daily single-player dungeon instance *Eternal Challenge (EC)* developed for the purpose of this dissertation. EC included (but was not limited to) encountering a DPBM-driven opponent trained on the behavior from their individual preceding battles and thus aggregated one-on-one combat behavior in a considerably challenging setting. After a study period of four weeks, behavioral data of 171 players could be accumulated that should resemble decision making for the desired application fields fundamentally closer. The following section reports on differences of prediction accuracy of further conceivable machine learning approaches including the established MLPs, and illustrates the effects of overfitting, sample size, parameter contribution and time series inclusion within. Prediction accuracy values exclusively describe the raw probability distribution calculated by the respective machine learning technique, without additional heuristic filtering.

Multi-Layer Perceptrons. The following section concerns the evaluation of one MLP with backpropagation and logistic sigmoid activation functions per player. The benchmarks were conducted using Keras 2.2.4 with TensorFlow 2.0.0 backend on a NVIDIA GeForce RTX2080.

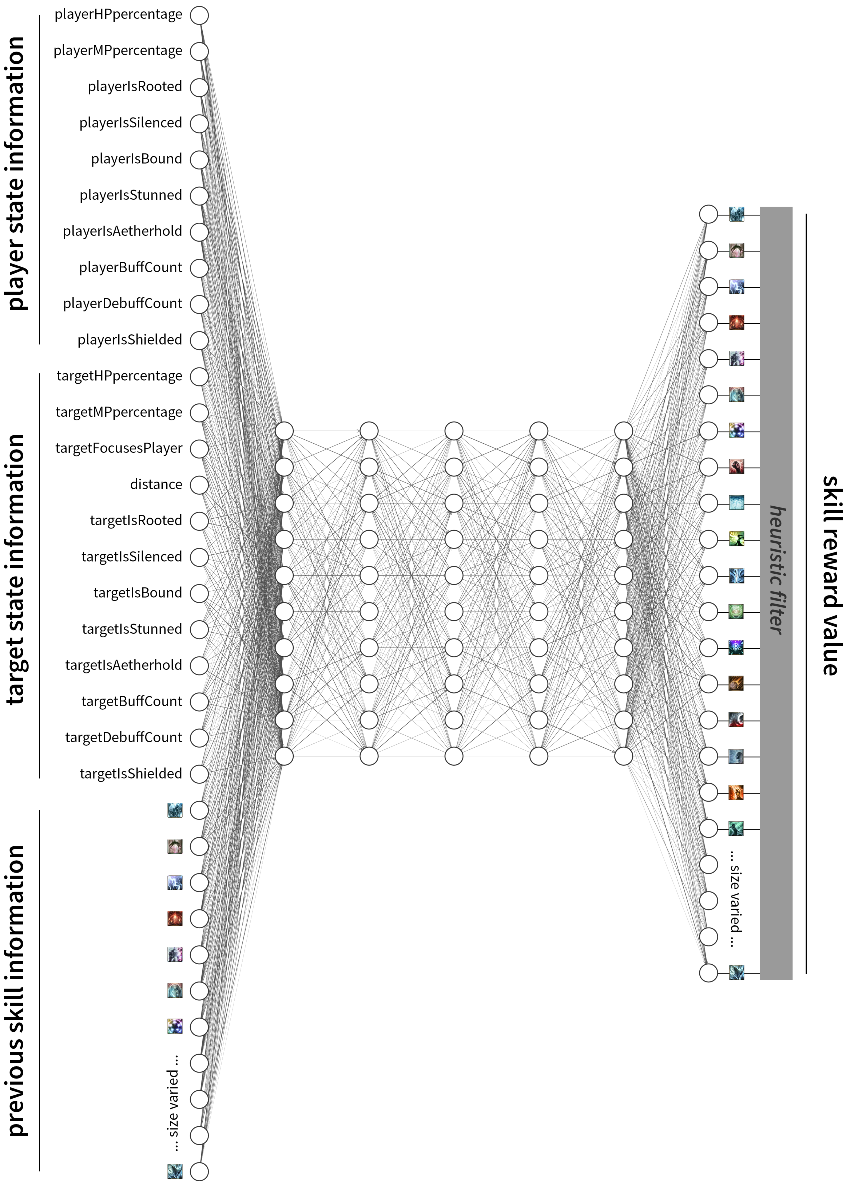


Figure 3.7: MLP architecture mapping game state (information about player, opponent and skill history) to action (skill usage) probabilities [*F*3]. Sizes of input and output layers varied depending on the player's class and skill usage.

Network architecture

In order to empirically estimate a suitable configuration of the network's hyperparameters, a two-dimensional benchmark was conducted that discerned the parameters of *hidden layers* and *amount of hidden nodes* within each of the former. Sizes of the input (I) and output (O) layer were determined in advance after one-hot encoding the individual player's skill set and incorporating further situation variables (cf. Figure 3.7), where I ranged from 86 to 122 $(M = 98.2, SD = 15.1)$ and O from 64 to 100 $(M = 76.2, SD = 15.1)$. Figure 3.8 visualizes the distribution of average testing prediction accuracy among varying hidden layers and nodes within. While the most suitable amount of hidden layers appears to lie in between 3 and 6, the quantity of nodes within each hidden layer shows a greater range of viable choices. Effectively, the latter is likely to be dependent on the input size which varies among individuals in this particular learning setup. In conclusion, the architecture for the following benchmarks was fixed to IxI^4xO (i.e. the same number of hidden nodes as input nodes for each of the 4 hidden layers). Notably, determining hyperparameters of deep learning techniques is a multi-dimensional problem where other features such as the number of training epochs might influence the benchmark outcome. Thus, this initial investigation was repeated with varying parameters of the following sections, but similar outcomes confirm the feasibility of the IxI^4xO architecture for this approach.

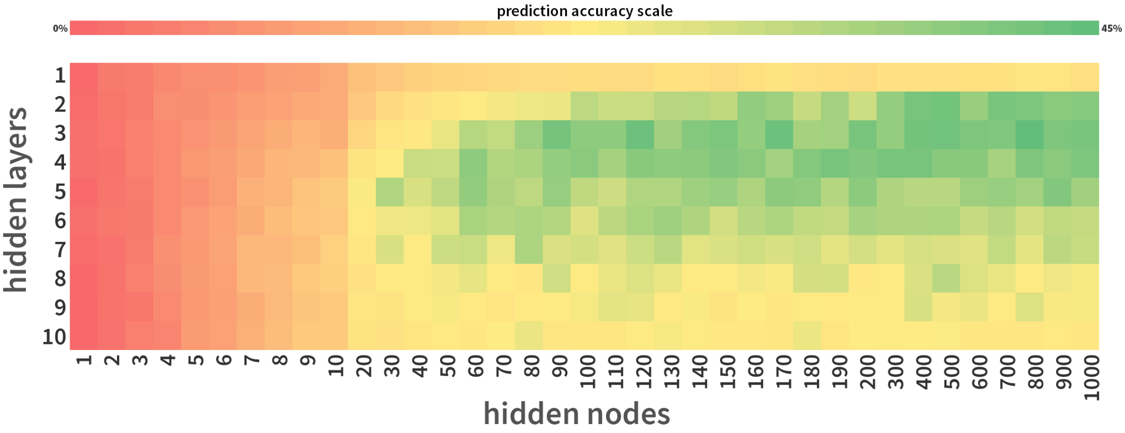


Figure 3.8: Testing prediction accuracy heatmap between hidden layers and nodes.

Estimating Overfitting

With the just optimized hidden node configuration, the influence of further parameters could be investigated, probably most importantly the number of backpropagation epochs used in training. Since neural networks in general keep increasing the fitness to the training data, accuracy on testing and other non-seen data might yield diminishing returns or even decreases from a certain point of time on. After a number of training series over the same dataset, overfitting could be identified beyond 1000 epochs (cf. Figure 3.9).

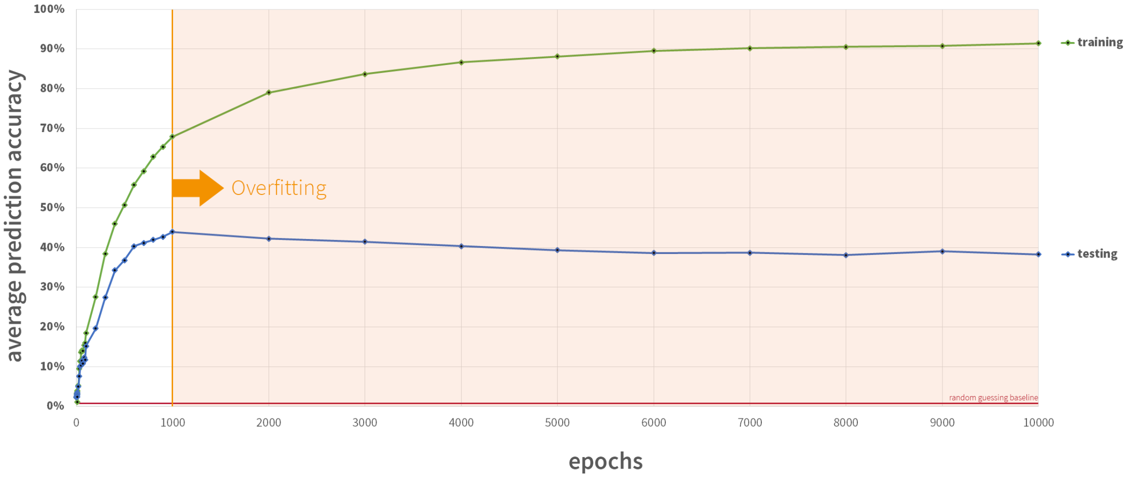


Figure 3.9: Average training and testing prediction accuracies over several iterations of IxI^4xO MLP modeling aggregated player behavior data [F3], with respect to the amount of training epochs. The orange line indicates overfitting based on a peak in testing accuracy.

Sample size

Fixing the network architecture to IxI^4xO with 1000 training epochs, the influence of the sample size (i.e. amount of data points) on the model quality could be estimated, which yields a rule of thumb for a minimal sample size constraint (cf. Figure 3.10). Within single sessions of a DPBM encounter in EC, recorded players used up to 91 skills, where the prediction accuracy converges not until approximately 200 data points. Conclusively, a single play-through is not sufficient for training and testing since situations and behavior emerge that did not appear in the training samples. From multiple play sessions on, both training as well as testing accuracy stabilize. A further reason for this convergence might be the familiarization of the player to the novel environment - when they encounter the same in-game setting again and again, their behavior is more likely to show similar patterns, compared to completely novel situations. Yet, this measurement approach can not distinguish between an increase of model quality and an increase of internal player behavior conformity. Above that, the recorded data originated from challenging an adaptive opponent where players reported that they have to rethink their behavior occasionally [$F3$].

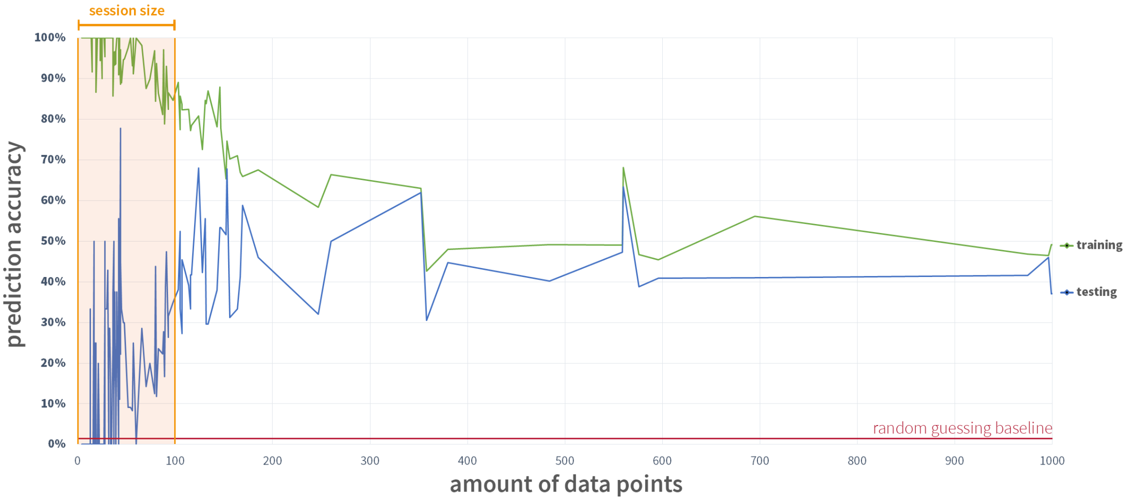


Figure 3.10: Change in training and testing prediction accuracy of IxI^4xO MLP after 1000 training epochs with increasing amount of data points. The orange window approximates the session size, where accuracy tends to converge from 2-3 sessions on.

Parameter Contribution

The input layer used in this approach (cf. Figure 3.7) is based on the insights and statements of multiple studies [$F1$, $F3$], that the reason for players' decision making considerably relies on adherence to individual skill *rotations* and deviations are mostly due to situational *responsive decisions*. In order to investigate the actual impact of these factors on the model quality and test the feasibility of sequence-focused models such as long short-term memory networks (LSTMs), parameters for those factors were taken apart and benchmarked in isolation of each other. Figure 3.11 contrasts *Responsive Decisions (RDs)*, which contain only the first 22 parameters describing the game state or situation, to *Main Rotations (MRs)* that only take the preceding action into account, as well as to the original combined (C) input layer serving as a baseline.

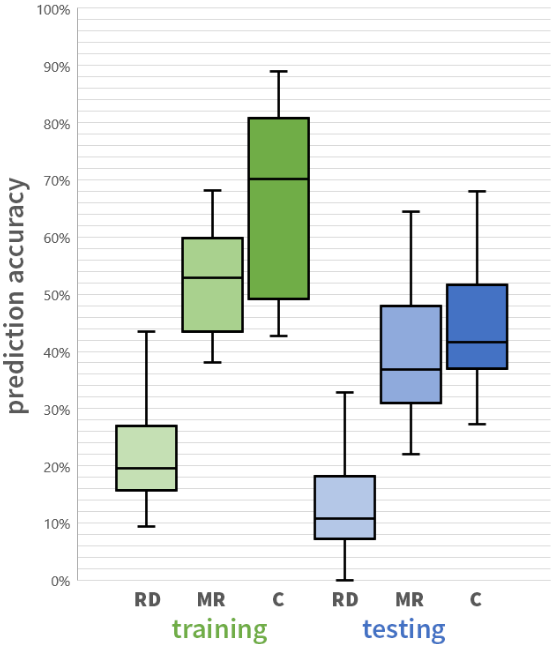


Figure 3.11: Training and testing prediction accuracy between only *RD*, only *MR* and the original combined (C) input vector.

Although the combined approach yields higher training prediction accuracy, a Welch's t-test between testing(*MR*) and testing(C) resulted in no significant difference. This suggests that rotations contribute a major descriptive part of the approach. Since composed of sequences, these make up for the evaluation of time series inclusion in the following section and introduces LSTMs utilizing only these sequences as a potential candidate for modeling.

Time series inclusion

Based on the conclusion that the predecessor action turned out to be a significant predictor, multiple preceding game states might result in a better contextualization. While the one-hot encoding of nominal skill IDs to input values causes notably large input layers that one would usually prefer to keep concise, the networks examined in this section were fed with even larger input vectors, proportional to the number of time steps included in retrospection. As seen in Figure 3.12, overfitting already starts at incorporating a single look-back step ($t = 2$), as the continuous decline in testing accuracy suggests. This indicates that this technique is not able to generalize from incorporating further retrospective states, which adds to the consideration of LSTMs as their temporally structured input and internal memorizing architecture can foster the learning of sequential successions.

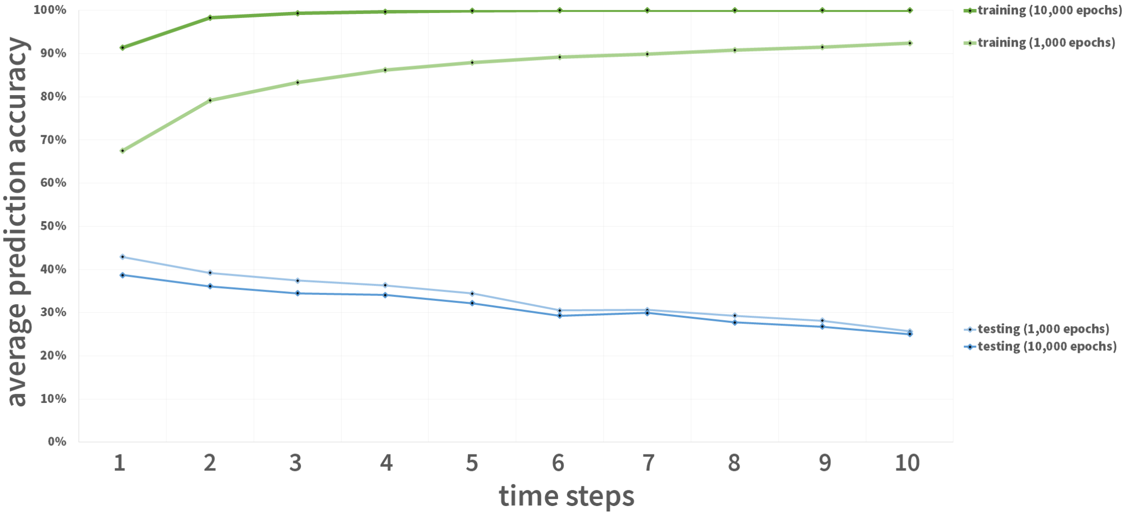


Figure 3.12: Average prediction accuracies of $(t * I)xI^4xO$ MLPs after 1,000 and 10,000 training epochs, where t is the amount of time steps included in retrospection.

Top-X accuracy metrics

As Partlan et al. (2019) and Justesen and Risi (2017) emphasize, even if human behavior follows patterns and preferences, player choices cannot be mapped from situation to action without ambiguity and decisive variance. Thus, they recommend to also compare Top-3 to Top-10 errors/accuracies among approaches, additionally to the traditional prediction evaluation that only takes the highest ranked answer (or a probabilistic choice) into account. Figure 3.13 displays that state-action MLPs are capable of representing individual player behavior within the five most probable choices with up to 84% ($M = 65.5\%, SD = 11.6\%$) testing accuracy and up to 92% ($M = 76\%, SD = 11.2\%$) within the ten most probable actions. Given the comparatively large action space of ($M = 76.2, SD = 15.1$) available skills (depending on the character class), they turn out to constitute a suitable technique for behavior modeling.

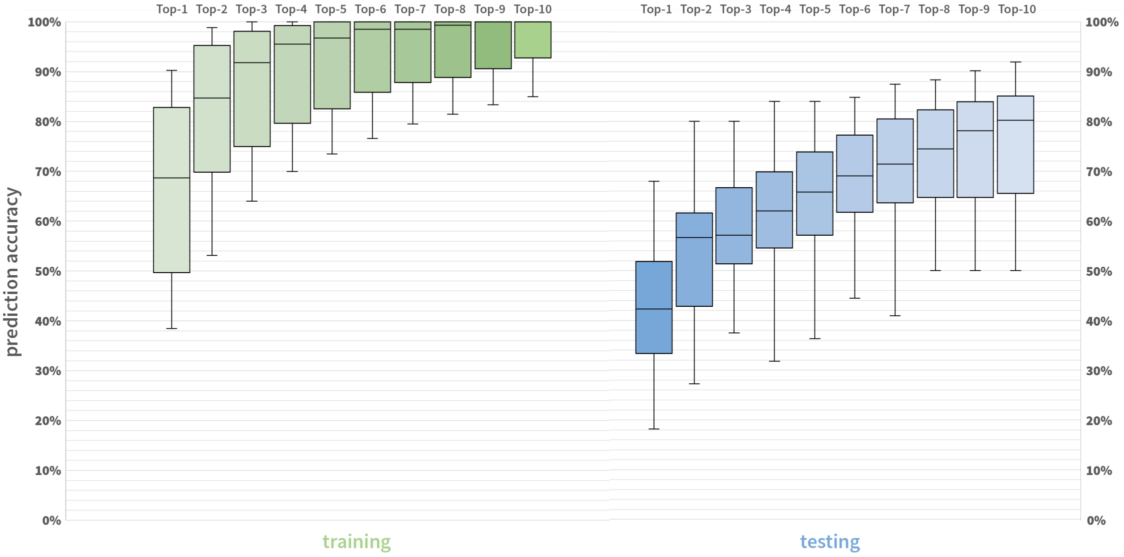


Figure 3.13: Top-1 to Top-10 training and testing prediction accuracies of a IxI^4xO MLP after 1000 training epochs.

Random Forest Classification. Since DTs proved to be a viable technique of explaining situational behavior in the initial benchmark while providing a decent prediction accuracy on the testing dataset, their conceptual successor, Random Forest Classification (RFC) (Ho, 1995), was included in this comparison. RFC establishes ensemble learning by drawing on a collection of DTs with varying parameters and computes classification based on the resulting probability distribution, which supports the mitigation of overfitting. For the purpose of parameter evaluation on this dataset, maximal depth was examined up to 30 nodes, where forest magnitudes ranged from 10 to 10000 DTs (cf. Figure 3.14). Eventually, constellations on a par with MLPs could be found ($M = 46.8\%, SD = 10.2\%$ testing prediction accuracy) around 7 nodes of maximal depth while using at least 400 DTs. The benchmark was conducted using the RFC implementation of scikit-learn 0.22.1 on an Intel(R) Core(TM) i7-8700K CPU @3.70GHz.

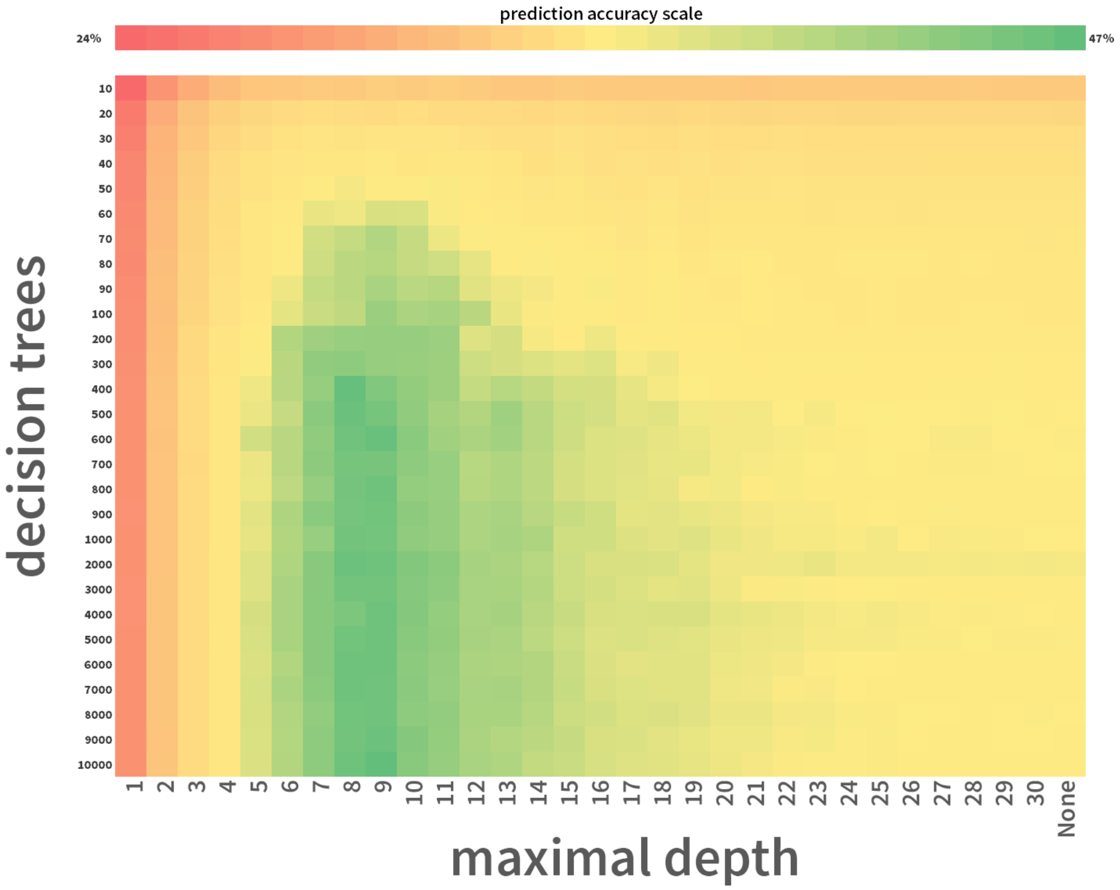


Figure 3.14: RFC testing prediction accuracy heatmap between the number of used DTs and the maximum depth within.

3.3.4 | Summary

In conclusion, MLPs as well as RFC proved to be suitable techniques for DPBM's core action prediction module (cf. Figure 3.15). The vanishingly low random guessing chance (**RND**) highlights the game environment's vast possible action space, while the **Baseline** illustrates the prediction accuracy incorporating only action frequencies, but omits information about action sequences or the contextual game state. Both MLPs as well as RFC provided high overall Top-1 testing accuracies (that were even able to be elevated with conservative heuristics) while demanding acceptable levels of computation time. As opposed to this, methods as HMMs or DTs are no longer considered to be viable candidates for DPBM since outclassed by the former. Similarly, no feasible results could be achieved for LSTMs, as they turned out to require a drastically larger sample size in order to reach a satisfying prediction level while at the same time, computational effort exceeds the former methods dramatically. For the same reason, alternative techniques such as recurrent, genetic, convolutional or generative adversarial networks were also disregarded, as the target application areas necessitate training deployable in real-time situations (e.g. to immediately substitute disconnected players) and appropriate player representations based on already small amounts of behavioral data (e.g. to quickly construct personalized DDA opponents that adapt to changes in player behavior from one

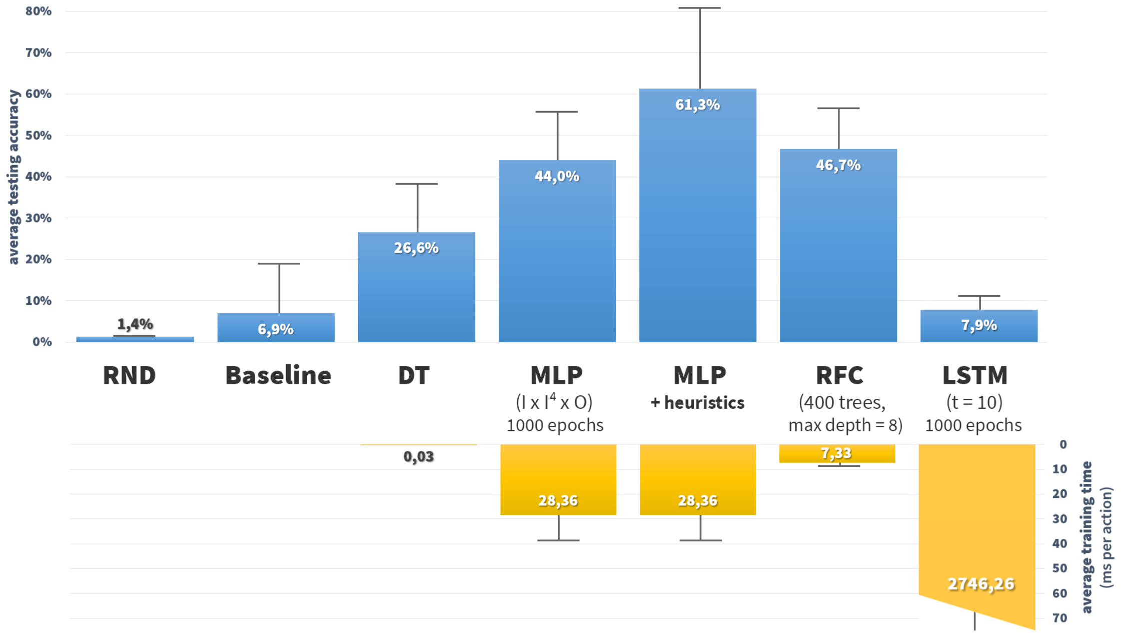


Figure 3.15: Comparison of average Top-1 testing accuracies and training times from the advanced benchmark methods

session to another). Eventually, with MLPs and RFC as efficient predictor modules, the initially mentioned design guidelines for usable game AI can be satisfied, in that they offer **believable results** (as affirmed by the following sections) at **computationally feasible** demands while still being **easy to implement** (once an in-game representation for the DPBM architecture is realized).

Notably, both of the benchmarks handled in this section only considered the numerically measurable prediction accuracy on the testing dataset. While this remains the measure most often used within the field, it neglects factors that are likely to impact individual likeness or believability. In the study of *Enemy Within* [*F*3] where this data is drawn from, limitations arose that replicating decision making is not the only dimension attributable to human player behavior. Thus, Section 5 acknowledges these shortcomings and proposes additional modules aiming at representing individual proficiency concerning precision, cognitive computation and reaction times, target selection and global as well as local movement.

3.4 | Dynamic Difficulty Adjustment

The following section summarizes the developments and advances made within the application field of dynamic difficulty adjustment. Within two evaluations, intrinsic motivation, representativeness and long-term commitment could be measured, contributing empirical evidence towards all constructed research sub-questions **Q1-Q4**.

3.4.1 | Comparison to Heuristic Opponent Behavior

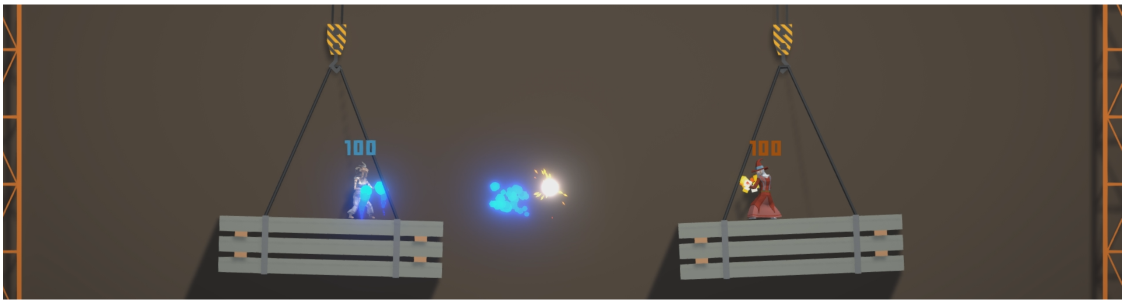


Figure 3.16: In-game screenshot of *Korona:Nemesis* used in multiple studies [*S*2, *F*2]. The player on the left utilizes **Water** to counter a **Fire** projectile.

To assess the player experience of challenging a DPBM-fueled opponent, the least confounded comparison would be against ordinary computer controlled opponents. Thus, three types of heuristic opponents (*basic:* predictable decision making, weak performance; *random:* unpredictable, moderate performance; and *optimal:* ideal, strong performance) were contrasted with DPBM (imitative, approximately player's performance) during the course of a two-week study [*S*2]. Since differences between these opponents had to be clearly distinguishable in a short amount of time and decision making should be the major underlying game mechanic (as opposed to movement or accuracy), the elemental fighting platformer *Korona:Nemesis* (cf. Figure 3.16) was constructed that facilitated individual strategies by providing an extended Rock-Paper-Scissors paradigm with no completely dominated strategies. Within the game, players can move, jump, shoot an elemental projectile or switch their elemental stance. When a projectile hits the opponent, a certain fraction of their HP is deducted, which is doubled in the case of a critical hit (when the projectile is superior to the opponent's stance, cf. Figure 3.17). As soon as only one player remains alive, the next level is presented, up to a prior defined limit (in the case oft this study [*S*2], a player fought ten levels against each opponent type).

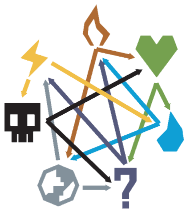


Figure 3.17: Elements in *Korona:Nemesis* and their respective interactions. Arrows illustrate when an element hits another critically, yielding symmetrical, extended Rock-Paper-Scissors dynamics.

The hypotheses of the study [*S*2] stated that players are less engaged in competing against too weak (*basic*) or too strong (*optimal*) encounters, but prefer a balanced opponent. In the case of the unpredictable *random* enemy, this balance should emerge from the symmetric game dynamics, since no matter what element the player chooses, the *random* opponent is equally likely to reach a superior choice as it is to make an inferior one. On the other hand, the DPBM opponent would approximate the player's proficiency implicitly by imitating their state-action behavior and reach a similar level of balance. Yet, challenging oneself might encourage the player to constantly self-improve and bears an even higher potential of increasing intrinsic motivation.

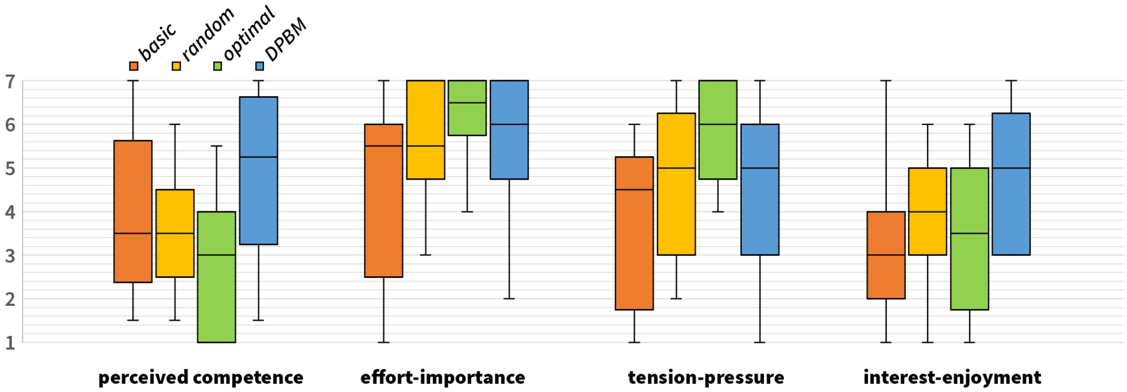


Figure 3.18: Results of the Intrinsic Motivation Inventory subscales for [*S*2].

Eventually, the hypothesis could be confirmed with quantitative insights via IMI (cf. Figure 3.18) and supported by qualitative statements. Even though players felt competent encountering DPBM, they made high efforts and felt appropriately tensioned. Most importantly, DPBM significantly outperformed all alternatives for **interest-enjoyment** ($p < .01, d = .75$ against basic, $p < .05, d = .57$ against random and $p < .05, d = .54$ against optimal), leading to a measurable increase in intrinsic motivation. These outcomes deliver first evidence for **Q3** in showcasing the motivational potential of DPBM as opposed to traditionally heuristic opponents, as well as for **Q4** (regarding implicit DDA) as a consequential effect. Viable testing accuracies of ($M = 70.3\%, SD = 13.5\%$) from individual MLPs (cf. Figure 3.19) further add to the practicality question of **Q1**.

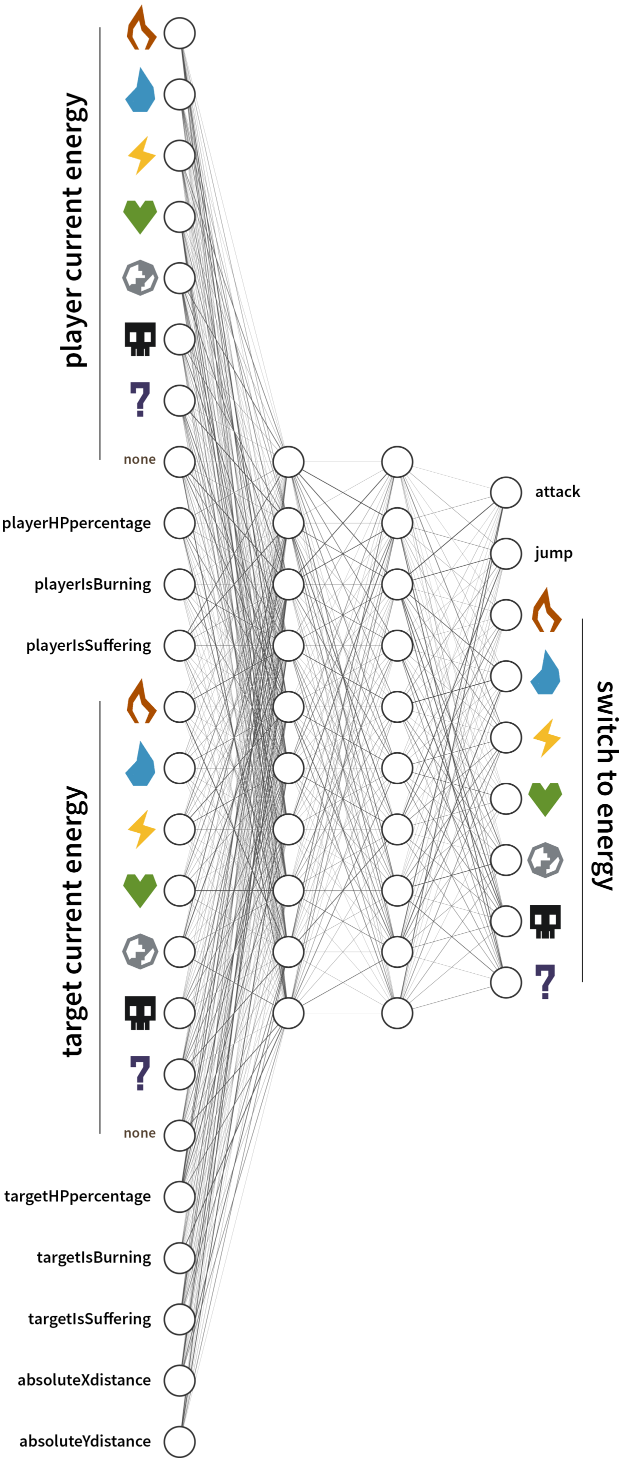
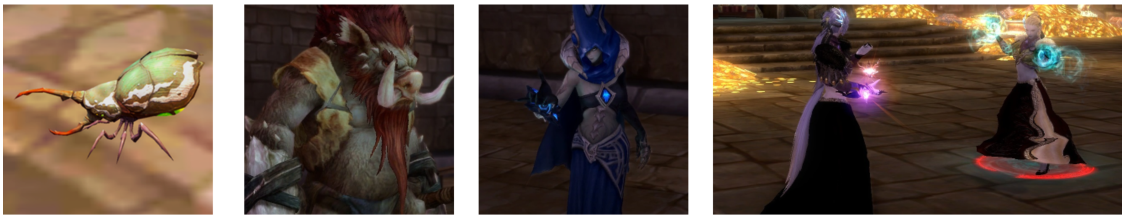


Figure 3.19: DPBM architecture utilized for each individual player in *Korona:Nemesis* $[S2, F2]$, mapping game state (information about player and closest target) to action probabilities.

3.4.2 | Long-Term Commitment



α: frequency β: perseverance γ: disturbance **DPBM**

Figure 3.20: Four opponent types in *AION* ([*F*3]), where α, β and γ employed traditional parameter tuning DDA (*rubber-banding*).

Based on the previous positive results [*S*2], *Enemy Within* [*F*3] sought to overcome potential weaknesses of the former and evaluate DDA capabilities of DPBM on a larger scale. First of all, it [*S*2] might have suffered from the probable influence of the novelty bias, where the outcome of the evaluation could have been influenced by the effect that participants played the game or experienced the particular setting or challenge before. Above that, the comparison only included DPBM and heuristically driven opponents without any form of adaptivity. Lastly, the study lacked a proper assessment of the perceived representativity, so no evidence that players actually rediscover their own behavior in their opponents could be accumulated yet. Thus, *Enemy Within* [*F*3] was placed in the popular MMORPG *Aion*, exposed to a community with months to years of prior game experience and conducted over the course of four weeks where subjects could participate up to once daily. Additionally, DPBM was contrasted against not only heuristically driven opponents, but a traditional DDA system employing *rubber-banding* throughout multiple parameters. Without exposure of the DPBM opponent's behavior in advance, players had to explain its decision making with their own words. The outcomes of *Enemy Within* [*F*3] contributed greatly to the research agenda of this thesis through its ecologically valid long-term field study. Even in the large action space of the MMORPG (cf. Figure 3.21), DPBM turned out to be a viable technique for approximating individual behavior, as testing prediction accuracies ($M = 60.6\%, SD = 22.6\%$)

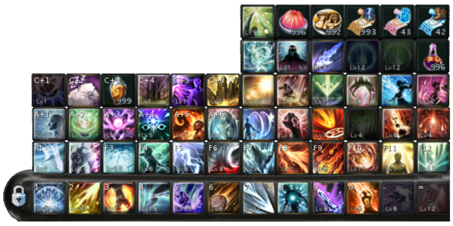


Figure 3.21: Exemplary arrangement of a subset of skills available to the Sorcerer class in Aion. Additionally, context-dependent skills (when the player or a target opponent is in a particular condition) and a multitude of items can be activated.

from daily trained networks (cf. Figure 3.7) suggest and qualitative statements confirm (*"at first he randomly used skills that I also used, later he added my combos"*, *"tried to replicate my own skills and techniques"*, *"it was hilarious when I played against myself"*). The adaptive instance dungeon *Eternal Challenge* constructed for this study and consisting of *rubber-banding* as well as DPBM opponents, managed to motivate players on a consistent scale, even after the predicted initial novelty spike (cf. Figure 3.22), where DPBM turned out to be the greatest motivational impact (cf. Figure 3.23).

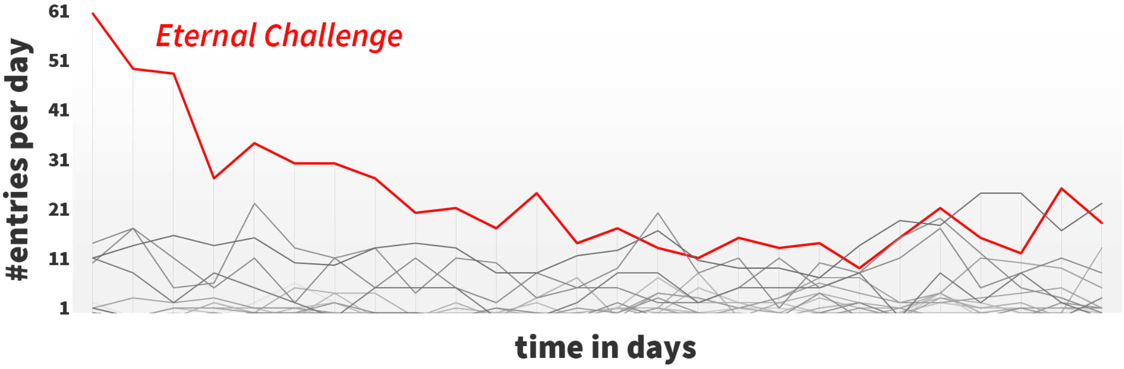


Figure 3.22: Daily number of unique players entering *Eternal Challenge* compared to all other available instances during the study period [*F*3].

The outcome that DPBM outperformed all parameter tuning DDA opponents in terms of IMI's *interest-enjoyment* indicates a high "fun factor", while *tension-pressure* and *effort-importance* highlight the considerable challenge, leading to an overall higher intrinsic motivation and linked potential to induce flow. The actual implicit DDA capabilites of DPBM are backed by qualitative statements that reveal an appropriate challenge, a noticeable difficulty adjustment over time and the perception of playing against an equal opponent that facilitates rethinking of habitual behavior (*"quite easy at first but afterwards I really was busy thinking about how I approach him"*, *"it's almost as good as I am"*). Eventually, the dataset describing behavior of 171 players in recurring sessions over four weeks led to the investigation of the Advanced Benchmark (cf. Section 3.3.3) used for deeper progressions of DPBM in general. Thus, *Enemy Within* [*F*3] supplied additional evidence for **Q1** that behavior can be assessed and reproduced, with the additional conclusion from qualitative measures that DPBM opponents incorporate the individual player's strategies, tactics and preferences, supporting **Q2**. The consistent commitment over a longer study period, paired with the significantly higher intrinsic motivation than traditional DDA parameter tuning responds to **Q3** and **Q4** in favor for DPBM.

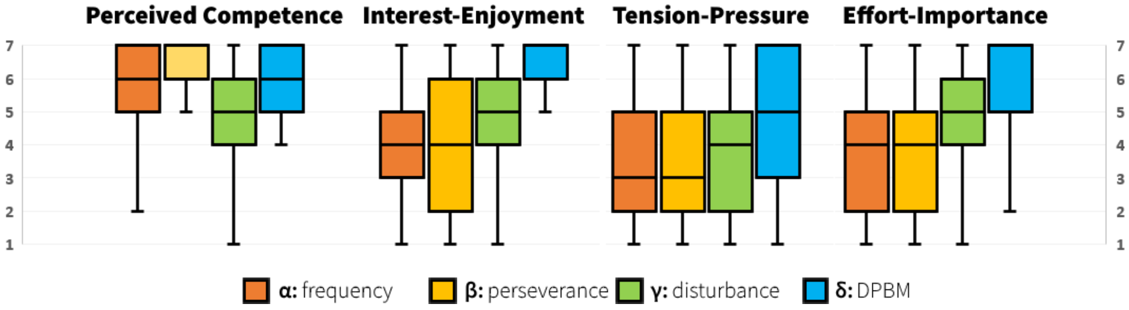


Figure 3.23: Results of the Intrinsic Motivation Inventory subscales for the compared traditional DDA variables and DPBM [*F*3].

3.5 | Player Substitution

Tackling the prevention of match disruptions in online games caused by network stability issues, *Bot or not* [*F*2] implements DPBM for the continuous training parallel to active online games and the automatic substitution of disconnected players by their individual representative DPBM agent. As previously used [*S*2], *Korona:Nemesis* was chosen for observation during its launch on the leading distribution platform *Steam*, where the appearance of a scientific study was concealed until the post-session questionnaire to not threaten the ecological validity of the field study. To control for the examined variable of substituting, the study setup of Figure 3.24 was developed, including four players. From an initial configuration, one human player was shifted into a mirrored match with substituted opponents after a random point of time, while in the original match the player is replaced utilizing a DPBM bot trained on the prior behavior (adopting the previous architecture [*S*2], cf. Figure 3.19). When training was not completed yet or players deliberately disconnected from the game before displaying enough behavior information for training, a heuristic *random* bot was used for substitution, based on prior insights [*S*2] that it at least offers a moderate challenge.

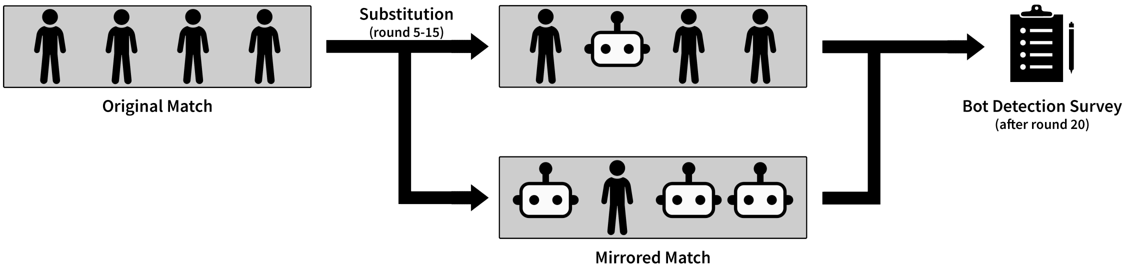


Figure 3.24: Study sequence for each match [*F*2].

		actual behavior		
		human	DPBM bot	heuristic bot
guessed behavior	isHuman	87.18% (68)	85.48% (53)	32.75% (75)
	isBot	12.82% (10)	14.52% (9)	67.25% (154)

Table 3.4: Percentages (and absolute numbers in parentheses) of bot detection estimates, according to the responses to the in-game bot detection survey [*F*2].

After a match consisting of 20 levels, participants were asked to judge if and which other players were human or computer-controlled (*"bots"*). Throughout 206 multi-player sessions, ($n = 312$) players submitted bot detection responses and 24 of these additionally completed an optional, web-located questionnaire that asked for further quantitative and qualitative measures. Table 3.4 demonstrates that participants were unable in distinguishing actual humans from agents deploying DPBM ($p > 0.05$), where they were indeed able to tell heuristic bots apart from humans ($p < 0.05$) or DPBM bots ($p < 0.05$). From those players who did notice the substitution, no significant change in performance or predictability caused by the substitution could be found (cf. Figure 3.25).

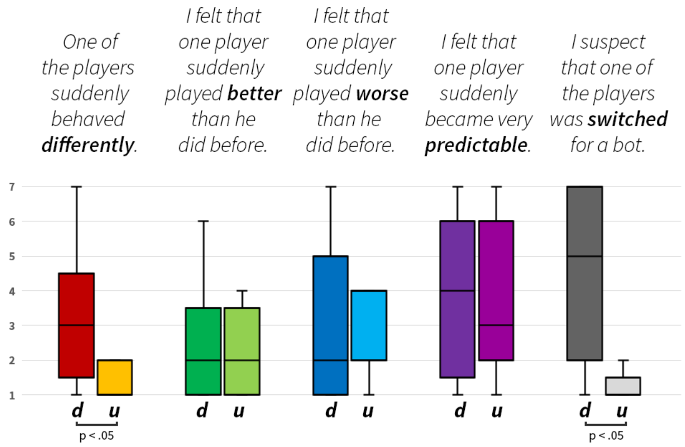


Figure 3.25: Boxplot illustrating the results of the custom awareness scale between players that detected *(d)* a bot and players unaware *(u)* of substitution [*F*2].

Bot or not $[F2]$ kept track of all DPBMs trained throughout the online sessions, resulting in ($M = 82.17\%, SD = 23.17\%$) testing prediction accuracy and a strong positive correlation between the amount of data points used for training and the resulting testing accuracy (Pearson's $r_{2871} = .64, p < .01$), which further strengthens the proof of technical implementability (**Q1**). Most importantly, the indistinguishability of DPBM agents and human players makes a strong claim about the capability of believably imitating individual player behavior (**Q2**). Even in detected replacements, participants only noticed the substitution itself, but did not perceive these as more or less proficient or predictable, which altogether delivers first evidence of the potential of DPBM to bridge undesirable online disruptions (**Q4**).

3.6 | Automated Game Testing

3.6.1 | Autonomous Solving

In order to augment the field of automated game testing, I designed and developed *Intelligent Completion of Adventure Riddles via Unsupervised Solving (ICARUS)* in collaboration with Daedalic Entertainment[4], a generic adventure game solver suitable for all of their in-house products ([*S*1]). Point-and-click adventures usually consist of deterministic procedures through symbolic environments, where players have to solve tasks by interacting with objects or characters, using items on these targets, combining items or other special actions. Figure 3.26 shows a single scene of *Anna's Quest* (Daedalic Entertainment, 2015) containing 19 targets and 11 items currently in the player's inventory which amounts to 379 possible actions. Given that most of the time not only one of these scenes is visitable, combinatorial explosion renders exhaustive human playtesting extremely inefficient and tedious for the respective testers. Yet, a complete search of state-action pairs would be best suited in order to detect the most errors possible. A random guessing solver that blindly chooses actions would already reduce the human testers' burden, but is likely to overlook actions that lead to bugs or glitches. Thus, *ICARUS* approached autonomous solving using discrete reinforcement learning by mapping which action leads to actual game progress given a particular game state (cf. Figure 3.27. De-

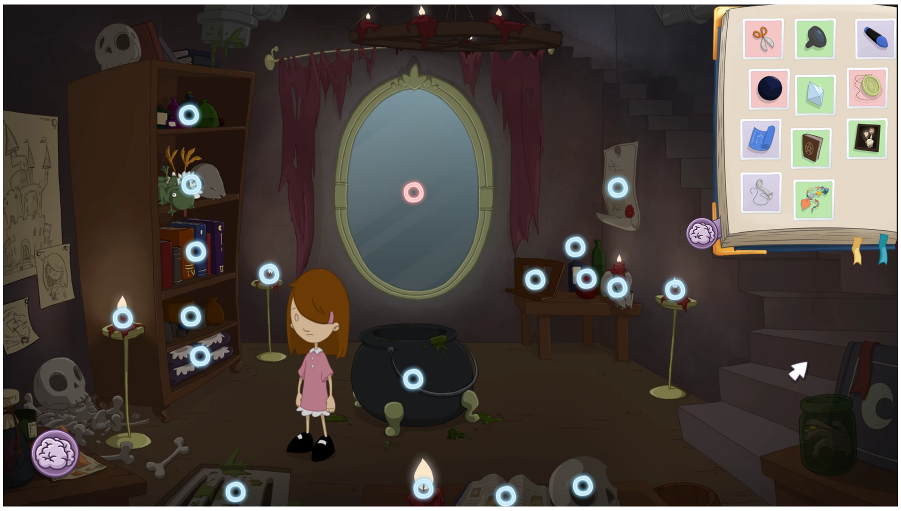


Figure 3.26: Scene from *Anna's Quest*. Circles represent objects (blue) or characters (red), which can be viewed, used or combined with one of the items (upper left corner). Additionally, items can be viewed, used or combined with other items.

[4] https://www.daedalic.com/

pending on the purpose, this progress is treated as either negative or positive reward, since *avoiding* progress eventually leads to a breadth-first search checking all possible actions for errors and *optimizing* for progress eventually leads to a depth-first search that yields to an efficient verification of overall playability (i.e. if the end of the game can be reached). *ICARUS* managed to aid playtesting by autonomously detecting game crashes, freezes, blockers and was additionally deployed in semi-autonomous test sessions where it automatically plays games, but human testers pinpoint graphical flaws, typos or glitches.

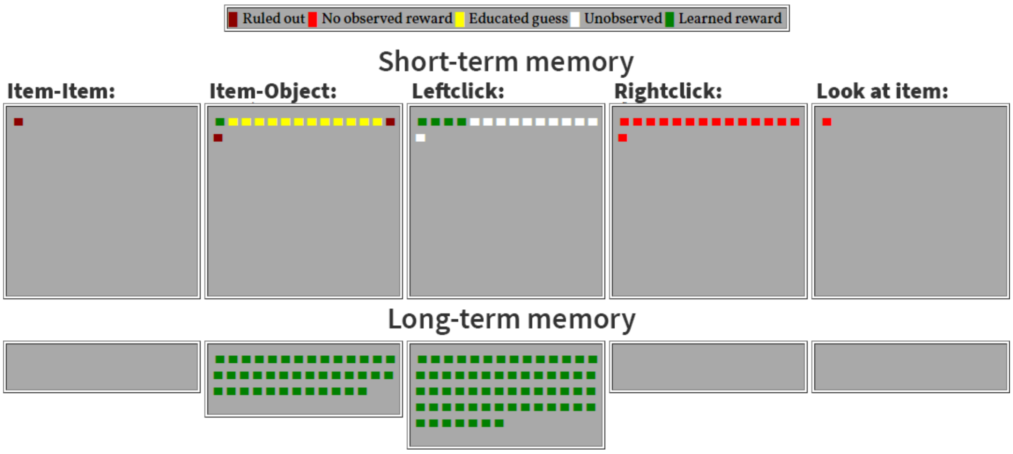


Figure 3.27: Reward map visualization of a single game state in *ICARUS* ([S1]).

Nevertheless, neither the optimal depth-first, nor the comprehensive breadth-first search through the game state space turned out to be sufficient to cover all potential error sources of human playing, since particular game states were not reached that followed from particular, non-optimal sequences of actions that human testers executed. On the other hand, simply recording testers' playtraces and reiterate them on following test runs proved also be not enough, since different game versions appear daily during development and completely mirroring human decisions led to dead ends there. Thus, the core *ICARUS* reinforcement learning module was combined with the previously outlined DPBM state-action player modeling architecture. Due to the deterministic and sequential nature of adventure games, game states could be recorded and recognized in a very precise and symbolic manner, leading to clearly distinguishable sequences of state-action pairs. Eventually, *ICARUS* was configured to favor human players' decisions when a recorded state-action pair is available and continues to follow rewards from the reinforcement learning module elsewise. By creating more exhaustive and human-like search patterns, this contributes to **Q4** as a functional and industrially deployed example of automated game testing.

3.6.2 | Automated Balancing

Extending the scope of automated testing to temporally tedious balancing issues, *Dungeons & Replicants* [F4] successfully demonstrates how DPBM can aid developers in detecting and regulating imbalances, incorporating the actual behavior of a player population for balance simulations instead of heuristic or optimal agents. The dataset of *Enemy Within* [F3], comprising atomic behavioral data of one-versus-one combat situations of the MMORPG *Aion*, was extended to include information of 6 months and $n = 213$ players in total and served for the generation of just as many DPBM-driven agents. In order to detect viability differences between choosable in-game classes of *Aion*, DPBM agents were grouped into the classes of their original players and benchmarked against first a set of opponents with incrementally increasing difficulty and eventually against each other. In the initial PvE evaluation, each substitute faced 100 heuristic encounters, where the offensive and defensive parameters of the latter were manipulated in order to investigate the performance range from fighting trivial to barely defeatable foes.

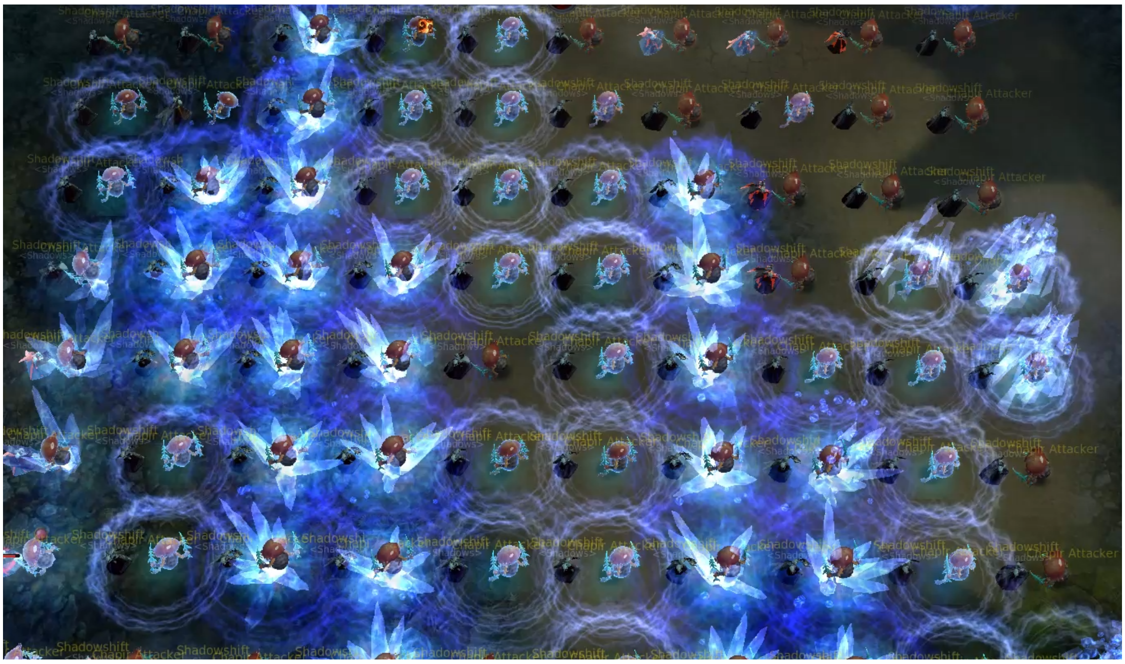


Figure 3.28: In-game screenshot of the PvE benchmark in *Aion* (NCsoft, 2008). DPBM-driven player replicas encounter 100 heuristic opponents with increasing difficulty in one-on-one situations (attack horizontally, maxHP vertically). Depending on the game state between the agent and its target, emerging behavioral patterns for action preferences and sequences can be observed.

Figure 3.28 visualizes the setup of DPBM agents of a single individual player, where the *attack* value of the opponents was increased on the horizontal axis and the *maximal health points* value increased on the vertical axis. In effect, the opponent of the lower left corner should be trivially, but the one of in the upper right corner hardly beatable, leading to a distribution of performances. To measure these performances and aggregate them to an overall score, a proficiency metric (ϕ) was constructed, incorporating the binary value of having won against the opponent (w), the normalized temporal duration of the fight (t), the remaining health point percentage of the agent (hp_a) and that of the opponent (hp_o):

$$\phi = \sum_{i,j=0}^{n} \frac{\alpha w + \beta(1-t) + \gamma hp_a + \delta(1-hp_o)}{(\alpha+\beta+\gamma+\delta)n^2}$$

Proficiency distributions (ranging from worst-case (0) to optimal (1) performance) over these 100 trials depict the *"soft boundaries"* of the individual agents (as visualized in Figure 3.29). Above that, the eventual proficiency ϕ could be used to compare the viability of different classes, based on the actual strategies that individual players employ (cf. Figure 3.30 (left)).

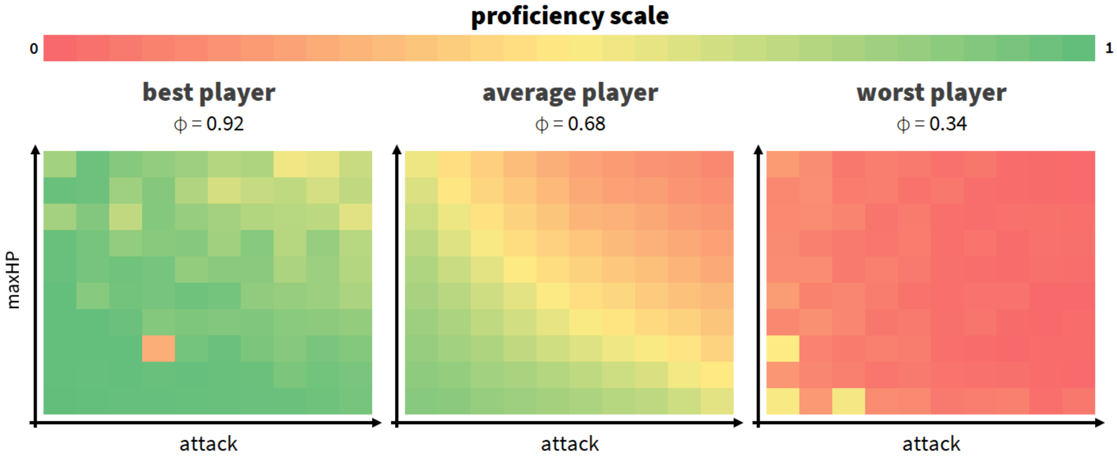


Figure 3.29: Proficiency heatmaps of the best, average and worst player replication of the benchmark. The horizontal axis denotes the increasing attack value of the heuristic opponent while the vertical axis describes the increasing HP value of it (+25% per step, respectively).

As significant differences of proficiency between in-game classes could be found ($F(9, 203) = 9.63$, $p < .01$; partial $\eta^2 = 0.3$), *Dungeons & Replicants* [F4] subsequently introduces one method of parameter adjustment per class by calculation of the mean squared error to a target proficiency and finding the center of mass in the respective parameter distribution, leading to a more balanced outcome without significant proficiency differences ($F(9, 203) = 1.42$, $p > 0.05$, cf. Figure 3.30 (right)).

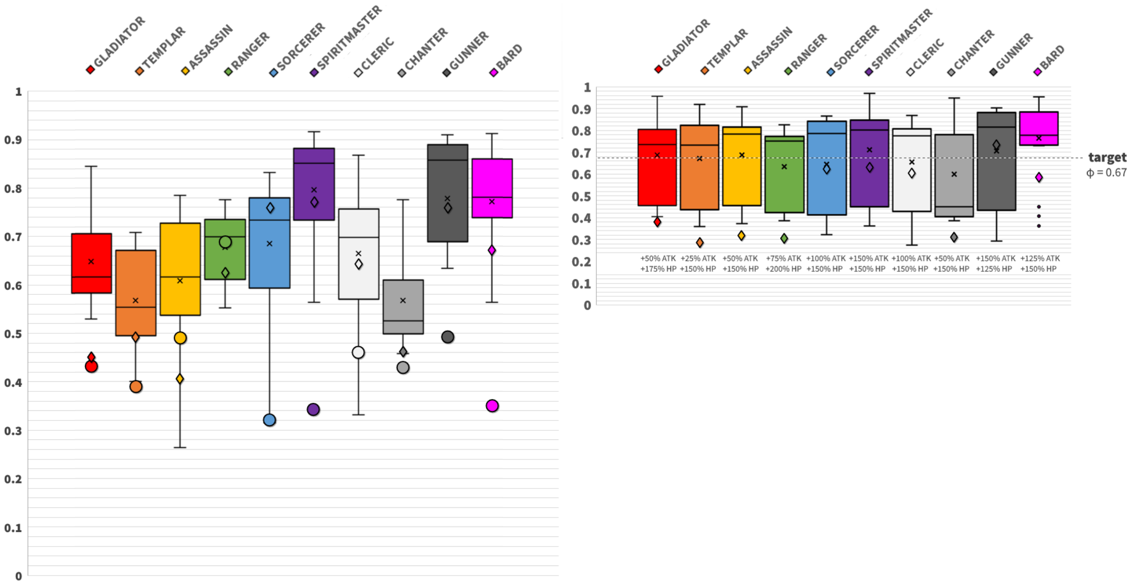


Figure 3.30: Proficiency results of player replicas across different classes from the PvE evaluation (left) and after parameter regulation (right). Includes means (indicated by **x**), medians (–) as well as the proficiency of a generalized model of the class (◊) and random play (O).

To not only be constrained on comparing differences between classes with respect to PvE encounters, but also include the internal balancing of classes, an additional Player versus Player (PvP) evaluation was conducted in which all of the 213 player replicas fought out one-on-one battles against each other. In effect, this was able to illustrate an extended rock-paper-scissors superiority scheme in which e.g. Melee classes outperform Rangers, Rangers dominate Magic classes and these outclass Melees, which is likely to reflect the developers' original design. By providing a technique that integrates the actual behavioral patterns of a whole player population into simulations, balance discrepancies could be detected and regulated successfully, effectively contributing to **Q4**. The insights that this distribution of individual player proficiency resembles the original players' performance closer than a generalized model or random play further add to **Q1** in that DPBM is a suitable mechanism for representing player behavior.

3.7 | Serious Games

To advance the field of serious games and integrate all previously evaluated mechanisms into the domain of games with a purpose (GWAPs), a series of human computation games was designed and developed during the formation of this thesis. Insights of serious game design and evaluation supported this development, as the degree of gamification, adaptation and visualization significantly impact the intrinsic motivation required for genuine data contribution $[A1 - 8]$.

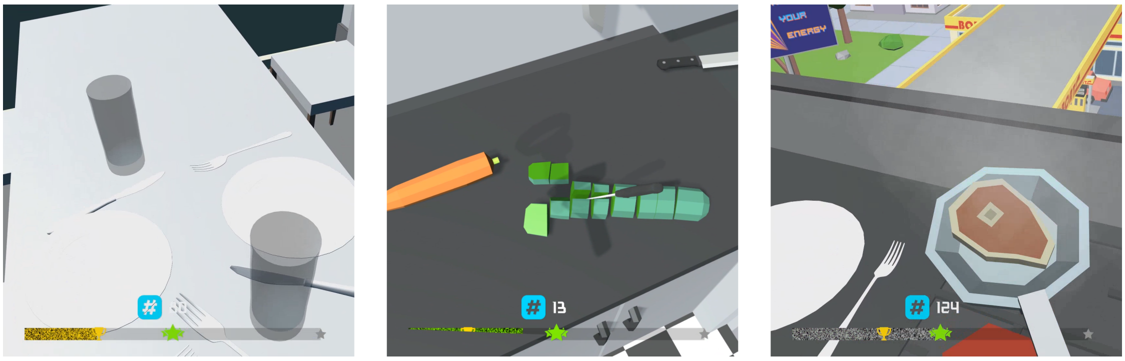


Figure 3.31: Screenshot excerpts from *Kitchen Clash* [*A*4]. Within a generative, natural language processing procedure, task descriptions for everyday activities are derived from written online corpora and have to be completed in a Virtual Reality (VR) game environment. Subsequently, compound executions are graded in terms of quantitative efficiency and qualitative assessments. While general object constraints marked requirements for completing the tasks, the actual action sequence and specific object usage could be utilized to solve underspecified roles and estimate human-like preferences distributions.

With increasing utilization of DPBM, *Kitchen Clash* (cf. Figure 3.31) [*A*4] as well as *Tool Feud* (cf. Figure 3.32) [*A*5] make use of the promoted state-action architecture to infer action and object preference distributions, while *Elevator Empire* (cf. Figure 3.33) extends this by incorporating *DPBM* for large-scale online agent management and player substitution for asynchronous representations. In effect, all of these accumulate symbolic world knowledge that is processed and used to augment robotic decision making in the context of everyday activities (*EASE*[5]) by modeling action sequences, object or tool preferences and filling underspecified semantic information. In the case of *Elevator Empire*, players train multiple DPBM-fueled agents in parallel in order to optimize everyday activity processes in a hotel management setting. They are encouraged to constantly improve their own agents' routines to be able to progress and withstand

[5] https://ease-crc.org/

competition, while they additionally can infiltrate rival establishments to sabotage their processes. Apart from producing world knowledge, this additionally provides labeled data for the discrimination of valuable and malicious behavior, demonstrating a feasible approach for the threat of human computation data corruption (as outlined in [$A2$]).

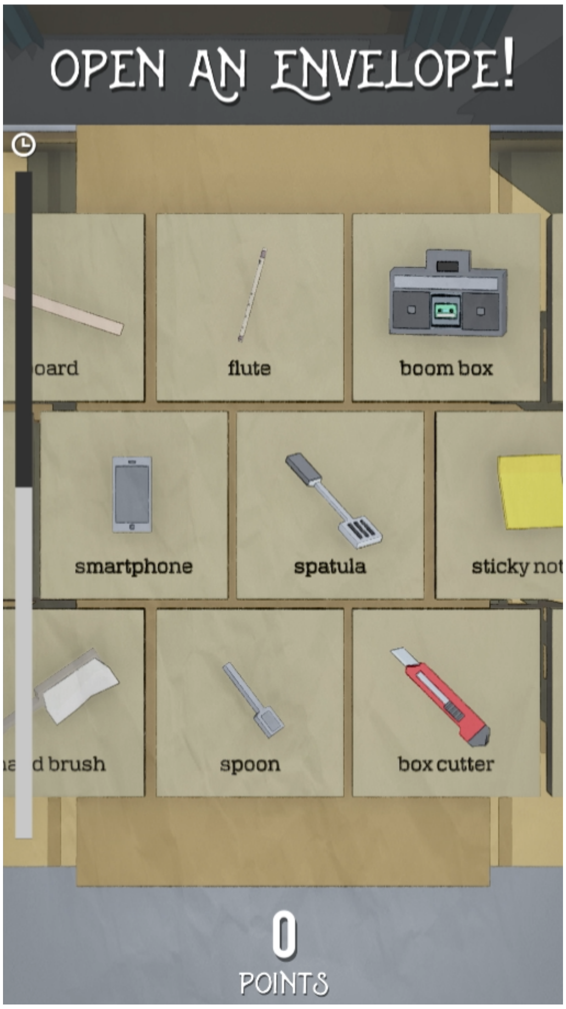


Figure 3.32: Screenshot from the mobile game *Tool Feud* [$A5$]. Task descriptions for everyday activities are presented with a constantly changing set of tool choices, where players have to choose the most appropriate. DPBM's preference distribution of the whole population are utilized to model the correctness of the respective answers, yet the most popular choice(s) are dynamically removed to filter out obvious answers and acquire alternative solutions.

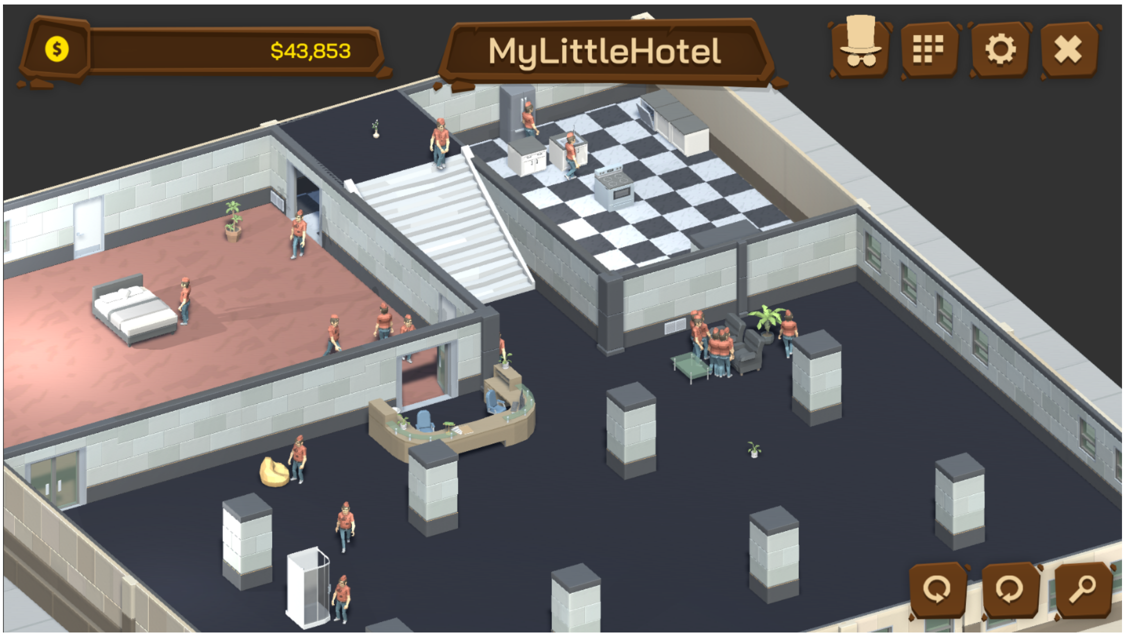


Figure 3.33: Screenshot from the online multi-player game *Elevator Empire*. While players seek to optimize and expand their own hotel enterprise, they have to train and manage their constantly increasing staff. The behavior of these employees is manifested by DPBM and manipulated by demonstration, leading to a successively accumulating optimization problem that adaptively challenges the player while simultaneously aggregating world knowledge about everyday activities.

4 Contributions Towards Research Questions

To be able to answer the overarching research question, it was split into the following four major sub-questions (cf. Section 3.1). With respect to the publications $[F1-4, S1-3]$ that constitute this thesis, evidence is consolidated and presented below.

Q1. Can generative player modeling be utilized to reproduce individual player behavior with measurably similar decision making?

In order to investigate similarity between original and replicated behavior, multiple measures had to be examined, since quantitative validation renders DPBM outcomes immediately comparable to different methods and approaches, whereas qualitative assessments highlight the subjective experiences of players exposed to DPBM agents. Separating these fundamentally different measurement approaches, **Q1** collects evidence for quantitative feasibility, whereas **Q2** focuses on assessments of individual participants. From the publications that contain an implementation or benchmarking of DPBM techniques, each reports on the measured prediction accuracy when evaluating individual player models on a testing set that comprised 20% of the overall training data. Bearing in mind that these studies were conducted in different games, genres, setups and with different purposes, DPBM consistently reached high accuracy values $(M = 71.4\%, SD = 13.2\%)$ $[F1]$, $(M = 82.2\%, SD = 23.2\%)$ $[F2]$, $(M = 60.7\%, SD = 22.6\%)$ $[F3]$, $(M = 61.3\%, SD = 22.4\%)$ $[F4]$ and $(M = 70.3\%, SD = 13.5\%)$ $[S2]$, compared to the marginal baseline accuracies of these high-dimensional action spaces.

Above that, the advanced benchmark (cf. Section 3.3.3) further investigates the impacts of hidden layers and nodes amounts, parameter contribution and time series inclusion, visualizes the influence of sample size and training epochs and displays testing prediction accuracy of DPBM according to the Top-X notation, where the dataset of *Enemy Within* [*F*3] can be expressed with e.g. up to 92% ($M = 76\%, SD = 11.2\%$) within the ten most probable actions. Together with the insights of *Bot or not* [*F*2] that DPBM is continuously improving with time and data (Pearson's $r = .64$ between testing accuracy and sample size) and is able to represent individual proficiency ($r = 0.91$ between original player's and the DPBM replica's score outcome) [*F*1], the state-action architecture and deep learning paradigm of DPBM turned out to be a suitable technique for individual generative player modeling.

Q2. Can generative player modeling convince players that it imitates individual behavior believably?

While believability or human-likeness was only examined in terms of general similarity in prior related work, this thesis focused on assessing individual behavior explicitly. In *Enemy Within* [*F*3], players encountered a DPBM-fueled opponent up to once daily over the course of four weeks. Without previous explanation or framing regarding this opponent's behavior, participants reported that after a couple of sessions it actually started to replicate their behavior, approximated their combos and evolved into a challenging contestant almost as good as themselves. In the even more drastic study setup of *Bot or not* [*F*2], one of four players in an online match was removed from the session and immediately substituted by their DPBM surrogate after a random period of time. To estimate whether DPBM could be an appropriate solution for temporary match disruptions (e.g. due to connection loss), participants were asked to judge if one of their fellow players was replaced by a computer controlled agent. The result that DPBM substitutes (85.5% judged as human) were indistinguishable from actual human players (87.2%) – while traditional heuristic bots (32.8%) were notably detected most of the time – and even the detected replacements did not differ in perceived proficiency or predictability, proves that DPBM is able to successfully convince players to replicate individual behavior, only supported by the qualitative descriptions of *Enemy Within* [*F*3].

Q3. Can challenging artificial agents that employ the player's individual decision making lead to a motivating experience?

To assess the player's motivation while challenging DPBM-driven opponents, the *Intrinsic Motivation Inventory* (IMI) was consulted and evaluated throughout the subscales *perceived competence*, *interest-enjoyment*, *tension-pressure* and *effort-importance*. Hypothesizing that an opponent which resembles individual strengths and weaknesses in a constantly updating loop leads to a notable challenge that elevates *tension-pressure*, where the incentive to "defeat oneself" implicitly increases *effort-importance* and *interest-enjoyment*, multiple studies [$S2, F3$] were conducted in different setups. One [$S2$] contrasted DPBM to heuristic opponents representing "too weak", "too strong" and "reasonable proficiency, but non-adaptive" enemies. Challenging the DPBM opponent resulted in high values of *effort-importance* ($M = 5.57, SD = 1.95$) and *interest-enjoyment* ($M = 5, SD = 1.57$), where the latter was significantly higher than all alternatives, which was backed by qualitative statements. In *Enemy Within* [$F3$], players evaluated the experience of daily encounters of their DPBM counterpart after a study period of four weeks. In contrast to *rubber-banding* DDA enemies, DPBM significantly outshined all other opponent types in terms of *tension-pressure* ($M = 4.91, SD = 2.15$), *effort-importance* ($M = 5.87, SD = 1.63$) and *interest-enjoyment* ($M = 6.17, SD = 1.11$). When asked for their personal opinion, participants emphasize the notable entertaining factor of the DPBM opponent and appreciate the increasing amount of challenge through behavior approximation. Since DPBM-driven opponents yielded considerable absolute IMI scores, positive qualitative appraisal and outclassed both traditional heuristic enemies as well as foes implementing established *rubber-banding* DDA, it can be concluded that they indeed can lead to intrinsically motivating experiences.

Q4. Can generative player modeling contribute added value to unresolved issues within dynamic difficulty adjustment, online disruptions and playtesting in ecologically valid game scenarios?

Sustaining the ecological validity of the publications, all studies that evaluated the impact of DPBM as the decision making module for in-game agents were conducted using a field study approach where the appearance of a scientific study was concealed during play and only revealed with the final questionnaire (*Can you rely on human computation?* [*A*2] additionally highlights the importance of this procedure). To this regard, the game *Korona:Nemesis* was inofficially published as a Beta test and advertised via online forums for its first study [*S*2] before released on the game distribution platform *Steam* together with the course of *Bot or not* [*F*2]. *Enemy Within* [*F*3] was conducted on a private server of the MMORPG *Aion* with an existing player base as part of one of the regular updates. This procedure made it possible to aggregate real-world human player data as well as to assess game experiences within the actual population. With respect to DDA, existing approaches rely on manually defined parameters and constraints, where the major motivational potential emerges from regulating the challenge. The first take on DDA [*S*2] successfully demonstrated that DPBM can produce decision making for game opponents that are intrinsically motivating, where *Enemy Within* [*F*3] additionally reveals that these even outclass established DDA implementations such as *rubber-banding*. With the mostly unrecognized (and otherwise as fair classified) substitution in *Bot or not* [*F*2], DPBM introduces a novel approach to bridging temporary online disruptions. Previous industrial implementations relied on choosing conservatively weak substitutions that often spoiled matches for teammates and opponents alike, while scientific strategies focused solely on the avoidance of disruptions through hardware and protocol improvement so far that are unlikely to extinguish the problem completely. *ICARUS* [*S*1] adds to the viability with respect to industrial application by demonstrating a working example of the benefits of DPBM in augmenting automated game testing software that is under active deployment. Eventually, *Dungeons & Replicants* [*F*4] outlines the advantages of incorporating atomic behavioral data of a whole player population instead of heuristic or optimal agents when applied to exhaustive balancing simulations.

> Under the assumption that all classes of the utilized MMORPG should be equally viable in one-on-one situations, significant performance differences could be detected (and regulated). Above that, the presented mechanism was able to confirm the probably intended design of superiority relationships between classes and was able to inform game development with quantitative insights about the viability of classes when actually played with individually believable behavior.

Uniting the particular insights regarding **Q1-Q4**, the overall contribution of this dissertation can be consolidated, answering the overarching research question **"How can generative player modeling be realized in order to substitute individual human-like decision making in a representative, fair and convincing manner?"** Using atomic state-action records that pinpoint player decisions in generalizable game states, individual behavior can be sufficiently represented, rendering deep learning methods such as MLPs as viable modeling and generation techniques that can produce believable substitutions, challenging competition and representative decision making at eye level of individual proficiency. This could be consistently proven in multiple games, genres and field studies, while remaining areas and limitations are discussed in Chapter 5.

Theoretical Contribution

Parallel to the specific research questions that investigate the proposed approach, this dissertation provides meaningful insights for the broader fields of game design, game development, game user research and game AI. The most important theoretical implications are enumerated in the following.

AI techniques have to ensure usability in order to be recognized from the industry. Scientific research in video game AI has never been as well investigated as in this day and age – yet, the industry shows itself very conservative in employing novel approaches from academia. Reasons for this rather cautious utilization are extracted from interviews with industrial professionals in *The Case for Usable AI* [S3], which immediately lead to design guidelines that novel AI methods should follow: In order to consider these as usable, they must not harm but explicitly add to the **plausibility/believability** of NPCs; the techniques have to be **easy to implement**, debug and adjust; they should not increase the game's **computational performance** requirements significantly and the added value of **player experience** should be proven. To advance the integration of game AI and development of academia and industry, upcoming scientific submissions should fulfill and refer to these criteria.

Games user research should necessitate ecological validity. Insights from a large-scale examination about player behavior in GWAPs have shown a drastic disparity between field studies as opposed to laboratory studies [A5]. The latter can be highly impacted by confounding factors such as the experimenter or novelty bias, the Hawthorne-effect or inaccurate target groups that can lead to overinterpretations of the impact of novel approaches or techniques. In contrast, ecologically valid field studies are able to observe the outcomes of the actual target group within real-world environments and contexts. This could be demonstrated multiple times within this thesis by evaluations within newly released and publicly accessible games [F2, S2] or by extending commercially successful games with already existing player communities with novel content [F1, F3]. As preference and proficiency distributions in video games and genres are highly multifarious and likely to be biased by traditional laboratory subject sampling, games user research should stress the importance of doing evaluations in the field, especially as there are more opportunities of altering/modding or publishing games than ever before.

Individual believability is the key for convincing behavior. The field of believable agents is well-researched and active, yet most of the advancements focus on either extracting and modeling **general** player behavior or engineer universal approaches that approximate **general** human-likeness. While these are in fact able to appear as human to other players, they lack the ability to appear as **individual** human players. While this might be sufficient for some scenarios, **individual** representations come closer to the behavior of a population [$F4$] and provide opportunities for even more application cases, as they are able to approximate individual proficiency [$F1, F3, F4, S2$], substitute specific players [$F2$] and give information about particular differences between players [$F1$]. This opens up opportunities for games user research and game AI in modeling decisions on every conceivable abstraction, e.g. atomic behavior actions, movement patterns or high-level decisions.

Player behavior mainly consists of individual strategies and preferred ways to deviate from these. As the expert interview of the AAA-MMORPG *Aion* highlights [$F3$], players in general have an individual pivotal strategy planned before executing it and only deviate from this when forced by outside influences (e.g., action of other players or NPCs). In the case of MMORPGs and fighting games, this is reflected in the *main rotation*, i.e. the loop of preferred action sequences, and situational *responsive decisions*. These can be recorded and utilized in order to describe, model and/or generate individual player behavior [$F1, F2, F3, F4, S2$]. While this thesis is only to provide evidence for these specific cases, it is likely that these patterns transfer to other games and genres, e.g. the building and unit choices of RTS games or movement and aim patterns within FPS.

Atomic behavior is mappable to higher-level variables. Outcomes of the large-scale analyis of behavior throughout a player population in *Dungeons & Replicants* [$F4$] suggest that representations of atomic behavior implicitly contain information of higher-level variables such as the in-game proficiency (as estimated by efficiency and effectiveness measures) of the recorded player (cf. Figure 4.1). This does not only open up possibilities of comparing the proficiency of different players (and thus might enhance match-making procedures or could provide personalized challenges where players could face representations of equally skilled players), but inspires to investigate the connection of atomic behavior to various high-level parameters of interest. In a broader context, this method could be used to predict engagement, frustration, commitment or motivation (and thus would diminish the need of often immersion- or flow-harming questionnaires), estimate personality types or traits (and thus would open immediate opportunities to adjust game mechanics [$A3$]) or evaluate the seriousness of displayed behavior (and thus could augment the selection of useful as against malicious behavior for GWAPs [$A2$]) – only by interpreting atomic decision making.

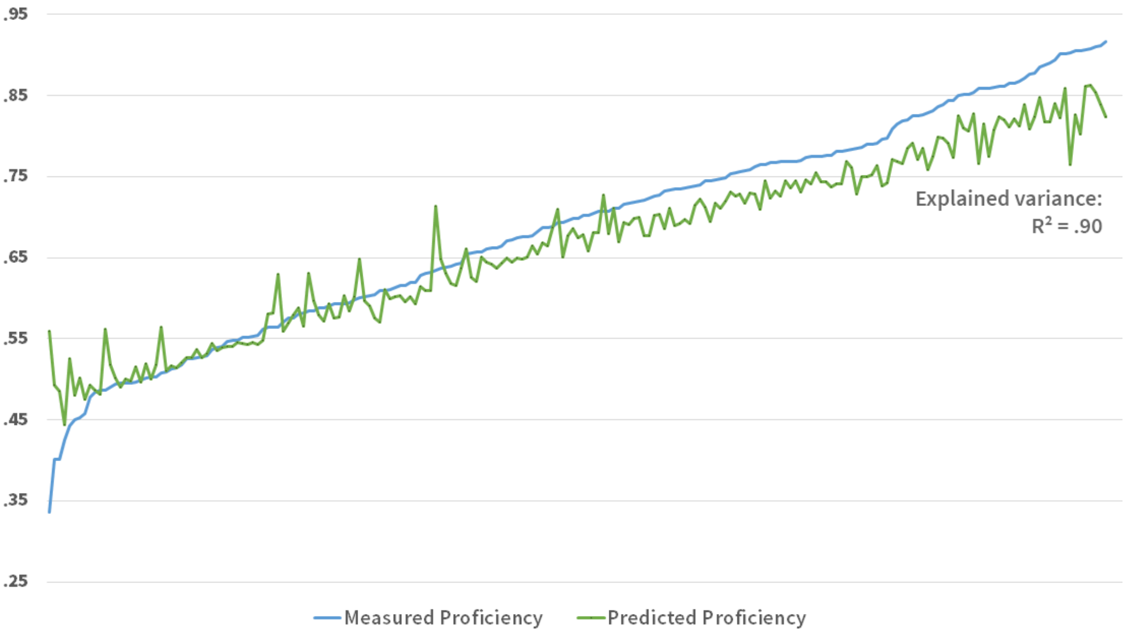


Figure 4.1: Measured proficiency values (blue) versus predicted proficiency (green), using a $1572x1572^4x1$ DPBM network that mapped game state variables and current and preceding actions to a single proficiency target value (ranging from 0 = worst to 1 = optimal proficiency).

5 Limitations & Future Work

With the overall research question as well as the sub-questions answered, certain limitations apply that facilitate future work from which some will we discussed in the following chapter. First of all, DPBM was applied to application fields that would benefit from generative player modeling within only a limited number of games and genres. While it could be argued that this technique is suitable for (MMO)RPGs with skill combat systems (as demonstrated by *Lineage II* [*F*1] and *AION* [*F*3]), point-and-click adventures (as shown in *ICARUS* [*S*1]) and arguably viable for Fighting games (as represented by *Korona:Nemesis* [*S*2, *F*2]), no evidence could be accumulated so far that justifies the installation in other genres. Thus, further empirical research would need to be executed that considers the environment of games representative for genres with considerably fuzzier decision making, such as FPS (where this mostly appears in making movement decisions and developing aiming and reaction proficiency), RTS (where vast global game states have to be considered that individual players are differently able to perceive) or genres with no clearly defined initial and goal state (such as simulation or sandbox games). Even with the successful implementation of DPBM in the aforementioned games, the previous game state representation only considers the situation between the individual player and one target, disregarding more complex scenarios with multiple opponents or allied players. Above that, the long-term application of the adaptive *AION* dungeon ([*F*3]) indicated that the DPBM opponent differed from its originator in several dimensions apart from decision making, such as reaction times or aiming precision. Most importantly, this version of DPBM is restricted to incorporating local movement information and neglects global movement patterns for now. While for these mentioned dimensions no trivial measures are available (as opposed to prediction accuracy for decision making), future evaluations will utilize an updated DPBM module to extract the impact of these factors on the overall individual believability (cf

Figure 5.1).

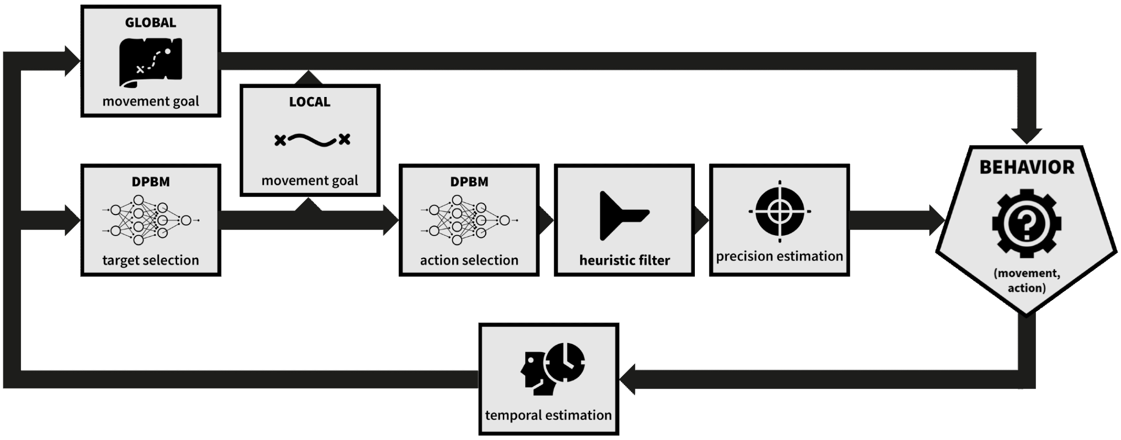


Figure 5.1: Behavior flowchart of an agent employing the succeeding DPBM module version.

With respect to this updated version of DPBM, **global movement** considers the underlying spatial goal, independent of a current target. Depending of the game, genre and situation, this can likely be broken down into two- or three-dimensional preference distributions over in-game maps which can be clustered into points of interest (Ahmad et al., 2019), transformed into semantic trajectories (Schertler et al., 2019) or utilized for pathfinding that extends traditional distance minimization with the minimal deviation of individual preference points. This will be joined with **local** movement behavior that informs mainly about relative distance preferences to a current target, so that approaching or fleeing behavior is incorporated depending on the respective situation. In situations that include multiple (allied or hostile) characters, a supplementary **target selection** module will be requested before actual action selection, to model mappings between game states and character focus (e.g. to recognize when a player would heal or support a team member or which opponent type or status is favored for attacks). After the core action selection, a **precision estimation** module is applied that approximates the player's precision proficiency (e.g. aiming in FPS or successful ground targeting in several other genres). Eventually, to accurately represent reaction and cognitive calculation times, the **temporal estimation** module calculates how much time the original player requires to execute a specific action in the respective situation and delays the execution of this action in DPBM opponents accordingly.

After thorough evaluations about the added value of perceived individual believability, DPBM can be utilized for application fields not yet covered by this dissertation. These include but are not limited to detecting bots in online games that carry out interminable and tedious tasks, revealing the use of cheats by recognizing behavior that is technically impossible or not by means of human capabilities, or yielding appropriate representations of opponents in asynchronous battles frequently used in mobile games (where challenging other players is often reduced to challenging a heuristic computer-controlled agent with the same equipment or setup). Additionally, further evaluations of DPBM for facilitating data aggregation in GWAPs follow, as realized in the online human computation games *Kitchen Clash* and *Elevator Empire*. Within these, players decisions, sequences and object preferences of everyday activity tasks are approximated to not only create highly adaptive and novel interaction mechanics, but also to aggregate a database of world knowledge and affordances for robotic learning (as explicated in [*A*4]).

Additional to the expansion onto further genres and application fields, DPBM offers capabilities for completely novel mechanisms and functionalities. When the mentioned limitations are overcome and the additional player characteristic approximators are both implemented and evaluated, it can be envisioned for e.g. maintaining perpetual presence of online game characters by substituting players by their individual surrogate when they go offline – or even enabling the possibility of playing alongside with their own alternative characters, driven by personalized behavior (in games where players can choose from multiple avatars). Future Work will evaluate how players will accept, experience and appraise these unprecedented situations.

6

Conclusion

Behavior of computer controlled agents in video games significantly differs from behavior displayed by human players. In many cases, this is desirable and introduced by design (as in elevating the players' perceived competence by impersonating a superhuman character), yet a multitude of application fields remain that would benefit from human-like behavior. While believability has been a major criterion within game AI and games user research that was approached by various algorithms for generative player modeling and artificial agents in general, modeling and replicating player behavior on an individual basis remained largely under-investigated. This dissertation introduced the design, development and evaluation of the Deep Player Behavior Modeling (DPBM) architecture that maps contextual situations (game states) to individual preference distributions (actions) of atomic decision making via machine learning techniques. After highlighting advances in scientific game AI, a comprehensive literature research represented the state of the art of player modeling approaches in related work, identified their most substantial features and classified them into descriptive research objective categories. Based on these, DPBM could be contextualized and the value of modeling individual and atomic decisions was justified by illustrating the advantages in the application fields of Dynamic Difficulty Adjustment (DDA), player substitution, automated game testing and serious games, in contrast to the recent advances of related work in these areas. Studies and insights of this thesis are presented in a twofold manner by first elaborating on the technical development through machine learning benchmarks, architecture progression and parameter examination and successively evaluating the constructed models in ecologically valid field studies of the aforementioned application fields with respect to player experience, motivation, commitment and believability. Answering the overall research agenda, DPBM was able to realize individually believable substitutions in a representative manner by reproducing atomic decision making on a similar level, convincing players that their own or a fellow player's behavior was achieved and eventually attaining an approximation of the original player's proficiency that can lead to an intrinsically motivating experience. These accomplishments are underlined by DPBM's industrial usability, since the approach was designed and revised to meet professional game developers' criteria: DPBM is able to learn and provide approximate player representations with reasonable temporal and sample size effort, allowing for instantaneous substitutions and rendering it **computationally performant**; its generic architecture that maps game states to action preference distributions is practicably transferable to most video game genres, ensuring **ease of implementation**; while quantitative as well as qualitative reports of multiple field studies highlight the distinct **believability** potential when employed for agent behavior. Limitations and opportunities are discussed subsequently, envisioning individual generative player modeling as a multi-dimensional approach that encompasses not only decision making in the sense of action selection, but also target choice, precision estimation, cognitive computation and reaction time while consolidating local and global movement factors. These will be further on applied to games within and beyond the genres examined in this thesis to strengthen the empirical results regarding DDA, player substitution, automated game testing as well as evaluating its capabilities for cheating or botting detection, asynchronous play and human computation augmentation.

7

Publications

The following publications constitute the most substantial amount of this thesis' contribution. To distinguish the personal contribution from the involvement of co-authors, each reference is shortly summarized and the personal contribution is indicated following the CRediT taxonomy [1].

7.1 | Foundational Publications

[*F*1] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. Towards deep player behavior models in mmorpgs. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '18, page 381–392, New York, NY, USA, 2018a. Association for Computing Machinery. doi: 10.1145/3242671.3242706

Contribution of this publication concerning this dissertation: In order to get a first dataset and evaluation about feasible machine learning techniques for DPBM, a study in the MMORPG *Lineage II* was conducted in which participants showed common single-player battle behavior. This was recorded and analyzed with the help of HMMs, DTs and MLPs and compared to the subjects' qualitative descriptions about their own decisions. This analysis laid ground for all succeeding publications integrating a form of DPBM.

Personal contribution to this work: Conceptualization, data curation, formal analysis, investigation, methodology, project administration, resources, software, validation, visualization and the major part of writing (90%).

[1] https://casrai.org/credit/

[*F*2] Johannes Pfau, Jan David Smeddinck, Ioannis Bikas, and Rainer Malaka. Bot or not? user perceptions of player substitution with deep player behavior models. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI ’20, page 1–10, New York, NY, USA, 2020b. Association for Computing Machinery. doi: 10.1145/3313831.3376223

Contribution of this publication concerning this dissertation: After the development of the online platform fighter *Korona:Nemesis*, it was officially launched on *Steam* and utilized for evaluating player substitution over the course of four weeks. Resulting in indistinguishability between DPBM opponents and human players (while notably heuristic opponents were detected often), this work constitutes the main factor in supplying evidence that deploying DPBM can overcome temporary online match disruptions.

Personal contribution to this work: Conceptualization, data curation, formal analysis, investigation, methodology, project administration, resources, major part of software, student supervision, validation, visualization and the major part of writing (85%).

[*F*3] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. Enemy within: Long-term motivation effects of deep player behavior models for dynamic difficulty adjustment. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI ’20, page 1–10, New York, NY, USA, 2020c. Association for Computing Machinery. doi: 10.1145/3313831.3376423

Contribution of this publication concerning this dissertation: Within the MMORPG *Aion*, I designed and developed the adaptive instance dungeon *Eternal Challenge* (EC) consisting of both traditional *rubber-banding* DDA as well as DPBM opponents. EC was published during a regular update on a private *Aion* server and evaluated across a study period of four weeks. Outcomes highlight the entertainment factor of encountering oneself represented by an DPBM opponent which even outclassed traditional DDA, confirm that players recognize their own behavior in the model and showcase the consistent motivational potential over long-term.

Personal contribution to this work: Conceptualization, data curation, formal analysis, investigation, methodology, project administration, resources, major part of software, validation, visualization and the major part of writing (90%).

[*F*4] Johannes Pfau, Antonios Liapis, Georg Volkmar, Georgios Yannakakis, and Rainer Malaka. Dungeons & replicants: Automated game balancing via deep player behavior modeling. In *2020 IEEE Conference on Games (CoG)*, pages 431–438. IEEE, 2020a. doi: 10.1109/CoG47356.2020.9231958

Contribution of this publication concerning this dissertation: Using an extended dataset of [*F*3] from the MMORPG *Aion*, a comprehensive set of DPBM agents could be established that represented the population of players. This served for two evaluations that were able to detect balance discrepancies between in-game classes with respect to fighting heuristic encounters as well as superiority relationships in the case of facing each other. Results approve the use of DPBM to incorporate the proficiency distribution of a player population instead of heuristic or generalized agents.

Personal contribution to this work: Conceptualization, data curation, formal analysis, investigation, methodology, project administration, resources, software, validation, visualization and the major part of writing (90%).

7.2 | Supportive Publications

[S1] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. Automated game testing with icarus: Intelligent completion of adventure riddles via unsupervised solving. In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '17 Extended Abstracts, page 153–164, New York, NY, USA, 2017. Association for Computing Machinery. doi: 10.1145/3130859.3131439
Contribution of this publication concerning this dissertation: The design and development of *ICARUS* aimed to accelerate industrial playtesting situated with *Daedalic Entertainment* and relief human testing procedures. While reinforcement learning was able to generically learn and play these adventure games, it could be even improved later by the incorporation of DPBM to resemble human playthroughs more closely.
Personal contribution to this work: Conceptualization, data curation, formal analysis, investigation, methodology, project administration, resources, software, validation, visualization and the major part of writing (90%).

[S2] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. Deep player behavior models: Evaluating a novel take on dynamic difficulty adjustment. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI EA '19, page 1–6, New York, NY, USA, 2019b. Association for Computing Machinery. doi: 10.1145/3290607.3312899
Contribution of this publication concerning this dissertation: To initially test the motivational potential of DPBM-driven opponents, a beta version of the elemental platform fighter *Korona:Nemesis* was released and evaluated. This field study was able to show the high intrinsic motivation of participants when facing DPBM in comparison to traditional game opponents with heuristic decision making.
Personal contribution to this work: Conceptualization, data curation, formal analysis, investigation, methodology, project administration, resources, software, validation, visualization and the major part of writing (90%).

[*S*3] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. The case for usable ai: What industry professionals make of academic ai in video games. In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '20, page 330–334, New York, NY, USA, 2020d. Association for Computing Machinery. doi: 10.1145/3383668.3419905

Contribution of this publication concerning this dissertation: This paper investigates the requirements of industrial video game companies in order to establish design guidelines for game AI, including the proposed methods of this thesis. To be applicable and usable for commercial development, AI techniques have to follow principles such as plausibility/believability, computational performance frugality, ease of implementation and evidenced increase of player experience. Based on these criteria, DPBM was designed, benchmarked and evaluated.

Personal contribution to this work: Conceptualization, data curation, formal analysis, investigation, methodology, project administration, resources, validation and the major part of writing (90%).

7.3 | Additional Related Publications

[*A*1] Johannes Pfau, Jan David Smeddinck, Georg Volkmar, Nina Wenig, and Rainer Malaka. Do you think this is a game? In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI EA '18, page 1–6, New York, NY, USA, 2018b. Association for Computing Machinery. doi: 10.1145/3170427.3188651
Contribution of this publication concerning this dissertation: This work assessed intrinsic motivation and flow between different degrees of gamification within the same serious game background. Insights about motivational research influenced further studies of DPBM regarding measurement, methodology and underlying theory.
Personal contribution to this work: Part of conceptualization, data curation, formal analysis, major part of investigation, methodology, part of project administration, resources, software, validation, visualization and part of writing (70%).

[*A*2] Johannes Pfau and Rainer Malaka. Can you rely on human computation? a large-scale analysis of disruptive behavior in games with a purpose. In *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*, CHI PLAY '19 Extended Abstracts, page 605–610, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3341215.3356297
Contribution of this publication concerning this dissertation: This work assessed quality of field studies where the appearance of a scientific purpose is concealed. Insights about credibility, potential biases and ecological validity influenced further studies of DPBM that all were held in online settings without explicating the academic background.
Personal contribution to this work: Conceptualization, data curation, formal analysis, investigation, methodology, project administration, resources, software, validation, visualization and writing (100%).

[*A*3] Georg Volkmar, Johannes Pfau, Rudolf Teise, and Rainer Malaka. Player types and achievements – using adaptive game design to foster intrinsic motivation. In *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*, CHI PLAY '19 Extended Abstracts, page 747–754, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3341215.3356278
Contribution of this publication concerning this dissertation: This work assessed intrinsic motivation and appraisal of game mechanics that adapt to individual players'

personalities. Insights about shaping individual challenges implicitly and general motivational research influenced further studies of DPBM regarding measurement, methodology and underlying theory.
Personal contribution to this work: Part of conceptualization, part of data curation, formal analysis, methodology, part of project administration, part of student supervision and part of writing (40%).

[*A*4] Johannes Pfau, Robert Porzel, Mihai Pomarlan, Vanja Sophie Cangalovic, Supara Grudpan, Sebastian Höffner, John Bateman, and Rainer Malaka. Give meanings to robots with kitchen clash: A vr human computation serious game for world knowledge accumulation. In *Joint International Conference on Entertainment Computing and Serious Games*, pages 85–96. Springer, 2019a
Contribution of this publication concerning this dissertation: This work assessed intrinsic motivation and efficiency of a serious game for robotic learning. Insights about motivational research influenced further studies of DPBM regarding measurement, methodology and underlying theory. Aggregated data could be utilized to study the integration of DPBM into robotic learning via human computation.
Personal contribution to this work: Part of conceptualization, data curation, formal analysis, part of investigation, methodology, part of project administration, resources, software, student supervision, validation, visualization and major part of writing (75%).

[*A*5] Johannes Pfau and Rainer Malaka. We asked 100 people: How would you train our robot? In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '20, page 335–339, New York, NY, USA, 2020. Association for Computing Machinery. doi: 10.1145/3383668.3419864
Contribution of this publication concerning this dissertation: This work assessed player experience and feasibility of a serious game for robotic learning and investigated automated methods of classifying unwanted behavior. Insights about motivational research influenced further studies of DPBM regarding measurement, methodology and underlying theory.
Personal contribution to this work: Conceptualization, data curation, formal analysis, part of investigation, methodology, part of project administration, resources, software, student supervision, validation, visualization and major part of writing (90%).

[*A*6] Mehrdad Bahrini, Nima Zargham, Johannes Pfau, Stella Lemke, Karsten Sohr, and Rainer Malaka. Enhancing game-based learning through infographics in the context of smart home security. In *Joint International Conference on Entertainment Computing and Serious Games*. Springer, 2020b
Contribution of this publication concerning this dissertation: This work assessed intrinsic motivation and learning effects of serious games with different visual representation of knowledge. Insights about motivational research influenced further studies of DPBM regarding measurement, methodology and underlying theory.
Personal contribution to this work: Part of conceptualization, part of formal analysis and minor part of writing (10%).

[*A*7] Mehrdad Bahrini, Nima Zargham, Johannes Pfau, Stella Lemke, Karsten Sohr, and Rainer Malaka. Good vs. evil: Investigating the effect of game premise in a smart home security educational game. In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '20, page 182–187, New York, NY, USA, 2020a. Association for Computing Machinery. doi: 10.1145/3383668.3419887
Contribution of this publication concerning this dissertation: This work assessed intrinsic motivation and learning effects of serious games with different premises. Insights about motivational research influenced further studies of DPBM regarding measurement, methodology and underlying theory.
Personal contribution to this work: Part of conceptualization, part of formal analysis and minor part of writing (10%).

[*A*8] Nima Zargham, Johannes Pfau, Tobias Schnackenberg, and Rainer Malaka. Handle with care: Exploring recognition error handling methodologies for speech-based systems. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. Under review.* ACM, 2020
Contribution of this publication concerning this dissertation: This work assessed appraisal and intrinsic motivation of participants using different versions of error handling for voice-controlled systems. Insights about user experience, usability and perception of interaction with AI agents influenced further research of DPBM.
Personal contribution to this work: Part of conceptualization, part of formal analysis and part of writing (20%).

[*A*9] Robert Porzel, Vanja Cangalovic, Mihai Pomarlan, Sebastian Höffner, Johannes Pfau, John Bateman, and Rainer Malaka. Understanding instructions all the way: A simulation-based approach. In *Proceedings of the 28th International Conference on Computational Linguistics. Under Review*, 2020
Contribution of this publication concerning this dissertation: This work introduces an architecture for processing natural language instructions in the domain of everyday activities, influencing related work on serious games and human computation.
Personal contribution to this work: Part of conceptualization and part of writing (10%).

Towards Deep Player Behavior Models in MMORPGs

Johannes Pfau
Digital Media Lab, TZI
University of Bremen
Bremen, Germany
jpfau@tzi.de

Jan David Smeddinck
Open Lab, School of Comp.
Newcastle University
Newcastle upon Tyne, UK
jan.smeddinck@ncl.ac.uk

Rainer Malaka
Digital Media Lab, TZI
University of Bremen
Bremen, Germany
malaka@tzi.de

ABSTRACT

Due to a steady increase in popularity, player demands for video game content are growing to an extent at which consistency and novelty in challenges are hard to attain. Problems in balancing and error-coping accumulate. To tackle these challenges, we introduce *deep player behavior models*, applying machine learning techniques to individual, atomic decision-making strategies. We discuss their potential application in personalized challenges, autonomous game testing, human agent substitution, and online crime detection. Results from a pilot study that was carried out with the *massively multiplayer online role-playing game Lineage II* depict a benchmark between *hidden markov models*, *decision trees*, and *deep learning*. Data analysis and individual reports indicate that *deep learning* can be employed to provide adequate models of individual player behavior with high accuracy for predicting skill-use and a high correlation in recreating strategies from previously recorded data.

CCS Concepts

•**Information systems** → **Massively multiplayer online games;** •**Human-centered computing** → *User models;* •**Computing methodologies** → *Machine learning approaches;*

Author Keywords

Neural networks; deep learning; HMM; decision trees; games; player modeling; personalization; game testing; adaptive agents; dynamic difficulty adjustment

INTRODUCTION

Video game production and maintenance, especially for flagship productions, is reaching the limits even of what large companies can sustain. Following the demand of players, games grow more complex in terms of content and mechanics, where the action spaces become nearly endless, greatly increasing the number of things that could potentially go wrong. This includes players facing unbalanced challenges, software execution or gameplay bugs that go undetected, connectivity issues with large-scale systems, and cheating or other unethical behavior. We approach the closing of multiple unsolved gaps in these areas of concern for game research and development based on an uncommon building block: *deep player behavior modeling* (DPBM). We discuss the potential of DPBM with regard to the challenges indicated above. To establish apt representation techniques we also explore the potential of different machine learning techniques for player modeling in *massively multiplayer online role-playing games* (MMORPGs) and implement a pilot study which provides a first data set and enables the comparison between selected models. We hypothesize that different advantages can be attained from *Hidden Markov Models* (HMMs), *decision trees* (DTs) and *deep learning* (DL), in terms of analytic capabilities and prediction power. After outlining the concept of employing user modeling with machine learning for approaching challenges in game research and development, we present the selected techniques and illustrate the results of an exploratory study that was carried out with the established MMORPG *Lineage II*. Deep learning appears most adequate in terms of prediction accuracy and behavior representation similarity, whereas HMMs and DTs offer useful visualization and analysis features. These models also constitute the basis for ongoing subsequent future work focusing on the evaluation of user experience in different game modes.

Through this paper, we contribute a general discussion of potential application fields of DPBM and machine learning in the context of digital games, provide an overview of the state of the art in research and industry, and point out distinct advantages of player behavior models. Additionally, our exploratory study exemplifies an early working utilization, highlighting advantages and disadvantages of the different models.

BACKGROUND

In related work, player behavior modeling has been approached mostly with the goal of facilitating *dynamic difficulty adjustments* (DDA) [1, 8, 20, 43, 50, 58]. Further application areas that have been discussed are the modeling of behavior impressions for informing game development [6, 9, 26] and the reproduction of atomic tasks [12, 47]). After briefly highlighting MMORPGs as an especially fitting class of games for the application of DPBM with machine learning in the next section, the subsequent sections introduce general categories that encompass the application areas of these isolated reports and discuss the potential of individual behavior models for tackling common challenges in games research and development **A-D**.

The Case for Player Modeling in MMORPGs

MMORPGs typically encompass several aspects that highlight the potential benefits of the application of player modeling. The vast amount of data that each individual player is generating constantly along with the immediate opportunity to compare it to the global behavior data of a big community provides a rich basis for powerful but data-demanding machine learning algorithms. Moreover, the player behavior in MMORPGs can most often be broken down into *movement behavior* and *action selection* (skill usage). Each skill in this context is unique and discrete, which allows for less noisy behavioral state categorization compared to other popular video game genres, such as *first-person shooters* (FPS) or *real-time strategy* (RTS) games, where behavioral data quickly gets noisy [53]). Also, the continuously required internet connection simplifies centralizing and outsourcing the computational effort (e.g. through cloud computing). In return, player behavior models can enrich MMORPGs in various ways, such as increasing novelty and the prevalence of interesting challenges (see **A**), human tester relief and predicting the effects of game changes (see **B**), preventing game or match disruptions (see **C**), and preventing unfair or unethical player behavior which can not only harm the player community but can also cause financial losses for players and companies (see **D**). These aspects are further discussed in the following sections to clarify how these potential benefits of DPBM can come to play in (massively) online multiplayer games.

A. Personalized Challenges

Due to the complexity and scale of the game environments and interactions it can be difficult to consistently present motivating, well-perceived and evenly balanced challenges in player versus environment (PvE) modes of MMORPGs. Challenges frequently task players with defeating a large number of enemies at once (all substantially weaker than the player character) or with approaching powerful boss enemies (substantially stronger than the player characters) in a group or crowd of players. An even, one-to-one challenge is generally only found in player versus player (PvP) modes, when players may face other players on approximately the same skill level. To be able to compete in these matches, players have to constantly improve their skills and adapt to the specific situation and opponent, which in the end leads to a less repetitive gameplay providing more motivation for long-term commitment. As indicated by Fuster et al. [16], PvP-focused players spend significantly more hours per week on playing MMORPGs than PvE-focused players. PvE lacks this kind of a continuously changing challenge since non-human opponents are almost always constructed by *"simple rule-based finite and fuzzy-state machines for nearly all their AI needs"* [55], which quickly become predictable as there is no way to tailor their behavioral skill level to individual players, or to vary their behavior in a complex yet not overly random manner.

Thus, the guiding idea behind an ongoing, adaptive challenge is the maximization of *interestingness* (as introduced by Yannakakis et al. [56]) through adequate player modeling [5, 57]. The models proposed in this paper are able to represent player behavior individually and therefore result in the behavior of an agent on approximately the same skill level as the original human player. Given such abilities, challenging "oneself" can present continuous and powerful DDA, since players have to adapt and overcome their own behavior in order to be less predictable. In this light, genuinely balanced challenges can be provided to players on a generative basis. Moreover, if the skill level of a player can be quantitatively assessed by means of player modeling, novel PvE modes are conceivable that confront the player with ever-changing enemies originating from a potentially large set of human players with adequate skill levels.

B. Autonomous Game Testing

Automatic simulations of video game play have proven to be usable in situations where human testing is too tedious or not exhaustive enough for the purpose of finding bugs and glitches [4, 15, 38, 45], parameter tuning [60], and assuring solvability [42]. Based on the insights and the potential of our previous work on a tool for completing and debugging adventure games [35]), we want to further extend the possibilities of autonomous game testing.

For developers, one of the most difficult and time-consuming phases of the game design process [22] is the balancing of different character classes. Following the definition of Sirlin [44], a multiplayer game is *"balanced if a reasonably large number of options available to the player are viable"* (where viability sets the requirement of having many meaningful choices throughout a game), while *"players of equal skill should have an equal chance at winning"*. Together with frequently desired asymmetrical character configuration possibilities this inherently leads to combinatorial explosions, which can become hazardous for the enjoyability of the game and the satisfaction of its players [39]. Even worse, balancing issues most of the time *"only become apparent after many months of play"* [19] and the trouble with these issues (in comparison to straightforward fixable bugs, glitches and solvability aspects) is that they do not only appear during the launch of a newly published game. Balancing is an ongoing and repeating task that is heavily influenced by the perceptions of the player community (*"after each patch, often the discussion begins again, factoring in new balancing or abilities for each class"* [26]). In the games industry this is most often approached through long-term expert analysis, excessive human play-testing, and persistent debates with the community.

Academic work presents approaches to tackling the issue of balancing different setups by simulation [3, 22] or genetic algorithms [25, 28, 29], yet without incorporating additional information about situated player behavior. Individual player models have the potential to unite automatic simulation methods with behavioral information. This gives developers the opportunity to receive practically immediate insights on a) which player strategies are popular, dominant and/or may require rework, b) how parameter tuning will likely alter the outcome of strategies before presenting it to the community and c) how to automatically balance game mechanics after large-scale permutations of classes, setups, parameters and behavior – in all stages of development.

C. Human Player Substitution

Most multiplayer games throughout the popular genres have to handle disconnected players (or those who stop providing input for a longer time). In some cases, for balancing or game-experiential reasons, these lost players are replaced by computer-controlled agents (e.g., Left 4 Dead [52] (an FPS), Heroes of the storm [13] (a multiplayer online battle arena game), Super Smash Bros. 4 [2] (a Beat 'em up), Mario Kart 8 [10] (a racing game), Civilization V [17] (a turn-based strategy game), Company of Heroes 2 [14] (an RTS), or Rocket League [37] (a sports game)). However, such substitution is frequently criticized, since the replacing agent is usually not able to compete with human players. On the other hand, developers cannot allow the deployment of computer-controlled agents that are clearly stronger than the replaced human player due to the obvious potential of abuse. Thus, the only satisfying replacement would be an agent that acts very much like the original human player and performs neither significantly better nor worse than her (at least until the original player returns). After the evaluation of application area **A** (approaching player behavior modeling for generative adaptive challenges), the computed models presented in this work will be used in order to assess whether computer-controlled agents perform on appropriate skill levels and present human-like behavior, controlling whether they a) even strike the attention of other players in the match and b) whether they can serve to replace lost players on an adequate level (i.e. showing a consistently comparable performance). At least one of the mentioned criteria should be accomplished in order to achieve temporary match disruption prevention.

Another possible application area is presented through the upcoming growth of asynchronous games [41], especially in the current age of mobile gaming: In many instances of this type of games players can challenge other players without the need of actually playing at the exact same time as the opponent. This allows the fast-paced and very situation-dependent world of mobile gaming to feign battles of various genres like strategy games, turn-based games, but also even real-time role-playing game battles so they can take place whenever it suits a player (cf. e.g. Clash of Clans [46], Pokeland Legends [51], Star Wars: Commander [21], Goddess: Primal Chaos [24]). As in the previously mentioned case, human opponents are represented by computer-controlled agents with the same character setup, equipment and/or further attributes, but lack individual decision making/behavior. This again leads to a misrepresentation of the actual human opponent's skill, a potentially unfair advantage for the attacking/challenging player, and consequently causes high, skill-independent fluctuations in leaderboards. Player behavior models as described in our approach could be integrated in asynchronous games or game modes to further extend the opportunities of this type of games and to give both the attacker a more appropriate challenge and the defender a better and fairer representation of herself in her absence.

D. Cheating and Botting Detection

One of the major classification paradigms in which player modeling has successfully been used before is the detection of unwanted automated software (botting) in online games [7, 18, 23, 31, 33, 48]. Malicious bot software has no or little access to the actual game variables and objects and is thus usually based on heuristic or predefined decision making. Above that, botting is used mostly in worthwhile areas and thus typically makes use of fixed paths, leading to rigid movement behavior. As such, differences between the classes of bot and human player can be identified quite accurately given the aforementioned techniques. Another problem in online games is the act of identity theft, where criminals gain unwanted access to user accounts. Existing approaches tackle the issue through different means of automatic detection [34, 54]. We argue that these approaches can be extended by employing in-depth player behavior models for the classification between real human account / character owners and imposters. Finally, competitive games are always prone to cheating or hacking. In such cases, DPBM can be employed to improve play-style analytics in order to classify suspicious or technically impossible behavior.

Lastly, beyond above major application areas, these models have the potential to aid in classifying player roles [11], to assess the player's experience [27] based on his behavior, and the live application of DPBM also opens the door for developers to enhance player experiences with completely novel game mechanics in existing or potentially newly created game modes.

APPROACH

Following the definition by Yannakakis et al. [56], a game is only interesting when it "is neither too hard nor too easy", shows "diversity in [opponents'] behavior over the games" and "[opponents'] behavior is aggressive rather than static". That means that a) strictly optimal behavior is just as little interesting as conventional, predictable heuristic non-player character (NPC) behavior, b) opponents should evolve over time in order to constitute a dynamic challenge and c) players should experience a tension similar to the confrontation of a human opponent. To these ends, we aim for a model that displays increasing player behavior fitness when presented with increasing amounts of data showing similar behavior.

Study

In order to assess which models and methods are sufficiently expressive and accurate, a pilot study has been conducted, resulting in a viable initial set of behavioral data. We chose to gather this data set in isolated play sessions in order to control the setting, collect verbal reports from participants, and to reduce confounding variables in comparison to noisy "in the wild" data. The convenient subject study participants were asked to maximize their score by defeating the highest possible number of enemies within 30 minutes in an open-world PvE game mode of the popular MMORPG *Lineage II* [32]. To model players with and without previous experience in the game, recruitment took place on a private server of this game and via email. As exemplary applications for investigating the applicability and performance of the different modeling approaches, we phrase the following assumptions for player behavior.

Players:

- prefer to use skills in *rotations*, i.e. sequentially
- use certain skills in certain situations , e.g.,
  - *initial* skills for each enemy (depending on whether the intention of the enemy is idling or attacking)
  - skills only viable when own/target HP is low/high
  - skills only viable when distance to the target is low/high
- choose different strategies in different game modes and against different enemies/classes

Thus, in order to allow the models to incorporate these cues, game states include (but are not limited to) the variables contained in these assumptions, such as previously used skill(s), health point (HP) conditions and distance between player and non-player character(s) (cf. Table 1), while game actions include all used skills and movement of the player.

Measures

An initial questionnaire asked for demographics and video game experience. During the task, we recorded movement data as continuous paths and skill usage by logging the most important character and target state information. After the play session, the participants were asked to complete a questionnaire containing *Player Experience of Need Satisfaction* (PENS) [40] and *Intrinsic Motivation Inventory* (IMI) [36] items in order to gather prediction training and validation data for later use. The participants were also asked to describe the strategies they employed when encountering enemies in detail while observing a replay of their player behavior. Skill icons and descriptions were shown during the replay for easier reference. The participants were asked to explain common skill rotations and rare / notably situated skill usages specifically. Lastly, the participants were asked to discuss their perception of the interestingness of computer controlled enemies in MMORPGs and to compare them to the experience of encountering a human player (in PvP). All data were evaluated in a pseudonymized fashion and stored in an encrypted file container.

Procedure

The study was executed in an online setting. Subjects were asked to download the game client in advance and met the experimenter on a TeamSpeak3 server, enabling voice communication throughout each session. Following informed consent and the pre-study questionnaire, participants chose between three different classes (Warrior, Archer or Wizard), were able to customize their skill configuration and test it on non-responding training dummy enemies without temporal restrictions (typically lasting 5-10 minutes). When the participant felt ready, the experimenter started the countdown of 30 minutes and teleported the player character to the treatment start location. In this place, a large number of common MMORPG enemies appeared that could be attacked in order to raise the participant's score. The score and the remaining time were displayed at all times. Upon death, the character was revived at the initial location. Throughout the whole in-game task no other player characters were present. After the countdown completed, the game shut down automatically and the remaining questionnaires (PENS, IMI) were presented. This setting is representative for many tasks that MMORPGs present. In this pilot study we chose to focus on the assessment of single-player behavior first in order to benchmark the respective models before we broadening the scope to more noisy and also socially dynamic multiplayer settings.

variable	value
time	08.02.2018 15:26:16
timeReuseAvailable	08.02.2018 15:26:22
skillID	10771
skillName	Multiple Arrow
casterID	268492397
casterName	TestArcher
casterClassID	162
casterClassName	Archer (Yul Sagittarius)
casterHPpercentage	100
locX	-11965
locY	237519
locZ	-3213
targetID	23355
targetName	Armor Beast
targetClassID	-1
targetClassName	NPC
targetHPpercentage	100
ai_intention	AI_INTENTION_ATTACK
distance	309.36
locXtarget	-11936
locYtarget	237827
locZtarget	-3227
zone	Hellbound (Study)
score	34

Table 1. Example database entry for *skillLogs*. Bold fields are used in the behavior model computation (DT, deep learning). *Italic fields* are used in movement analysis along with further detailed path data, while the remaining variables serve purposes in readability and visualization. *Zone* consists of location and game mode information. AI_intention describes the current aim of the target, most notably IDLE, ACTIVE, ATTACK or CAST.

Participants

In total, $N = 24$ subjects completed the task (87.5% male, 12.5% female, 22 to 28 years of age (M=25.2, SD=1.96), yielding a fair representation of MMORPG demographics [59]. All of them stated being active gamers with 11 to 25 (M=16, SD=3.96) years of previous video game experience and 20 to 55 (M=30.4, SD=12.4) hours spent on games per week, while 83% also indicated that they had played *Lineage II* before.

Results

The final score varied greatly among participants (122 to 434 defeated enemies, M=287.4, SD=92.6), with a significant performance difference between participants with and without previous experience in *Lineage II* ($p < 0.01$ with a Welch's t-test, Cohen's $d = 1.69$, $df = 22$). We found positive (Pearson) correlations between score and the surveyed PENS: in-game autonomy (r=0.37), presence (r=0.38) and IMI interest-enjoyment (r=0.48) sub-scales. Overall, the notable variance in their performance indicates that the participants have chosen different strategies to approach the enemies. The analysis and differentiation between strategies is further assessed in the next section. Regarding general performance as indicated by score, no significant differences due to sex, education or other demographic variables were found. When they were asked to discuss the interestingness of computer enemies in MMORPGs generally, subjects shared a common opinion, that NPCs are "predictable", "no real enemies", "not comparable to the experience of PvP", "often boring [...] without good AI" and that "the only fun comes from the rewards, not the battle itself".

Data analysis & Modeling

After testing several machine learning algorithms and fitting techniques, namely HMMs, DTs, deep learning, clustering, regression, splines, support vector machines (utilizing TensorFlow and DeepLearning4j) for their applicability to capture the individual behavior accurately, we selected to report on HMMs, DTs, and deep learning (DL), which bear distinct advantages in performance or visualization capabilities. All models were trained on a training set (80%) from the gathered study data (one model per player) and later used to predict action selection on a testing set (20%) from the corresponding player (see *Prediction Results*) and to classify behavior between players (see *Player Differentiation*). The prediction accuracy was compared between all models, with and without the incorporation of the heuristics that stem from former assumptions (see **Study**).

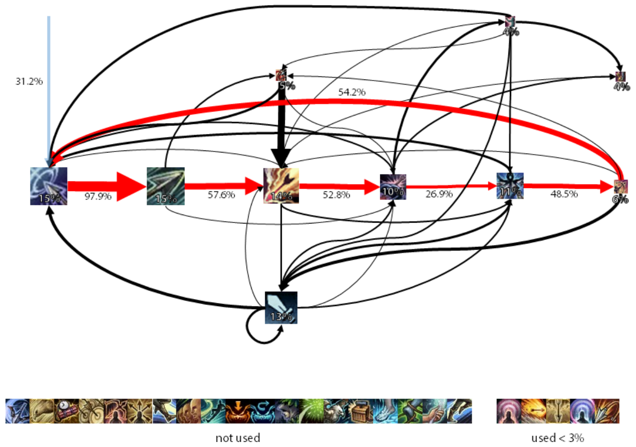


Figure 1. HMM skill transition graph of a single participant. White percentages and the skill icon sizes display the relative usage of the respective skill. The width of the transition arrows is proportional to the transition probability from one skill to another. The blue arrow shows the most likely skill to begin attacking each enemy, while red transition arrows depict the main rotation (transition probabilities are labeled black). Skills used in less than 3% of encounters are included in the calculations of the model but excluded from the visualization due to visibility reasons.

Hidden Markov Models

Following our first assumption (see **Study**), which was supported by similar reports in the post-test questionnaire, one major behavioral criterion is the adherence of individual skill rotations. Since HMMs shine in their capabilities of visualizing state sequences, we formulated the estimation of the main rotation as a markov chain with the respective previous skills as observable variables, while the complex behavior strategy stays hidden.

Figure 1 displays the HMM for a single player and demonstrates the intuitive illustration of the probability of the skill successors given a previously known state. The most used skill together with its most probable successors constitute the individual main rotation, which differs from player to player (cf. Figure 2).

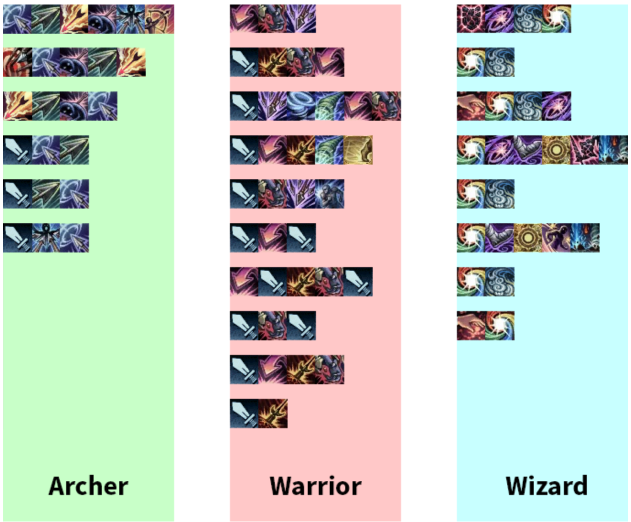


Figure 2. HMM-computed main rotations of all participants. Each line stands for the most likely rotation of a given player, from the first (most probable initial) skill across the following most likely successors. The highest transition probability of the last skill in line is the first skill again.

In order to increase the contextual integration capabilities of HMMs, we chose to extend the dimensionality of the original method by using 2nd order HMMs [49] at the cost of requiring more training data. We also integrated an *initial heuristic*, which assumes that players might prefer to attack enemies with certain skills initially (and thus, outside of their main rotation) and a *cooldown heuristic* which filters out idiosyncratic strategies that are not executable at the given point of time due to cooldown restrictions.

Decision Trees

While HMMs struggle with incorporating larger numbers of dimensions, DTs can break data down following the most discriminatory variables. This allows for pinning down decisive factors for skill usage accurately from a selection of many contextual game state factors that are potentially relevant. We included information about the enemy's intention (IDLE, ACTIVE, ATTACK or CAST), the previously used skill and binary choices whether the player's HP, the enemy's HP or the distance between them is above or below the respective mean of the current player's data (cf. Table 1, bold entries). Discriminativeness was calculated via Shannon entropy. As our outcomes show, this approach does not only yield a higher accuracy in predicting skill usage compared to HMMs, but it is even capable of "explaining" the intention of rarely used skills. By reversing the tree and collecting all paths ending in a particular skill leaf, the situational context of this skill can be illustrated.

For example, when asked for skills that were rarely used, one participant stated that he activated a skill ("Death Lord") whenever his HP dropped to a low level, in order to transfer some of the enemy's HP to his own. Reversing the tree returns "NOT HP above mean" (95.2% accuracy) as the top criterion for this skill, followed by "target HP above mean" (90.5%). The player's contextual usage of this skill is thus accurately described by "having low HP while the enemy has high HP". Furthermore, one player stated to use "Bow Strike" whenever an enemy gets too close, and thus knocking the enemy back - where the tree returned "NOT distance above mean" and "target HP above mean" as top criteria, explaining even more (since an approaching low HP target could be defeated quickly, but only approaching high HP targets are countered with the knock-back). One subject reported the usage of "Power Provoke" if – and only if – his HP are full and many enemies are around, since this skill taunts all of them to attack him, so that he can face all of them at the same time. The HP situation could be reflected, but the number of possible enemies to attack is a metric that was not logged, which should be considered in further research. Nevertheless, this process of reversing DTs produces a ranked set of meaningful variables in which particular skills are used and is capable of delivering clearly understandable insights to developers.

Deep Learning

Neural networks add further modeling performance since the learning process does not rely on manually defined discrimination criteria and all variables (again, cf. Table 1) contribute their real values instead of binary decisions, as is the case with decision trees. As a drawback, neither can the learned model be easily visualized nor can the process be reversed in order to describe situations in which particular skills are used. Above that, the computational and temporal effort of training and retrieving is considerably larger than for the former techniques. Nevertheless, the scalability of outputs beyond situations explicitly provided in the training data and high accuracy in prediction render deep learning a viable candidate for behavior generation. To establish general applicability we chose a multilayer perceptron with backpropagation and a logistic sigmoid activation function and trained one network for each participant, where input and output array sizes varied due to different numbers of skills used / available. Each network consisted of up to 38 input nodes and up to 34 output nodes (cf. Figure 3), while we ran a number of simulations for the best fit/effort ratio in terms of the number of hidden layers (1 to 5), nodes within (5 to 30) and training epochs (cf. Figure 2). Since skillIDs of previous skills are nominal and bear no meaning in their values, they had to be realized as individual input nodes. Target values were constituted by the use of particular skills given the situation defined from the input array. For the prediction afterwards, the computed output array is translated to a density function from which the guessed skill is picked probabilistically. Most fitting iterations did not improve significantly beyond 1000 training epochs, which were reached after about 7 minutes on a local i7-6700HQ CPU @2.60GHz (using a single core). Retrieval time from a trained model did not exceed 20 milliseconds.

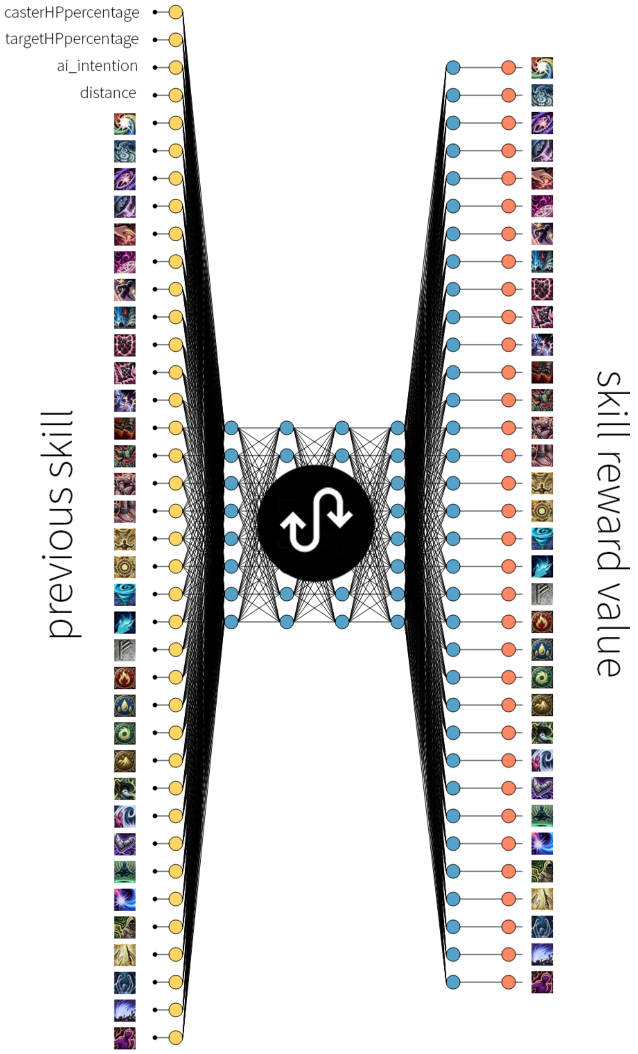


Figure 3. Example network for one participant. Real valued variables are mapped to the range from 0 to 1, previous skills are encoded as binary switches. Hidden layer and neuron count varied after optimizing for prediction accuracy.

Prediction Results

As shown in Figure 4, deep learning outshines our previous approaches with 55-97 % (M=71.4%, SD=13.2%) prediction accuracy across individual models. We did not compare the outcomes to complete random guessing, since the accuracy of random guessing in this high-dimensional action space would be <3 %. Rather, the baseline (BL) depicted stems from guessing with only the mere skill frequency probabilities of a given player, without further contextual information. HMMs succeed in extracting the most probable main rotations of the participants, but fail to explain the usage of rarely used skills. Second order HMMs (HMM²) yielded no significant difference in prediction compared to the former, while still inducing the cost of a considerably slower training curve. DTs

#	1	2	3	4	5	6	7	8	9	10
h	2	1	3	3	2	2	1	4	3	2
hn	22	5	15	12	3	13	8	5	17	12
acc	92	86	56	63	59	66	66	59	63	74

#	11	12	13	14	15	16	17	18	19	20
h	1	2	4	2	2	3	1	2	2	2
hn	6	8	14	7	6	12	17	21	17	9
acc	65	67	55	97	78	56	64	67	76	81

#	21	22	23	24
h	3	2	2	1
hn	8	11	7	12
acc	66	89	97	71

Table 2. Number of hidden layers (h) and nodes within (hn) used for each participant (#), resulting in the respective accuracy (acc) in %.

produce human-readable outcomes and facilitate attributing contextual factors to the situational usage of skills. However, they rely on manually defined criteria to accurately grasp borderline cases and introducing a large set of variables and/or decision criteria for these may harm the readability of reverse tree queries.

Among participants, prediction accuracy was significantly higher ($p < 0.01$ with Welch's t-tests, Cohen's $d > 1.03$ for all models) for experienced players compared to those who have not played *Lineage II* before (cf. Table 3). This indicates that the former stick more tightly to learned strategies and patterns whereas the latter are more eager in trying out different styles. Accordingly, this yields a slight correlation between score and prediction accuracy (HMM: r=0.19, DT: r=0.16, DL: r=0.21).

		BL	HMM	HMM²	DT	Deep Learning
exp	*M*	36.5%	48%	47.9%	61.7%	73.4%
	SD	18.5%	17.1%	17.0%	16%	13.6%
inexp	*M*	19.3%	29.5%	29.5%	49.5%	61.8%
	SD	5.3%	7%	7%	4.7%	5.1%

Table 3. Average prediction accuracies between players with (exp.) and without (inexp.) previous *Lineage II* experience.

Regarding the former assumptions, the *cooldown heuristic* (temporarily discarding candidate skills from prediction that are not usable by design) significantly increased ($p < 0.05$ with Welch's t-test) the accuracy in HMMs and DTs, where no difference in the case of deep learning networks was found, since they inherently incorporated this information. The *initial heuristic* increased the accuracy in some participants' cases, but decreased it in others, not leading to significant improvements.

Player Differentiation

Genereally speaking, the study participants used skills differently. E.g., while some focused on defeating single enemies as quickly as possible, others attempted to gather larger group of enemies in order to utilize skills that damage multiple targets.

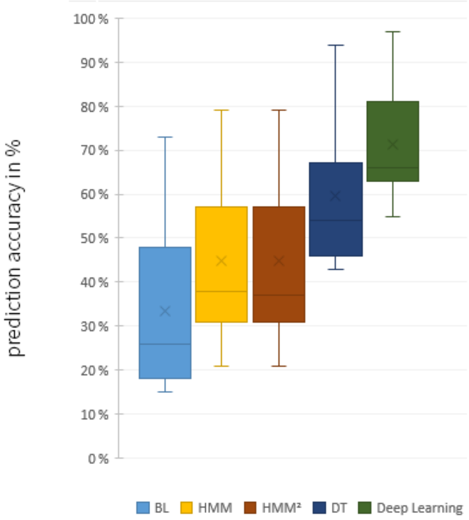
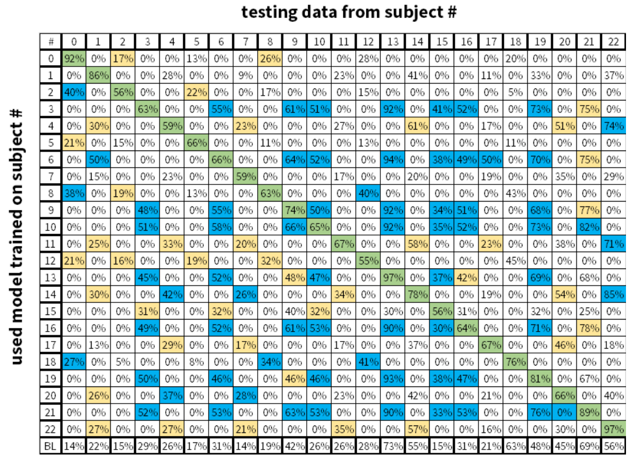


Figure 4. Prediction accuracies for BL, HMM, HMM², DT and DL. Heuristics are included in the respective model variant if they yielded higher accuracy.

Certain players made efficient use of time-limited reinforcement skills (buffs), weakening skills (debuffs) and/or approximated the theoretical optimal damage rotation, whereas others stuck to personal preferences or even a seemingly random selection. Since our models should not only be able to predict skills from the trained player, but also be usable for differentiation between them, we benchmarked the respective model on the data from all other players (cf. Figure 5). In most of the cases (82.4%), player behavior is different enough so that it can not be predicted accurately from another model. However, 13.2% of the time, models explain considerable portions of the behavior of another player. Importantly, this does not necessarily reveal similarity between two players, since the explained behavior might only be a subset of the other player's behavior. If we want to establish a similarity measure, which could be used in order to approach cheating or identity theft detection, we have to examine if the prediction accuracy is bidirectional - which is only the case in between subjects 3, 9, 10, 13 and 16.

Movement Analysis

We aimed to represent both fast-paced movement decisions as well as long-term movement plans. Engaging an enemy often evokes situational movement decisions depending on the individual strategy of the player (e.g., chasing an opponent or building up/maintaining a larger distance), therefore we included local movement decisions in the former presented skill usage model architecture by treating movement as a unique skill with distance and location parameters. Above that, players move according to their global intention [30] (e.g., reach certain points or areas), so decision making on a bigger scale has to be considered. We evaluated a number of fitting techniques and ended up with B-splines to approximate the overall movement behavior, compressing it to a smooth function (cf. Figure 6). The importance of global movement decision making will become more apparent in our follow-



testing data from subject #

used model trained on subject #

#	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
0	92%	0%	17%	0%	0%	13%	0%	0%	26%	0%	0%	0%	28%	0%	0%	0%	0%	0%	20%	0%	0%	0%	0%
1	0%	86%	0%	0%	28%	0%	0%	9%	0%	0%	0%	23%	0%	0%	41%	0%	0%	11%	0%	33%	0%	0%	37%
2	40%	0%	56%	0%	0%	22%	0%	0%	17%	0%	0%	0%	15%	0%	0%	0%	0%	0%	5%	0%	0%	0%	0%
3	0%	0%	0%	63%	0%	0%	55%	0%	0%	61%	51%	0%	0%	92%	0%	41%	52%	0%	0%	73%	0%	75%	0%
4	0%	30%	0%	0%	59%	0%	0%	23%	0%	0%	0%	27%	0%	0%	61%	0%	0%	17%	0%	0%	51%	0%	74%
5	21%	0%	15%	0%	0%	66%	0%	0%	11%	0%	0%	0%	13%	0%	0%	0%	0%	0%	11%	0%	0%	0%	0%
6	0%	50%	0%	0%	0%	0%	66%	0%	0%	64%	52%	0%	0%	94%	0%	38%	49%	50%	0%	70%	0%	75%	0%
7	0%	15%	0%	0%	23%	0%	0%	59%	0%	0%	0%	17%	0%	0%	20%	0%	0%	19%	0%	0%	35%	0%	29%
8	38%	0%	19%	0%	0%	13%	0%	0%	63%	0%	0%	0%	40%	0%	0%	0%	0%	0%	43%	0%	0%	0%	0%
9	0%	0%	0%	48%	0%	0%	55%	0%	0%	74%	50%	0%	0%	92%	0%	34%	51%	0%	0%	68%	0%	77%	0%
10	0%	0%	0%	51%	0%	0%	58%	0%	0%	66%	65%	0%	0%	92%	0%	35%	52%	0%	0%	73%	0%	82%	0%
11	0%	25%	0%	0%	33%	0%	0%	20%	0%	0%	0%	67%	0%	0%	58%	0%	0%	23%	0%	0%	38%	0%	71%
12	21%	0%	16%	0%	0%	19%	0%	0%	32%	0%	0%	0%	55%	0%	0%	0%	0%	0%	45%	0%	0%	0%	0%
13	0%	0%	0%	45%	0%	0%	52%	0%	0%	48%	47%	0%	0%	97%	0%	37%	42%	0%	0%	69%	0%	68%	0%
14	0%	30%	0%	0%	42%	0%	0%	26%	0%	0%	0%	34%	0%	0%	78%	0%	0%	19%	0%	0%	54%	0%	85%
15	0%	0%	0%	31%	0%	0%	32%	0%	0%	40%	32%	0%	0%	30%	0%	56%	31%	0%	0%	32%	0%	25%	0%
16	0%	0%	0%	49%	0%	0%	52%	0%	0%	61%	53%	0%	0%	90%	0%	30%	64%	0%	0%	71%	0%	78%	0%
17	0%	13%	0%	0%	29%	0%	0%	17%	0%	0%	0%	17%	0%	0%	37%	0%	0%	67%	0%	0%	46%	0%	18%
18	27%	0%	5%	0%	0%	8%	0%	0%	34%	0%	0%	0%	41%	0%	0%	0%	0%	0%	76%	0%	0%	0%	0%
19	0%	0%	0%	50%	0%	0%	46%	0%	0%	46%	46%	0%	0%	93%	0%	38%	47%	0%	0%	81%	0%	67%	0%
20	0%	26%	0%	0%	37%	0%	0%	28%	0%	0%	0%	23%	0%	0%	42%	0%	0%	21%	0%	0%	66%	0%	40%
21	0%	0%	0%	52%	0%	0%	53%	0%	0%	63%	53%	0%	0%	90%	0%	33%	53%	0%	0%	76%	0%	89%	0%
22	0%	27%	0%	0%	27%	0%	0%	21%	0%	0%	0%	35%	0%	0%	57%	0%	0%	16%	0%	0%	30%	0%	97%
BL	14%	22%	15%	29%	26%	17%	31%	14%	19%	42%	26%	26%	28%	73%	55%	15%	31%	21%	63%	48%	45%	69%	56%

Figure 5. Prediction accuracy of participants using trained deep learning networks from different subjects. White values are equal or less than BL probabilities, yellow ones slightly better and blue ones significantly better. Green values depict the accuracy on the corresponding player, which is never surpassed by a different model.

ing research, which aims to facilitate categorizing decisions between points of interest and/or incorporating more complex gameplay choices in multiplayer settings. For now, the computed movement models were used for an approximate representation in the following replay section.

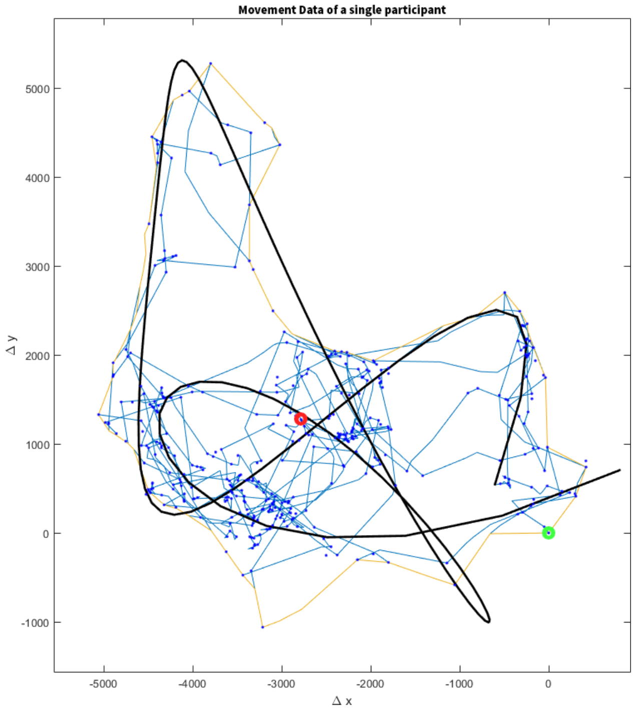


Figure 6. Movement data of a single participant. Blue lines visualize the users trajectory (starting at green, ending in the red spot), blue dots indicate skill usage. A yellow line encloses the travelled area. The thick black line shows the approximation of the movement behavior via a B-spline.

Replay

Since we are lacking clear and distinctive automatically measurable criteria on "what is a good representation" of human behavior and cannot objectively draw a threshold from which percentage on theoretical prediction accuracy establishes close behavior in actual in-game situations, we chose to replay the study session for each individual participant with the difference that the behavior stemmed not from the player directly, but from the computed model for the respective player. The simulated agent followed the approximated path from **Movement Analysis**, targeted nearby enemies and acted according to the individual **Deep Learning** model, which computed the most probable action given the situational parameters. We compared scores from both groups (human agent and replicated behavior) using a paired t-test (p=0.42, Cohen's $d = -0.07$) and Pearson correlation (r=0.91), which supports the quality of the model, since the outcome does not differ significantly and higher scores in the task completion correlate to higher scores in the replay.

DISCUSSION

Testing the performance of the different candidate techniques for player behavior modeling, deep learning showed the best prediction accuracy for immediate skill use, while HMMs and DTs show clear causal paths leading to the prediction decision, which can provide benefits in those potential application categories that require human interpretation of the modeled decisions (mostly B, game testing; and partially D, cheating and botting detection). The ability of the models to support the exemplary applications for analysis and exploration in the pilot study delivers early evidence for the potential of DPBM in the context of the general application categories.

However, some limitations apply. Due to the explorative nature of this study a number of factors that potentially impact player behavior were not included. Participants reported some variables that they perceived to be influencing their decision making, such as the amount of enemies in their immediate surrounding, the presence of buffs applied to the player or debuffs to the enemy that were not yet recorded and represented in the training and validation datasets. When describing PvP combat, players also adapt their strategy to their enemies' skill usage (and/or cool-downs). We did not model which enemies were attacked, which not and why not, yet this could have been helpful in order to further improve the replay session. Furthermore, the simple task did not bear any differences due to location that could be interpreted in movement analysis (aside from the area covered). In terms of the **replay** evaluation, a more expressive interpretation (such as analyzing human impressions by e.g. confronting players with videos of replay sessions and asking them to indicate which one represents themselves, without the information that the behavior is replicated) have to be considered in the future. For example, should players be able to correctly attribute model simulations to players for whom they were able to observe authentic gameplay behavior, it could be argued that the models appear to express characteristic and differentiable individual player behaviors.

Future Work

In general, we are looking forward to extend the evaluation in terms of number of players captured and in terms of the observed play duration, while also improving the models in terms of accuracy and efficiency, as to develop a fully-fledged toolkit which can be effectively employed for academic and industrial use. We will examine further the application of deep learning methods, exploring and evaluating the use of recurrent, deep-belief, or more context-driven (e.g., long / short-term memory) networks. Regarding the most crucial limitations of this paper (i.e. the small sample size and the missing consideration of the influence of additional players), we will broaden the scope of observations substantially: Based on the insights gained from single-player behavior we seek to expand the models towards the inclusion of true multiplayer situations (both cooperating with and competing against human players). This includes gathering additional data from wide-ranging "in the wild" behavior rather than from heavily scoped tasks, which is more representative for real-world application. Within the models, we will add events as possible predecessors for skills (such as a major location change, receiving a buff or debuff, starting a match, dying, etc). Finally, we aim to evaluate the concept of deep player behavior models in settings that are more directly representative of the established categories **A-D**.

Personalized Challenges will be approached by recording and modeling behavior of individual players, who will then face the task of defeating an agent controlled by a generative DPBM. Since deep learning has shown to yield a high prediction accuracy while still providing fast output retrieval, it will be the prioritized model for implementation. We will assess whether there is a perceived difference in engaging an opponent that acts in the same fashion as the player, compared to traditional NPCs, human players, and agent behavior that stems from a blend of multiple DPBM (modeled from players that are on approximately the same skill level as the former player). Based on these observations, we will focus on the perceived challenge, interestingness and long-term motivation of the involved players.

For **Autonomous Game Testing**, we will utilize DPBM for simulations of player behavior in order to spot game balance issues and establish automatic parameter tuning. This will be carried out in an iterative fashion, in order to approximate a theoretically solid balancing.

To assess the possibilities of DPBM for **Human Agent Substitution**, we will replace players in running multi-player PvP matches with their individual models without notifying the affected or the opposing team. Afterwards, interviews with all participants can uncover a Turing-test-style impression, whether they actually recognized the substitution and in how far it was perceived as too weak/idiosyncratic, too strong/imbalanced, or as a fair representation.

For **Cheating and Botting Detection**, we seek to collect a ground truth of behavior data between players utilizing forbidden methods and tools that yield unfair advantages and regular players (e.g. Aim-/TriggerBots, Keyboard macros, NoClip/-Gravity/-Animation Hacks or Memory manipulation might be reflected in movement and action selection). Looking forward to find discriminative variables to classify unwanted behavior, we will deploy and evaluate the resulting detection tool in live, real-world scenarios.

CONCLUSION

We introduced the concept of deep player behavior models (DPBM) in order to analyze, explain, and generate behavior stemming from individual human players in the MMORPG *Lineage II*. Different machine learning techniques were shown to bear different advantages in visualization (Hidden Markov Models; most useful for main skill rotation extraction), analysis (decision trees; most useful for pinning down of skill usage in specific situations; explaining overall usage of particular skills by reversing trees), and performance (deep learning; yielding high accuracy overall and proved to replicate behavior close to the original strategies by human players). Based on the computational models and on verbal reports, we can support the formerly constructed assumptions that players use skills a) in rotations and b) adjusted to specific situations (e.g., own/target HP status), while the adherence to use initial skills remains player-dependent. We also provided a working example of movement behavior approximation, were successful in calculating a difference metric between player behavior, and in replicating play sessions that displayed comparable performance to the modeled players. These exploratory applications establish the potential of DPBM for analysis and for behavior generation that can be beneficial in both academic and industrial use cases.

ACKNOWLEDGEMENTS

We would like to thank all participants, as well as the private *Lineage II* server *Imoriath* for advertising the study. This work was partially funded by the *German Research Foundation* (DFG) as part of Collaborative Research Center (Sonderforschungsbereich) 1320 *EASE - Everyday Activity Science and Engineering*, University of Bremen (http://www.ease-crc.org/), subproject H2 "Mining and Explicating Instructions for Everyday Activities", as well as by the *German Federal Ministry of Education and Research* (BMBF) through the project *Adaptify*, and the *DAAD FITweltweit* program.

REFERENCES

1. G. Andrade, G. Ramalho, H. Santana, and V. Corruble. 2005. Challenge-sensitive action selection: an application to game balancing. In *IEEE/WIC/ACM International Conference on Intelligent Agent Technology*. 194–200. DOI:http://dx.doi.org/10.1109/IAT.2005.52

2. Sora Ltd. BANDAI NAMCO Studios Inc. 2014. *Super Smash Bros. for Nintendo 3DS / for Wii U*. Game [WiiU, 3DS]. (13 September 2014). BANDAI NAMCO Studios Inc., Tokyo, Japan. Last played 2017.

3. P. Beau and S. Bakkes. 2016. Automated game balancing of asymmetric video games. In *2016 IEEE Conference on Computational Intelligence and Games (CIG)*. 1–8. DOI: http://dx.doi.org/10.1109/CIG.2016.7860432

4. B. Chan, J. Denzinger, D. Gates, K. Loose, and J. Buchanan. 2004. Evolutionary behavior testing of commercial computer games. In *Proceedings of the 2004*

Congress on Evolutionary Computation (IEEE Cat. No.04TH8753), Vol. 1. 125–132 Vol.1. DOI: http://dx.doi.org/10.1109/CEC.2004.1330847

5. Darryl Charles and Michaela Black. 2004. Dynamic Player Modelling: A Framework for Player-centred Digital Games. *Proceedings of the International Conference on Computer Games: Artificial Intelligence, Design and Education* (Jan. 2004).
6. Darryl Charles, Michael Mcneill, Moira Mcalister, Michaela Black, Adrian Moore, Karl Stringer, Julian Kücklich, and Aphra Kerr. 2005. Player-centred game design: Player modelling and adaptive digital games. *Proceedings of DiGRA 2005 Conference: Changing Views - Worlds in Play* (Jan. 2005).
7. Kuan-Ta Chen, Jhih-Wei Jiang, Polly Huang, Hao-Hua Chu, Chin-Laung Lei, and Wen-Chin Chen. 2009. Identifying MMORPG Bots: A Traffic Analysis Approach. *EURASIP J. Adv. Signal Process* 2009 (Jan. 2009), 3:1–3:22. DOI: http://dx.doi.org/10.1155/2009/797159
8. Gustavo Danzi, De Andrade, Hugo Pimentel Santana, André Wilson, Brotto Furtado, André Roberto Gouveia, A. Leitão, Geber Lisboa Ramalho, and Luis Freire. 2004. Online Adaptation of Computer Games Agents: A Reinforcement Learning Approach.
9. A. Drachen, R. Sifa, C. Bauckhage, and C. Thurau. 2012. Guns, swords and data: Clustering of player behavior in computer games in the wild. In *2012 IEEE Conference on Computational Intelligence and Games (CIG)*. 163–170. DOI:http://dx.doi.org/10.1109/CIG.2012.6374152
10. Nintendo EAD. 2014. *Mario Kart 8*. Game [WiiU, Switch]. (29 May 2014). Nintendo EAD, Kyoto, Japan. Last played 2017.
11. Christoph Eggert, Marc Herrlich, Jan Smeddinck, and Rainer Malaka. 2015. Classification of Player Roles in the Team-Based Multi-player Game Dota 2. In *Entertainment Computing - ICEC 2015*, Konstantinos Chorianopoulos, Monica Divitini, Jannicke Baalsrud Hauge, Letizia Jaccheri, and Rainer Malaka (Eds.). Springer International Publishing, Cham, 112–125.
12. Roberto Flores Emmett Tomai. 2009. Adapting in-game agent behavior by observation of players using learning behavior trees - Semantic Scholar. (2009).
13. Blizzard Entertainment. 2015. *Heroes of the Storm*. Game [PC]. (2 June 2015). Blizzard Entertainment, Irvine, California. Last played 2017.
14. Relic Entertainment. 2013. *Company of Heroes 2*. Game [PC]. (25 June 2013). Relic Entertainment, Vancouver, Canada. Last played 2017.
15. Julian Togelius Andy Nealen Fernando de Mesentier, Scott J Lee. 2017. AI as Evaluator: Search Driven Playtesting of Modern Board Games - Semantic Scholar. (2017).
16. Hector Fuster, Xavier Carbonell, Andres Chamarro, and Ursula Oberst. 2013. Interaction with the Game and Motivation among Players of Massively Multiplayer Online Role-Playing Games. *The Spanish journal of psychology* 16 (Nov. 2013), E43. DOI: http://dx.doi.org/10.1017/sjp.2013.54
17. Firaxis Games. 2010. *Civilization V*. Game [PC]. (21 September 2010). Firaxis Games, Hunt Valley, Maryland. Last played 2017.
18. S. Hilaire, H. c Kim, and C. k Kim. 2010. How to deal with bot scum in MMORPGs?. In *2010 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR 2010)*. 1–6. DOI:http://dx.doi.org/10.1109/CQR.2010.5619911
19. K. Hullett, N. Nagappan, E. Schuh, and J. Hopson. 2012. Empirical analysis of user data in game software development. In *Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*. 89–98. DOI: http://dx.doi.org/10.1145/2372251.2372265
20. Robin Hunicke and Vernell Chapman. 2004. AI for Dynamic Difficulty Adjustment in Games.
21. Disney Interactive. 2014. *Star Wars: Commander*. Game [iOS, Android, Windows Phone, PC]. (21 August 2014). Disney Interactive, Glendale, California. Last played 2017.
22. Alexander Jaffe, Alex Miller, Erik Andersen, Yun-En Liu, Anna Karlin, and Zoran Popović. 2012. Evaluating Competitive Game Balance with Restricted Play. In *Proceedings of the Eighth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE'12)*. AAAI Press, 26–31. http://dl.acm.org/citation.cfm?id=3014629.3014635
23. Hyungil Kim, Sungwoo Hong, and Juntae Kim. 2005. Detection of Auto Programs for MMORPGs. In *AI 2005: Advances in Artificial Intelligence (Lecture Notes in Computer Science)*. Springer, Berlin, Heidelberg, 1281–1284. DOI: http://dx.doi.org/10.1007/11589990_187
24. Koramgame. 2016. *Goddess: Primal Chaos*. Game [iOS, Android]. (2016). Koramgame, Hongkong, China. Last played 2017.
25. Ryan Leigh, Justin Schonfeld, and Sushil J. Louis. 2008. Using Coevolution to Understand and Validate Game Balance in Continuous Games. In *Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation (GECCO '08)*. ACM, New York, NY, USA, 1563–1570. DOI: http://dx.doi.org/10.1145/1389095.1389394
26. Chris Lewis and Noah Wardrip-Fruin. 2010. Mining Game Statistics from Web Services: A World of Warcraft Armory Case Study. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games (FDG '10)*. ACM, New York, NY, USA, 100–107. DOI:http://dx.doi.org/10.1145/1822348.1822362

27. Nicholas Liao, Matthew Guzdial, and Mark Riedl. 2017. Deep convolutional player modeling on log and level data. In *Proceedings of the 12th International Conference on the Foundations of Digital Games*. ACM, 41.

28. T. Mahlmann, J. Togelius, and G. N. Yannakakis. 2012. Evolving card sets towards balancing dominion. In *2012 IEEE Congress on Evolutionary Computation*. 1–8. DOI: http://dx.doi.org/10.1109/CEC.2012.6256441

29. Joe Marks and Vincent Hom. 2007. Automatic Design of Balanced Board Games. (01 2007).

30. John L. Miller and Jon Crowcroft. 2009. Avatar Movement in World of Warcraft Battlegrounds. In *Proceedings of the 8th Annual Workshop on Network and Systems Support for Games (NetGames '09)*. IEEE Press, Piscataway, NJ, USA, 1:1–1:6. http://dl.acm.org/citation.cfm?id=1837164.1837166

31. Y. Mishima, K. Fukuda, and H. Esaki. 2013. An Analysis of Players and Bots Behaviors in MMORPG. In *2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA)*. 870–876. DOI:http://dx.doi.org/10.1109/AINA.2013.108

32. NCsoft. 2003. *Lineage II*. Game [PC]. (1 October 2003). NCsoft, Seongnam, South Korea. Last played April 2018.

33. J. Oh, Z. H. Borbora, D. Sharma, and J. Srivastava. 2013. Bot Detection Based on Social Interactions in MMORPGs. In *2013 International Conference on Social Computing*. 536–543. DOI: http://dx.doi.org/10.1109/SocialCom.2013.81

34. J. Oh, Z. H. Borbora, and J. Srivastava. 2012. Automatic Detection of Compromised Accounts in MMORPGs. In *2012 International Conference on Social Informatics*. 222–227. DOI: http://dx.doi.org/10.1109/SocialInformatics.2012.69

35. Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2017. Automated Game Testing with ICARUS: Intelligent Completion of Adventure Riddles via Unsupervised Solving. In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '17 Extended Abstracts)*. ACM, New York, NY, USA, 153–164. DOI: http://dx.doi.org/10.1145/3130859.3131439

36. Robert W Plant and Richard M Ryan. 1985. Intrinsic motivation and the effects of self-consciousness, self-awareness, and ego-involvement: An investigation of internally controlling styles. *Journal of personality* 53, 3 (1985), 435–449.

37. Psyonix. 2015. *Rocket League*. Game [PC, XboxOne, PS4, Switch]. (7 July 2015). Psyonix, Satellite Beach, Florida. Last played 2017.

38. Stefan Radomski and Tim Neubacher. 2015. Formal Verification of Selected Game-Logic Specifications. *on Engineering Interactive Computer Systems with SCXML* (2015), 30.

39. Andrew Rollings and Dave Morris. 2003. Game Architecture and Design. (2003). https://dl.acm.org/citation.cfm?id=1209229

40. Richard M Ryan, C Scott Rigby, and Andrew Przybylski. 2006. The motivational pull of video games: A self-determination theory approach. *Motivation and emotion* 30, 4 (2006), 344–360.

41. Hannamari Saarenpää, Hannu Korhonen, and Janne Paavilainen. 2009. Asynchronous Gameplay in Pervasive Multiplayer Mobile Games. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems (CHI EA '09)*. ACM, New York, NY, USA, 4213–4218. DOI: http://dx.doi.org/10.1145/1520340.1520642

42. M. Shaker, M. H. Sarhan, O. A. Naameh, N. Shaker, and J. Togelius. 2013. Automatic generation and analysis of physics-based puzzle games. In *2013 IEEE Conference on Computational Inteligence in Games (CIG)*. 1–8. DOI: http://dx.doi.org/10.1109/CIG.2013.6633633

43. Kyong Jin Shim, Richa Sharan, and Jaideep Srivastava. 2010. Player Performance Prediction in Massively Multiplayer Online Role-Playing Games (MMORPGs). In *Advances in Knowledge Discovery and Data Mining (Lecture Notes in Computer Science)*. Springer, Berlin, Heidelberg, 71–80. DOI: http://dx.doi.org/10.1007/978-3-642-13672-6_8

44. D. Sirlin. 2009. Balancing multiplayer competitive games. Game Developer's Conference 2009, San Francisco, CA.

45. Finnegan Southey, Gang Xiao, Robert C. Holte, Mark Trommelen, and John W. Buchanan. 2005. Semi-Automated Gameplay Analysis by Machine Learning. 123–128.

46. Supercell. 2012. . Game [iOS, Android]. (2 August 2012). Supercell, Helsinki, Finland. Last played 2017.

47. Gabriel Synnaeve and Pierre Bessière. 2011. Bayesian Modeling of a Human MMORPG Player. *AIP Conference Proceedings* 1305, 1 (March 2011), 67–74. DOI:http://dx.doi.org/10.1063/1.3573658

48. Ruck Thawonmas, Yoshitaka Kashifuji, and Kuan-Ta Chen. 2008. Detection of MMORPG Bots Based on Behavior Analysis. In *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology (ACE '08)*. ACM, New York, NY, USA, 91–94. DOI: http://dx.doi.org/10.1145/1501750.1501770

49. Scott M. Thede and Mary P. Harper. 1999. A Second-order Hidden Markov Model for Part-of-speech Tagging. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics (ACL '99)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 175–182. DOI: http://dx.doi.org/10.3115/1034678.1034712

50. Julian Togelius, Renzo De Nardi, and Simon M Lucas. 2006. Making racing fun through player modeling and track evolution. (2006).

51. UNINCONGAME. 2016. *Pokeland Legends*. Game [iOS, Android]. (2016). UNINCONGAME, China. Last played 2017.

52. Valve. 2008. *Left 4 Dead*. Game [PC, XBox360]. (18 November 2008). Valve, Bellevue, Washington. Last played 2017.

53. Di Wang, Budhitama Subagdja, Ah-Hwee Tan, and Gee-Wah Ng. 2009. Creating human-like autonomous players in real-time first person shooter computer games. In *Proceedings, Twenty-First Annual Conference on Innovative Applications of Artificial Intelligence*. 173–178.

54. Jiyoung Woo, Hwa Jae Choi, and Huy Kang Kim. 2012. An automatic and proactive identity theft detection model in MMORPGs. *Appl. Math* 6 (Jan. 2012), 291S–302S.

55. Steven Woodcock. 2001. Game AI: the state of the art industry 2000-2001. *Game Developer* 8, 8 (2001), 36–44.

56. Georgios N Yannakakis and John Hallam. 2004. Evolving opponents for interesting interactive computer games. *From animals to animats* 8 (2004), 499–508.

57. Georgios N. Yannakakis and Manolis Maragoudakis. 2005. Player Modeling Impact on Player's Entertainment in Computer Games. In *User Modeling 2005*, Liliana Ardissono, Paul Brna, and Antonija Mitrovic (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 74–78.

58. Georgios N. Yannakakis, Pieter Spronck, Daniele Loiacono, and Elisabeth André. 2013. Player Modeling. In *Artificial and Computational Intelligence in Games*, Simon M. Lucas, Michael Mateas, Mike Preuss, Pieter Spronck, and Julian Togelius (Eds.). Dagstuhl Follow-Ups, Vol. 6. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 45–59. `http://drops.dagstuhl.de/opus/volltexte/2013/4335` DOI: 10.4230/DFU.Vol6.12191.45.

59. Nick Yee. 2006. The Demographics, Motivations, and Derived Experiences of Users of Massively Multi-user Online Graphical Environments. *Presence: Teleoper. Virtual Environ.* 15, 3 (June 2006), 309–329. DOI: `http://dx.doi.org/10.1162/pres.15.3.309`

60. Alexander Zook, Eric Fruchter, and Mark O. Riedl. 2014. Automatic playtesting for game parameter tuning via active learning. In *FDG*.

Bot or not? User Perceptions of Player Substitution with Deep Player Behavior Models

Johannes Pfau
Digital Media Lab, TZI,
University of Bremen
Bremen, Germany
jpfau@tzi.de

Jan David Smeddinck
Open Lab, School of Comp.,
Newcastle University
Newcastle upon Tyne, UK
jan.smeddinck@newcastle.ac.uk

Ioannis Bikas
Digital Media Lab, TZI,
University of Bremen
Bremen, Germany
bikasio@tzi.de

Rainer Malaka
Digital Media Lab, TZI,
University of Bremen
Bremen, Germany
malaka@tzi.de

ABSTRACT

Many online games suffer when players drop off due to lost connections or quitting prematurely, which leads to match terminations or game-play imbalances. While rule-based outcome evaluations or substitutions with bots are frequently used to mitigate such disruptions, these techniques are often perceived as unsatisfactory. Deep learning methods have successfully been used in deep player behavior modelling (DPBM) to produce non-player characters or bots which show more complex behavior patterns than those modelled using traditional AI techniques. Motivated by these findings, we present an investigation of the player-perceived awareness, believability and representativeness, when substituting disconnected players with DPBM agents in an online-multiplayer action game. Both quantitative and qualitative outcomes indicate that DPBM agents perform similarly to human players and that players were unable to detect substitutions. In contrast, players were able to detect substitution with agents driven by more traditional heuristics.

Author Keywords

Player Substitution; Game Disruption Prevention; Player Modeling; Neural Networks; Deep Learning; Games; Games User Research

CCS Concepts

•**Human-centered computing** → *User models;* •**Computing methodologies** → *Neural networks;* •**Applied computing** → *Computer games;*

CHI '20, April 25–30, 2020, Honolulu, HI, USA.

ACM 978-1-4503-6708-0/20/04…$15.00
DOI: http://dx.doi.org/10.1145/3313831.3376223

INTRODUCTION

Match disruptions in online games are one of the major causes for frustration reported by players and make for a frequent occurrence given varying network quality depending on location and over time [23, 5]. Designing and deploying scalable online games that avoid interruptions remains an important challenge [17]. Even with recent advances in network stability, the complete prevention of any disruptions is highly unlikely [4]. Apart from unintended cut-offs, disconnecting on purpose can also occur due to a range of reasons, such as *escaping*, in which players avoid their loss to be recorded, resentful behavior ("rage-quitting"), in which players seek to deprive their opponent(s) of victory or intentionally hurt their own team in collaboratively competitive games, as well as forced disconnects of opponents via glitches or third-party tools [58, 57, 32, 60, 31]. To counteract purposely caused interruptions, some games record them as losses or penalize them, which can lead to even higher frustration for non-self-inflicted disconnects [40]. Other examples of successful commercial games substitute disconnected players by heuristic, computer-controlled bots that continue playing (in some examples only until the original player reconnects), e.g. Left 4 Dead [54] (an FPS), Heroes of the Storm [13] (a multiplayer online battle arena game), Super Smash Bros. 4 [46] (a Beat'em up), Mario Kart 8 [11] (a racing game), Civilization V [15] (a turn-based strategy game), Company of Heroes 2 [14] (an RTS), or Rocket League [39] (a sports game). However, such substitution is frequently criticized, since the replacing bot is usually under-performing and not able to compete with human players. While modern machine learning approaches have proven to master a variety of games by continual improvement through simulated play [30, 49, 43], over-performing bots would also miss the point of adequate, representative substitutions, since they would yield an obvious and considerable potential for abuse.

Challenging all of the aforementioned issues, we approach the bridging of temporary match disruptions with a novel method, utilizing *Deep Player Behavior Modeling* (DPBM) to substitute disconnected players in ongoing online matches by learning agents that replicate the specific behavior of a given player.

In order to assess the applicability of this technique, the awareness of other involved players and whether DPBM replacements are perceived as representative of the prior player-behavior, we designed a study to accumulate evidence on the following research questions:

- **Can disconnected players in running online matches be substituted by DPBM agents without being detected?**
- **Do DPBM agents yield an adequate, fair representation that does not improve or worsen the original player's performance?**
- **Is DPBM capable of providing measurably better substitutions than traditional (heuristic) methods?**

We hypothesize that a sufficiently accurate representation of individual behavior will be indiscernible from the original human player and that DPBM is capable of implicitly approximating the player's game proficiency, leading to no significant perceived deviation in performance.

In order to establish a suitable test bed for the evaluation, we designed and implemented *Korona:Nemesis* [7], a platform fighter focused on player skills around prediction, learning and decision making. The game facilitates competitive skill-based play using an extended rock-paper-scissors mechanic to allow a broad range of play styles to arise by preference rather than encouraging dominant strategies. In an ecologically valid real-world field study ($n = 312$), we simulated substitutions of players during online matches and assessed detection rates, awareness towards bot presence and DPBM fitness over the course of four weeks. Our study shows that participants were not able to discriminate DPBM behavior from original human players and – at the same time – that they were significantly more likely to detect replacements with classic heuristically-driven bots. Between players that successfully detected a DPBM bot and those who were unaware, there were no differences in perceived performance or predictability. Supported by additional qualitative results, we conclude that DPBM are a suitable method for temporarily substituting disconnected players in online games and generate adequate and desirably human-like behavior. These findings contribute to game user research and game development alike, by demonstrating a technically feasible and successfully evaluated approach that can lay the foundations for considerable advancements in the challenge of overcoming negative consequences of online match disruptions.

RELATED WORK

Network stability and connection maintaining are under steady improvement, both in terms of progress on physical connections, as well as through the development of architectures and protocols for tackling discontinuity issues [55, 27, 38] or prediction of traffic anomalies to counteract bandwidth- or connectivity-loss before it becomes critical [16, 22]. Yet, online games are still vulnerable to connectivity disruptions, since they can arise from a large variety of potential error sources, ranging from fast-paced real-time mechanics over massively large amounts of simultaneous players to vast connection distance differences that can span continents. In combination, these issues are improbable to be overcome completely and can significantly impact the motivation of affected players and of other players in the same play-session. Disconnected players in cooperative team fights for example, have to be compensated for by allies which – depending on the game and genre – is unlikely to be manageable beyond short durations [18].

Originating from the more general approach of user modeling [3, 56, 61], the relatively young field of *player modeling* has developed steadily during the last decade, with approaches rooted in applications of machine learning techniques for data mining large sets of game protocols for purposes of analysis, prediction or classification [8, 26, 48, 42, 12], informing game development with player-specific insights [9, 6, 25], or the reproduction of limited, atomic tasks [51, 47]. Holmgård et al. studied *personas* for player decision modeling [19, 20] that continually observe and adapt to human behavior in order to produce agents with different decision making styles. These *personas* were realized via evolutionary linear perceptrons and compared to heuristic agents in a test-bed 2D dungeon crawler game, resulting in a higher player-rated human-likeness that could be utilized for game analysis, testing or providing believable opponents. They also assessed player models as defined as "deviations from theoretically rational actions" in a study of *Super Mario Bros.* [21, 1] and clustered these by means of feature extraction. Using the same game, Ortega et al. [34] imitated human playing styles by means of neuroevolution and dynamic scripting, reaching higher scores of human-likeness than performance-directed AI agents, based on subjective judgments. Missura and Gärtner utilized player modeling in a 2D test-bed shooter via support vector machines as a predictor for difficulty mismatches and to enable dynamic difficulty adjustment (DDA) based on the results [29]. Transforming the tracks of a racing game, Togelius et al. successfully deployed player modeling as a method of assessing entertainment metrics [50]. In previous work, we were successful in showing that player modeling agents yield significantly higher motivation potential than heuristic opponents [36]. In addition, we contrasted different machine learning techniques in a player modeling study of the MMORPG *Lineage 2* [33, 35], showing that deep learning offers the highest individual prediction accuracies with the ability to reproduce playing sessions that closely resemble the original behavior, as well as offering the potential to differentiate between players. Based on this, we embedded DPBM into a long-term DDA evaluation about competing against agents of own behavior on a daily basis in the MMORPG *AION* [37], in which DPBM opponents were perceived to be significantly more engaging than traditional DDA opponents adjusted by heuristic parameter tuning.

In computer generated behavior in general, human likeness or believability has been established as one of the most important metrics to facilitate engaging game play [52, 24, 2, 28, 53, 34, 19]. However, these approaches have focused on producing a general closeness to human behavior so far, not explicitly on representing behavior from specific individual players within

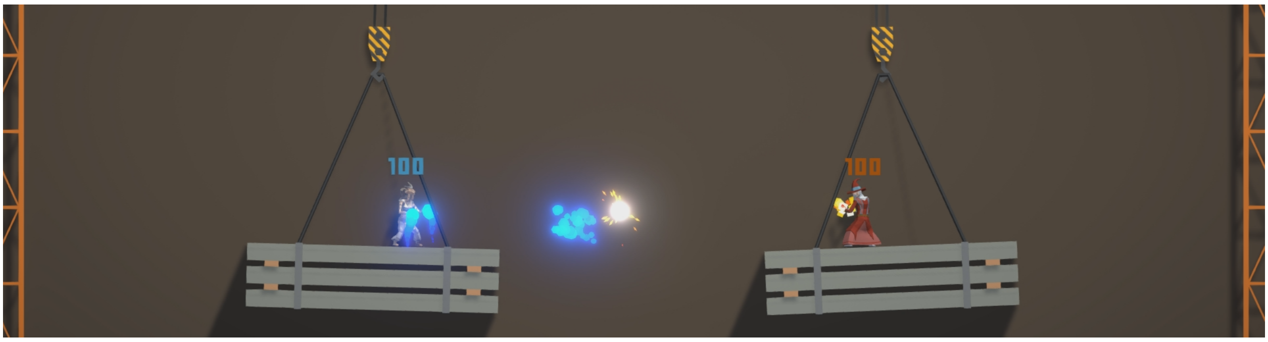


Figure 1. Screenshot extract of *Korona:Nemesis*. The player on the left utilizes Water to counter a Fire projectile, which will be extinguished.

the same game session. Although player disconnects pose long-standing challenges, substituting disconnected players by means of player modeling bots has not been approached in openly published materials before, neither academically nor in the games industry, and – to the best of our knowledge – there is no prior scientific research on alternative temporary replacements.

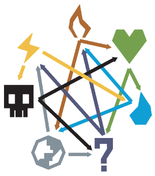
APPROACH

In this section, we outline a description, critical design decisions and mechanics of the game utilized for the evaluation, and provide a detailed overview of the architecture, method and parameters of the DPBM approach.

Game Design

To provide a setting for studying crucial decision making in real-time, we designed a fast-paced physic-based platform fighter called *Korona:Nemesis* that extends the classic rock-paper-scissors scheme to seven types of element projectiles (cf. Table 1). In each level, players are placed in a 2D environment, start with 100 health points (HP) and face the objective of eliminating their opponents' HP (last player standing wins). Players can *move* (left or right), *jump*, *attack* or *switch* actions using mouse and keyboard or an XBox or Playstation controller. *Switching* changes the current stance to one of the 7 elements. Giving the ability to chose any element at any time remedies potential balancing-issues, as the available action-spaces are – in principle – symmetric. *Attack* will launch an elemental projectile depending on the current stance. Getting hit by a hostile projectile subtracts 10 HP. Since this damage is doubled on a critical hit and projectiles can be destroyed, reflected or influenced by other projectiles (cf. Table 1), players constantly have to be aware of present projectiles, their own and enemies' stances and adapt quickly to the situation. As in rock-paper-scissors, predicting the opponent is key to success and since players adapt and react constantly, there is no single dominant strategy.

- Exemplary game-play scenario:
  When facing an incoming **Fire** projectile, there are multiple viable choices. The player might react with a **Water** attack, since **Water** projectiles destroy **Fire** projectiles (cf. Figure 1). A more offensive choice would be to counter this attack with a **Pain** attack, which would not stop the incoming projectile, but critically hit and ignite the opponent. At the same time, the opponent has the opportunity to re-counter this, depending on making good predictions (e.g. if predicting a **Water** counter-attack and intending to counter it with **Lightning**. Yet again, this strategy may fail: If the **Water** prediction turns out to be wrong, attacking **Pain** with **Lightning** will incur a critical hit).



Fire	Cancels **Restoration** Critically hits **Restoration/Steel** Destroys **Steel** projectiles Applies **burning** damage over time
Water	Immunity against **burning** Critically hits **Fire/Steel** Destroys **Fire** projectiles
Lightning	Immunity against **suffering** Critically hits **Water/Death** Destroys **Water** projectiles
Restoration	Restores 10HP Converts **Water** projectiles into 10HP Immunity against **Pain**
Steel	Reflects **Lightning** projectiles Reflects **Pain** projectiles Critically hits **Lightning/Pain**
Death	Inverts **Restoration** Critically hits **Restoration/Pain** Applies **suffering** damage over time
Pain	Self-ignites **Fire** Critically hits **Fire/Lightning** Applies 0.4 seconds stun

Table 1. Elements and their interactions in *Korona:Nemesis*.

Players need to learn not only the in-game element-interactions, but also their preferred way to counter attacks and maximize their chances, depending on the current situation. The presence of multiple viable choices, preferences and dislikes makes for a fertile ground for player modeling and decision making studies. For the evaluation of this work, participants were introduced to the mechanics via an in-game tutorial and were then able to play online matches consisting of 20 rounds in total.

Deep Player Behavior Modeling

Based on insights about expressive data and suitable modeling techniques from our earlier work [35, 36], we recorded all crucial player action decisions (*attacking* with – or *switching* to – a specific element and *jumping*) together with situational data from the current game state. After every level and for each player, the recorded behavioral data from all preceding levels was fed into a dedicated 24x10x10x9 feed-forward multi-layer perceptron with backpropagation and a logistic sigmoid activation function (cf. Figure 2). The network was initialized randomly and trained in a background thread over 1000 epochs, based on previous findings [35, 36] and benchmarks prior to the study that indicated diminishing returns beyond these parameters. When a DPBM bot substituted a player, it applied the trained model generatively to retrieve a set of action probabilities based on the given state description in real-time. After a weighted choice, it executed the most likely predicted skill and proceeded with querying the DPBM for the next situation, effectively approximating the learned behavior from the player's decision making so far. Since movement characteristics are rather limited within the game, motion behavior is approached heuristically. This implementation realizes a *model-free* (bottom-up) player modeling approach mapping *gameplay data* to actions via *preference learning* and *classification*, employing the player modeling taxonomy of Yannakakis et al. [59]. According to the player modeling description framework of Smith et al. [44], DPBM directly utilizes *game actions* **(domain)** to *generate* **(purpose)** *individually* **(scope)** modeled behavior by means of *induced* **(source)** training of machine learning techniques.

Heuristic Bots

Instead of DPBM bots, heuristic bots substituted players in situations where no recorded behavior or trained model was available, i.e.:

- When players waited for over 2 minutes in the online multiplayer lobby, heuristic bots filled the remaining slots to enable constant, comparable 4-player situations. Since DPBM training took place on the involved local machines parallel to the matches and the game followed a client-hosted design, no existing behavioral data could be acquired from a centralized server.
- When a player disconnected, but the background training thread for his DPBM counterpart was not completed. Yet, due to the considerably low temporal demand (cf. Results), this incidence occurred rarely.
- When a player deliberately disconnected before displaying enough behavior information for training.

Based on the insights of previous work [35, 36], we chose to endow the heuristic bots with *random* decision making between elements, since it yields a balanced performance level, analogous to random decision making in rock-paper-scissors. Thus, contrary to human and DPBM opponents, it was impossible for other players to predict this behavior. Movement was realized in the same heuristic fashion as for DPBM bots to avoid the detection of differences based on movement characteristics.

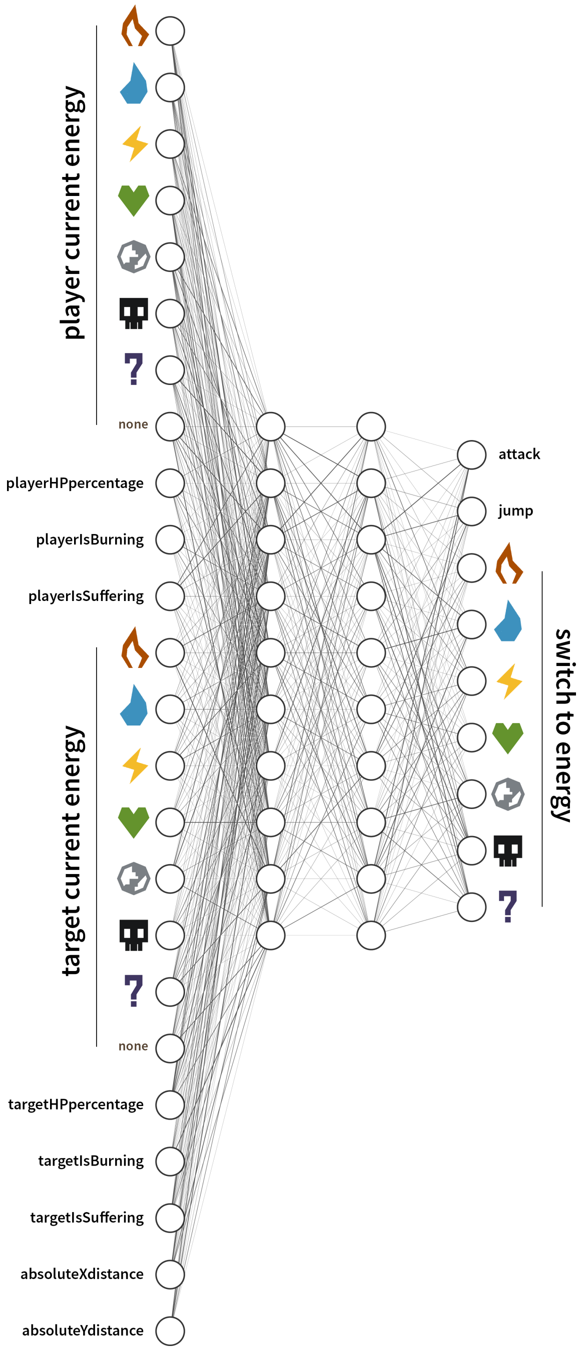


Figure 2. DPBM architecture for a single player; mapping game state (information about player and closest target) to action probabilities.

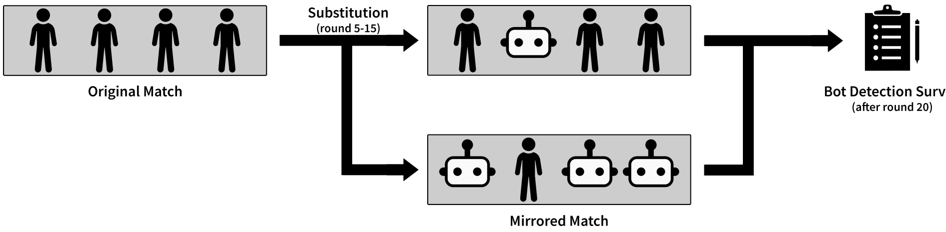


Figure 3. Study sequence for each match: from an initial configuration, one human player is shifted into a mirrored match with substituted opponents, while the player is replaced in the original match utilizing a DPBM bot trained on their prior behavior.

EVALUATION

The following section discusses the approach, design, setup and execution of the evaluation, separated into a pilot laboratory study and the main field study. For better clarity and explainability, we first elaborate on the field study, since the laboratory study only adds a qualitative assessment.

Field Study

To get a sufficiently large and expressive data set of ecologically valid measurements, we deployed the main study of this approach directly to a real-world target audience via a public release on the most popular game distribution platform *Steam* and gathered data during a study period of four weeks. We offered the game as free-to-play and concealed the appearance of an academic study during initial play to avoid confounding effects (e.g. experimenter bias [41]) until the point where players were asked to complete a follow-up survey. At this point, informed consent was gathered and data was stored in a pseudonymized fashion.

Measures

In-game, we recorded state-action data for DPBM training (cf. Figure 2), local training times and prediction accuracies of the DPBM, and the player's estimation whether and which players were controlled by a bot after every completed match. Additionally, players were asked to complete an online post-study questionnaire concerning demographics, subjective remarks and quantitative assessments of substitution awareness, asking the following set of 7-point Likert scale questions (separated by page transitions) that were constructed for this purpose:

- One of the players suddenly behaved differently.
- I felt that one player suddenly played better than they did before.
- I felt that one player suddenly played worse than they did before.
- I felt that one player suddenly became very predictable.
- I suspect that one of the players was switched for a bot.

Procedure

Participants could download *Korona:Nemesis* and play any number of matches without restrictions. Following a tutorial that demonstrated the basic mechanics of the game, they were able to enter the online multi-player lobby in which they waited for other players to join their match. If less than four players connected after two minutes, the remaining slots were filled by heuristic bots. During every active match, we intervened by substituting a random player by a DPBM bot that was trained in parallel to the playing session up until that point. If no trained model was available at that point, a heuristic bot took the place of the player. This replacement happened at a randomized point in time between round 5 and 15. To avoid discriminating the substituted players or diminishing their playing experience by being removed from play, they were immediately shifted into a new match that mirrored the original, differing only in the fact that the remaining three players were substituted in this version (cf. Figure 3).

The displayed appearance, name and score of replaced players was kept consistent in both matches at the time of the fork. After 20 rounds, players entered an end-screen depicting the ranking of all competitors, were encouraged answer the single in-game bot detection question and were then redirected to the main menu. In case they accepted the additional post-study questionnaire, they were referred to it using their standard browser.

Participants

During the study period, 1397 unique players downloaded *Korona:Nemesis*. ($n = 312$) submitted complete, pseudonymized game protocols and bot detection responses, encompassing 206 multi-player sessions in total. 24 of the players from these sessions (82.61% male, 17.39% female (self-identified), aged ($M = 22.4, SD = 3.75$)) completed the optional post-study questionnaire. 91.3% stated to be active gamers (playing multiple times a week), while 4.35% indicated that they only play occasionally (multiple times a month) and another 4.35% do not regularly play video games.

Explorative Laboratory Study

In order to pilot our approach and study design and to accumulate qualitative statements about reasons for detecting bots, the general perception of them and desirable behavior, we also conducted an explorative laboratory study ($n = 7$). Participants were publicly recruited on-campus of a university, asked to play a match of *Korona:Nemesis* and subsequently participated in a semi-structured interview. For reasons of clarity in our observations, only one of the four players necessary for a match was controlled by a participant, while trained experimenters filled the remaining slots, with one of them randomly being substituted. The experiment lasted about 30 minutes in total.

Measures

In addition to the aforementioned measures of the field study, a semi-structured interview assessed qualitative aspects of the player experience. Participants were able to provide free responses about the game, game experience and the behavior of their opponents, before the following directed questions were asked (in order and on separate pages).

- What do you think of the game?
- Did you notice anything strange during the game?
- Did you notice a change of behavior of other players?
- Do you think that there was a bot playing in this match?
- How can you tell that a player is actually a bot **(in general)**?
- How do you think bots in general should be improved to be (more) enjoyable?

Procedure

Following informed consent, participants were introduced to the game and asked to play the tutorial, without an enforced time limit. Once a player decided to proceed to visiting the online multi-player lobby, the experimenters joined soon thereafter, starting the match once the player count completed to four. All experimenters were kept spatially separated from the participants during the time of the match to avoid confounding factors from association or observation. The following procedure was analogous to the field study, only differing in the additional semi-structured interview that took place between match and post-study questionnaire.

Participants

($n = 7$) subjects participated in the explorative pilot study (62.5% male, 37.5% female (self-identified), aged ($M = 23.86$, $SD = 3.34$)). 42.86% self-identified as active gamers (playing multiple times a week), while 28.57% respectively indicated that they only play occasionally (multiple times a month) or do not regularly play video games.

RESULTS

The following quantitative outcomes resulted from the main field study, while qualitative insights of the laboratory pilot study are discussed at the end of the section.

guessed behavior	actual behavior human	 DPBM bot	 heuristic bot
isHuman	87.18% (68)	85.48% (53)	32.75% (75)
isBot	12.82% (10)	14.52% (9)	67.25% (154)

Table 2. Percentages (and absolute numbers in parentheses) of bot detection estimates, according to the responses to the in-game bot detection survey.

Using a chi-square test of independence with Yates-correction, a significant difference in guessing whether a player's behavior stems from a human or bot could be found based on the groups of **actual human players**, **DPBM bots** and **heuristic bots** ($\chi^2_{2,369} = 97.11, p < .001$, Kramer's $v = 0.36$), (cf. Table 2 for percentages and absolute estimate numbers). For differentiation between bot types, we further assessed the differences between the three particular groups:

Actual human players and **DPBM bots**:
$\chi^2_{1,140} = .002$ (not significant)
Actual human players and **heuristic bots**:
$\chi^2_{1,307} = 67.1, p < 0.001$ (significant), $\phi = .47$
DPBM bots and **heuristic bots**:
$\chi^2_{1,291} = 52.95, p < 0.001$ (significant), $\phi = .43$

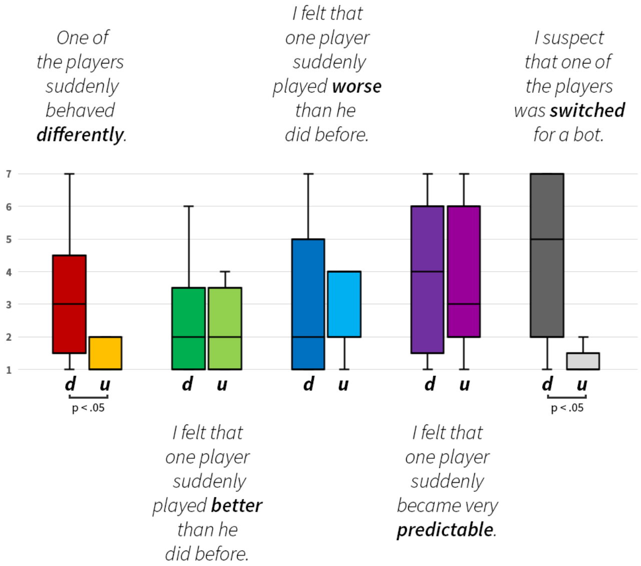


Figure 4. Boxplot illustrating the results (medians, standard deviations as boxes, minima and maxima as whiskers, significant differences in-between) of the custom awareness scale between players that detected *(d)* a bot and players unaware *(u)* of substitution.

Concerning the awareness scale constructed for this study, we compared answers between players that managed to successfully **detect** a substitution and players **unaware** of it, in order to gain insights about if detected bots would alter the perceived behavior or performance (cf. Figure 4). Using a two-tailed unpaired t-test (after validations for uniform distribution), we found no significant difference in the subjective predictability ($t_{23} = .17, p = .86$), performance improvement ($t_{23} = .33, p = .74$) or performance decline ($t_{23} = -.02, p = .98$) between these groups. There were significant findings regarding the questions

"One of the players suddenly behaved differently."
($t_{23} = 2.10, p = .04$, Cohen's $d = 1.3$)
and *"I suspect that one of the players was switched for a bot."*
($t_{23} = 3.11, p = .005$, Cohen's $d = 1.98$).

The average DPBM training time (computed locally on each game client) amounted to ($M = 2.23, SD = 2.87$) seconds. Within each iteration, 80% of the recorded data was used for training, while the remaining 20% allowed for following routine tests, resulting in a prediction accuracy of ($M = 82.17\%, SD = 23.17\%$). There was a strong positive correlation between the *amount of data* points used for training and the *prediction accuracy* of the following test (Pearson's $r_{2871} = .64, p < .01$).

Explorative Laboratory Study

Additionally, the laboratory pilot study yielded augmentative qualitative results. 6 of 7 participants remarked that they liked the game overall. None of them noticed anything generally strange in the session, nor a change in behavior of one of the players. Regarding the question whether they recognized a bot, no one managed to provide a correct answer (4 of them did not detect a substituted player, 3 incorrectly judged human players to be bots). In a notable contradiction to this finding, when asked, what they expect from the behavior of a bot, the participants consistently responded that bots are typically noticeable due to their bad performance (5x) or predictable strategies (3x). In response to the question *"How do you think bots in general should be improved to be (more) enjoyable?"*, they stated that they *"would like them to be as human as possible"*, would want bots that are *"adaptive (like humans), but not with superhuman performance"*, and that *"playing with real people feels better"*.

DISCUSSION & FUTURE WORK

With respect to the initial research question **"Can disconnected players in running online matches be substituted by DPBM agents without being detected?"**, we found quantitative as well as qualitative outcomes that support our hypothesis that DPBM yields a feasible approach for player substitution. The results of the bot detection estimation (cf. Table 2) indicate that participants were not able to differentiate between human and DPBM behavior, even if they were substituted during a running match. The significant difference of this finding to the frequent detection of heuristic bots answers **"Is DPBM capable of providing measurably better substitutions than traditional (heuristic) methods?"** in favor for DPBM and amplifies the expressiveness of the former results, since players evidently *were* able to detect bots, if their behavior was less human-like. Qualitative insights from the laboratory study complete the picture of a successful substitution, since participants stated to be unaware of changes in behavior after DPBM substitutions and were unable to correctly name replaced players.

The true positive rate of 87.18% for human behavior aligns fittingly with related research in which participants were asked to judge game sessions according to whether a human was playing *The Legend of Zelda* [10] (88.7%) or *Boulder Dash* [45] (80.7%) [24].

Regarding the remaining research question **"Do DPBM yield an adequate, fair representation that does not improve or worse the original player's performance?"**, we provide evidence based on the awareness questionnaire constructed for the purpose of this study. Player proficiency or performance can develop during game play, but there was no significant increase or decrease or change in predictability between detected bots and undetected bots or regular players. Together with a considerably high DPBM prediction accuracy, this supports the claim that DPBM behavior does not significantly deviate from the original human player behavior. Additionally, our approach meets the desired ideal behavior of bots, according to the qualitative statements that players prefer to play against opponents that are as human-like as possible.

Still, this study faces limitations. In general remarks on the field study, 3 participants stated that they played *Korona:Nemesis* simultaneously with a friend who took part in the same session, while constantly communicating. The discrepancy between the original and the mirrored match (that could be communicated between the players) was the main cause of detecting the substitution for these players, as opposed to actually judging changes in behavior. We were not able to prevent this potentially confounding factor in the large-scale field study, as we aimed for maximizing the ecological validity of the approach. However, even if this introduced a bias to our results, it would have *increased* the correct bot detection rates, which actually would *decrease* the possibility of a non-significant result of the bot detection estimation between human and DPBM opponents. The result, that people were not able to discriminate human and DPBM behavior nonetheless indicates that this bias was not significantly confounding.

Furthermore, one player claimed that a real-time game might not be the best test bed for substitution awareness, since players are too focused on themselves. While we can not disprove this assertion or control for some extent of bias, we explicitly designed *Korona:Nemesis* in an extended rock-paper-scissors fashion in which players have to pay attention to their opponent. Moreover, we argue that artificial behavior would likely be even more indiscernible in many other types of games, such as turn-based games, since action decisions that might seem idiosyncratic or not human-like would likely be assumed to be part of larger complex strategies that are common to turn-based games. Altogether, our study can only provide high certainty that DPBM player substitution works adequately, fairly and indiscernibly as implemented for the game *Korona:Nemesis*. Yet, we designed the game to be complex enough to facilitate individual preference formation and to require attention, prediction, learning and tactical decision-making without incorporating dominant strategies. We argue that the insights formed in this approach can be extended and generalized to other games in the genres of fighting games and decision-making-focused action RPGs. We are looking forward to assess awareness, believability and representativity of DPBM opponents in these and further genres, including turn-based, cooperative games and games that encompass complex movement characteristics.

The DPBM architecture was kept as frugal as possible, in order to ensure feasible training times on the uncharted multitude of different hardware constellations that were able to acquire the game via *Steam*. The low average time required for network training, however, suggests some room for elaborating more ambitious deep player modeling techniques (e.g. recurrent, deep belief, GAN or context-driven LSTM networks) to further improve the proximity to human-like behavior, or to model more complex observation-to-action mappings. Since – to the best of our knowledge – no evidence in the field of player modeling exists that would give an estimation about the connection of prediction accuracy and perceived human-likeness, we seek to aggregate data for a large-scale evaluation in which participants are asked to watch game sessions of DPBM agents with different gradations of prediction accuracy, judge them according to their human-likeness and allocate them to the correct human player from which the behavior originated. Additionally, no prior research exists that evaluates the perception of fairness when it comes to substituting players. Thus, we plan to assess this from both the substituted player's perspective, as well as the impression from involved team mates and opponents.

Eventually, we envision DPBM as an effective instrument for elevating autonomous game testing and balancing, since realistic player behavior can be employed, as well as for facilitating novel dynamic difficulty adjustment approaches that adapt to individual strengths, weaknesses and progresses of players over time.

CONCLUSION

Since unintentional, as well as deliberate disconnects, drop offs or client terminations are unlikely to disappear with conventional, stability-improving hardware and software methods, we demonstrated an alternative approach that bridges (temporary) player absence by substituting them with Deep Player Behavior Models (DPBM). An ecologically valid online field study ($n = 312$) with a duration of four weeks simulated the replacement of a human player in the online multi-player fighting platformer *Korona:Nemesis*, assessing the remaining players' awareness, the believability of the substitution, and the performance-related representativeness. We conclude that players were not able to distinguish between DPBM bots and original human players, but notably managed to detect bots based on heuristic behavior. Perceived performance and predictability changes did not differ between players who did detect DPBM bots and players who indicated that they thought that they had been playing against other human players only. All together, we implemented and evaluated a novel approach to tackling online match disruptions and lay ground for further evaluations spanning additional games, genres and integrations.

According to the guidelines of transparent statistics, the collected data of this approach, as well as its implementation, will be made openly available upon publication, using an open-access repository.

ACKNOWLEDGMENTS

We would like to thank all participants. This work was funded by the German Research Foundation (DFG) as part of Collaborative Research Center (SFB) 1320 EASE - Everyday Activity Science and Engineering, University of Bremen (http://www.easecrc.org/), subproject H2.

REFERENCES

[1] Nintendo Research Development 4. 1985. *Super Mario Bros.* Game [NES]. (13 September 1985). Nintendo, Kyōto, Japan.

[2] Giovanni Acampora, Vincenzo Loia, and Autilia Vitiello. 2012. Improving game bot behaviours through timed emotional intelligence. *Knowledge-Based Systems* 34 (2012), 97–113.

[3] Nikola Banovic, Antti Oulasvirta, and Per Ola Kristensson. 2019. Computational Modeling in Human-Computer Interaction. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, W26.

[4] Sandra Braman. 2016. Instability and internet design. *Internet Policy Review* 5, 3 (2016).

[5] Fabio Reis Cecin, Rodrigo Real, Rafael de Oliveira Jannone, CF Resin Geyer, Marcio Garcia Martins, and JL Victoria Barbosa. 2004. Freemmg: A scalable and cheat-resistant distribution model for internet games. In *Eighth IEEE International Symposium on Distributed Simulation and Real-Time Applications*. IEEE, 83–90.

[6] Darryl Charles, A Kerr, M McNeill, M McAlister, M Black, J Kcklich, A Moore, and K Stringer. 2005. Player-centred game design: Player modelling and adaptive digital games. In *Proceedings of the digital games research conference*, Vol. 285. 00100.

[7] Nevermind Creations. 2019. *Korona:Nemesis*. Game [PC]. (18 August 2019).

[8] A. Drachen, A. Canossa, and G. N. Yannakakis. 2009. Player modeling using self-organization in Tomb Raider: Underworld. In *2009 IEEE Symposium on Computational Intelligence and Games*. 1–8.

[9] Anders Drachen, Rafet Sifa, Christian Bauckhage, and Christian Thurau. 2012. Guns, swords and data: Clustering of player behavior in computer games in the wild. In *2012 IEEE conference on Computational Intelligence and Games (CIG)*. IEEE, 163–170.

[10] Nintendo EAD. 1986. *The Legend of Zelda*. Game [NES]. (21 February 1986). Nintendo EAD, Kyoto, Japan.

[11] Nintendo EAD. 2014. *Mario Kart 8*. Game [WiiU,Switch]. (29 May 2014). Nintendo EAD, Kyoto, Japan. Played 2019.

[12] Christoph Eggert, Marc Herrlich, Jan Smeddinck, and Rainer Malaka. 2015. Classification of player roles in the team-based multi-player game dota 2. In *International Conference on Entertainment Computing*. Springer, 112–125.

[13] Blizzard Entertainment. 2015. *Heroes of the Storm*. Game [PC]. (2 June 2015). Blizzard Entertainment, Irvine, CA, USA. Played 2017.

[14] Relic Entertainment. 2013. *Company of Heroes 2*. Game [PC]. (25 June 2013). Relic Entertainment, Vancouver, Canada. Played 2017.

[15] Firaxis Games. 2010. *Civilization V*. Game [PC]. (21 September 2010). Firaxis Games, Hunt Valley, Maryland. Played 2017.

[16] Chengjie Gu, Shunyi Zhang, Xiaozhen Xue, and He Huang. 2011. Online wireless mesh network traffic classification using machine learning. *Journal of Computational Information Systems* 7, 5 (2011), 1524–1532.

[17] Yong Guo, Siqi Shen, Otto Visser, and Alexandru Iosup. 2012. An analysis of online match-based games. In *2012 IEEE International Workshop on Haptic Audio Visual Environments and Games (HAVE 2012) Proceedings*. IEEE, 134–139.

[18] Brian Guthrie, Kevin Reuter, Michael Barkdoll, and Henry Hexmoor. 2014. Small team group dynamics in online games. *COOS: Scope and theme* (2014), 42.

[19] Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N Yannakakis. 2014a. Evolving personas for player decision modeling. In *2014 IEEE Conference on Computational Intelligence and Games*. IEEE, 1–8.

[20] Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N Yannakakis. 2014b. Generative agents for player decision modeling in games. In *FDG '14*. Citeseer.

[21] Christoffer Holmgård, Julian Togelius, and Georgios N Yannakakis. 2013. Decision making styles as deviation from rational action: A super mario case study. In *Ninth Artificial Intelligence and Interactive Digital Entertainment Conference*.

[22] Shay Horovitz and Danny Dolev. 2009. Collabrium: Active traffic pattern prediction for boosting P2P collaboration. In *2009 18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*. IEEE, 116–121.

[23] Arnaud Kaiser, Dario Maggiorini, Nadjib Achir, and Khaled Boussetta. 2009. On the objective evaluation of real-time networked games. In *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference*. IEEE, 1–5.

[24] Ahmed Khalifa, Aaron Isaksen, Julian Togelius, and Andy Nealen. 2016. Modifying MCTS for Human-Like General Video Game Playing.. In *IJCAI*. 2514–2520.

[25] Chris Lewis and Noah Wardrip-Fruin. 2010. Mining game statistics from web services: a World of Warcraft armory case study. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games*. Citeseer, 100–107.

[26] Tobias Mahlmann, Anders Drachen, Julian Togelius, Alessandro Canossa, and Georgios N Yannakakis. 2010. Predicting player behavior in tomb raider: Underworld. In *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games*. IEEE, 178–185.

[27] Philip Mildner, Tonio Triebel, Stephan Kopf, and Wolfgang Effelsberg. 2011. A scalable Peer-to-Peer-overlay for real-time massively multiplayer online games. In *Proceedings of the 4th international ICST conference on simulation tools and techniques*. ICST (Institute for Computer Sciences, Social-Informatics and …, 304–311.

[28] Maximiliano Miranda, Antonio A Sánchez-Ruiz, and Federico Peinado. 2016. A Neuroevolution Approach to Imitating Human-Like Play in Ms. Pac-Man Video Game.. In *CoSECivi*. 113–124.

[29] Olana Missura and Thomas Gärtner. 2009. Player Modeling for Intelligent Difficulty Adjustment. In *Discovery Science*, João Gama, Vítor Santos Costa, Alípio Mário Jorge, and Pavel B. Brazdil (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 197–211.

[30] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, and others. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (2015), 529.

[31] Ryan M Moeller, Bruce Esplin, Steven Conway, and others. 2009. Cheesers, pullers, and glitchers: The rhetoric of sportsmanship and the discourse of online sports gamers. *Game Studies* 9, 2 (2009).

[32] K Mørch. 2003. Cheating in online games-threats and solutions. *Publication No: DART/01/03. January* (2003).

[33] NCsoft. 2003. *Lineage 2*. Game [PC]. (1 October 2003). NCSoft, Seongnam, South Korea.

[34] Juan Ortega, Noor Shaker, Julian Togelius, and Georgios N Yannakakis. 2013. Imitating human playing styles in super mario bros. *Entertainment Computing* 4, 2 (2013), 93–104.

[35] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2018. Towards Deep Player Behavior Models in MMORPGs. In *Annual Symp. on Computer-Human Interaction in Play Ext. Abstracts (CHI PLAY '18)*. ACM, New York, NY, USA, 381–92.

[36] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2019. Deep Player Behavior Models: Evaluating a Novel Take on Dynamic Difficulty Adjustment. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, LBW0171.

[37] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2020. Enemy Within: Long-term Motivation Effects of Deep Player Behavior Models for Dynamic Difficulty Adjustment. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM.

[38] Jared N Plumb, Sneha Kumar Kasera, and Ryan Stutsman. 2018. Hybrid network clusters using common gameplay for massively multiplayer online games. In *Proceedings of the 13th International Conference on the Foundations of Digital Games*. ACM, 2.

[39] Psyonix. 2015. *Rocket League*. Game [PC,XboxOne,PS4,Switch]. (7 July 2015).

[40] Rosslin John Robles, Sang-Soo Yeo, Young-Deuk Moon, Gilcheol Park, and Seoksoo Kim. 2008. Online games and security issues. In *2008 Second International Conference on Future Generation Communication and Networking*, Vol. 2. IEEE, 145–148.

[41] Robert Rosenthal and Kermit L Fode. 1963. The effect of experimenter bias on the performance of the albino rat. *Behavioral Science* 8, 3 (1963), 183–189.

[42] Kyong Jin Shim, Richa Sharan, and Jaideep Srivastava. 2010. Player performance prediction in massively multiplayer online role-playing games (MMORPGs). In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 71–80.

[43] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, and Laurent Sifre et al. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529 (2016), 484–489.

[44] Adam M. Smith, Chris Lewis, Kenneth Hullet, Gillian Smith, and Anne Sullivan. 2011. An Inclusive View of Player Modeling. In *Proceedings of the 6th International Conference on Foundations of Digital Games (FDG '11)*. ACM, New York, NY, USA, 301–303.

[45] First Star Software. 1984. *Boulder Dash*. Game [Atari]. (1984). First Star Software, New York City, NY.

[46] BANDAI NAMCO Studios Inc Sora Ltd. 2014. *Super Smash Bros. for Nintendo 3DS / for Wii U*. Game [WiiU,3DS]. (13 September 2014). Sora Ltd, BANDAI NAMCO Studios Inc, Tokyo, Japan. Played 2017.

[47] Gabriel Synnaeve and Pierre Bessière. 2011. Bayesian modeling of a human MMORPG player. In *AIP Conference Proceedings*, Vol. 1305. AIP, 67–74.

[48] Marco Tamassia, William Raffe, Rafet Sifa, Anders Drachen, Fabio Zambetta, and Michael Hitchens. 2016. Predicting player churn in destiny: A hidden markov models approach to predicting player departure in a major online game. In *2016 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 1–8.

[49] Gerald Tesauro. 1994. TD-Gammon, a self-teaching backgammon program, achieves master-level play. *Neural computation* 6, 2 (1994), 215–219.

[50] Julian Togelius, Renzo De Nardi, and Simon M Lucas. 2006. Making racing fun through player modeling and track evolution. (2006).

[51] Emmett Tomai and Roberto Flores. 2014. Adapting in-game agent behavior by observation of players using learning behavior trees. In *FDG '14*.

[52] AM Turing. 1950. Mind. *Mind* 59, 236 (1950), 433–460.

[53] Iskander Umarov and Maxim Mozgovoy. 2014. Creating believable and effective AI agents for games and simulations: Reviews and case study. In *Contemporary Advancements in Information Technology Development in Dynamic Environments*. IGI Global, 33–57.

[54] Valve. 2008. *Left 4 Dead*. Game [PC]. (18 November 2008). Valve, Bellevue, WA, USA. Played 2017.

[55] Amir Yahyavi and Bettina Kemme. 2013. Peer-to-peer architectures for massively multiplayer online games: A survey. *ACM Computing Surveys (CSUR)* 46, 1 (2013), 9.

[56] Huan Yan, Chunfeng Yang, Donghan Yu, Yong Li, Depeng Jin, and Dah-Ming Chiu. 2019. Multi-site user behavior modeling and its application in video recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2019).

[57] Jeff Yan and Brian Randell. 2005. A systematic classification of cheating in online games. In *Proceedings of 4th ACM SIGCOMM workshop on Network and system support for games*. ACM, 1–9.

[58] Jeff Yan and Brian Randell. 2009. An investigation of cheating in online games. *IEEE Security & Privacy* 7, 3 (2009), 37–44.

[59] Georgios N Yannakakis, Pieter Spronck, Daniele Loiacono, and Elisabeth André. 2013. Player modeling. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[60] George Yee, Larry Korba, Ronggong Song, and Ying-Chieh Chen. 2006. Towards designing secure online games. In *20th International Conference on Advanced Information Networking and Applications-Volume 1 (AINA'06)*, Vol. 2. IEEE, 44–48.

[61] Hongzhi Yin, Bin Cui, Ling Chen, Zhiting Hu, and Xiaofang Zhou. 2015. Dynamic user modeling in social media systems. *ACM Transactions on Information Systems (TOIS)* 33, 3 (2015), 10.

Enemy Within: Long-term Motivation Effects of Deep Player Behavior Models for Dynamic Difficulty Adjustment

Johannes Pfau
Digital Media Lab, TZI,
University of Bremen
Bremen, Germany
jpfau@tzi.de

Jan David Smeddinck
Open Lab, School of Comp.,
Newcastle University
Newcastle upon Tyne, UK
jan.smeddinck@newcastle.ac.uk

Rainer Malaka
Digital Media Lab, TZI,
University of Bremen
Bremen, Germany
malaka@tzi.de

ABSTRACT

Balancing games and producing content that remains interesting and challenging is a major cost factor in the design and maintenance of games. Dynamic difficulty adjustment (DDA) can successfully tune challenge levels to player abilities, but when implemented with classic heuristic parameter tuning (HPT) often turns out to be very noticeable, e.g. as “rubber-banding”. Deep learning techniques can be employed for deep player behavior modeling (DPBM), enabling more complex adaptivity, but effects over frequent and longer-lasting game engagements, as well as comparisons to HPT have not been empirically investigated. We present a situated study of the effects of DDA via DPBM as compared to HPT on intrinsic motivation, perceived challenge and player motivation in a real-world MMORPG. The results indicate that DPBM can lead to significant improvements in intrinsic motivation and players prefer game experience episodes featuring DPBM over experience episodes with classic difficulty management.

CCS Concepts
•Human-centered computing → **User models;** **•Computing methodologies** → *Neural networks;* **•Applied computing** → *Computer games;*

Author Keywords
Dynamic difficulty adjustment; Player Modeling; Neural Networks; Deep Learning; MMORPGs; Games

INTRODUCTION

With the ongoing rise of complexity, popularity and content production cost of video game development, consistently balanced challenges that keep players motivated over the long term are becoming hard to attain, especially with large player communities encompassing broad ranges of proficiency. Dynamic Difficulty Adjustment (DDA) [23] denotes the principle of adapting video game challenges to players' abilities - both mental and physical / dexterity - in order to allow motivation-fostering flow states [10] to arise. It has been successfully deployed in scientific [23, 24, 46] and industrial [7, 16, 55] contexts and is usually accomplished by continuous tuning of core game variables (such as speed, damage or hit ratio). However, these systems are inherently limited to a small number of high-level parameters, require careful tuning of thresholds in heuristic parameter tuning (HPT) [51] and have to be hidden to avoid exploitation; e.g. as not to incentivize players to perform badly on purpose [43]. This results in limited expressivity and complexity of system behavior, as well as in considerable development cost. To address these limitations of classic HPT, we present a novel DDA strategy by implicitly assessing individual player proficiency using Deep Player Behavior Modeling (DPBM) [38] and generating adaptive, personalized challenges. Player behavior, in terms of state-action decision making, is captured while fighting an in-game opponent and trained onto an individual, initially randomized model. Upon the next encounter, the opponent uses this model generatively by retrieving action probabilities for each game state emerging in an interaction. As a consequence, its decision making approximates the original player's behavior, implicitly representing the particular game proficiency. Evaluating the real-world applicability of DPBM, we aim to answer the following research questions:

- **Do players perceive behavior from DPBM as representative of their own decision making?**
- **Is the players' self-reported intrinsic motivation when interacting with a DPBM opponent higher than for traditional HPT encounters?**
- **Can we measure a substantial long-term motivation achieved by DPBM?**

We hypothesize that an agent that keeps up with the progress of the player, displays similar weaknesses and challenges players to constantly improve or rethink strategies will yield a novel and captivating take on DDA. For the purpose of evaluating the differences between HPT and DPBM, we implemented *Eternal Challenge* – an adaptive instance dungeon inside of the popular Massively Multiplayer Online Role-Playing Game (MMORPG) *Aion* [34] – and assessed players' motivation in a field study ($n = 171$) on an existing private server. After a deployment of four weeks, we were successful in showing a significantly higher long-term usage of the instance compared to all alternatives and that the DPBM opponent contributes significantly more to this motivation than HPT mechanics,

CHI '20, April 25–30, 2020, Honolulu, HI, USA.
© 2020 Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-6708-0/20/04 ...$15.00.
http://dx.doi.org/10.1145/3313831.3376423

Figure 1. Appearance of the different opponents with DDA through HPT variables and DPBM.

based on quantitative – and supported by qualitative – insights. Furthermore, some players stated that they noted the progress of the DPBM opponent in learning their own individual strategies, leading to a unique game experience. We contribute to games user research and inform game development by investigating DPBM as a form of novel DDA in a situated medium- to long-term study, showcasing its distinct potential for fostering intrinsic motivation and demonstrating a working approach with learning adaptive opponents in the wild.

RELATED WORK

Providing and balancing an accurate level of difficulty is critical for keeping players constantly engaged [2, 23]. Disparities can ultimately lead to boredom/underload or frustration/overload, which make for two of the main causes why players stop playing games [11]. Since individual skill and its progression are hard to foresee throughout potentially large player bases and difficulty and it's progression can not be defined or programmed precisely, the field of DDA attempts to regulate emergent mismatches dynamically. To estimate imbalanced challenge-proficiency-discrepancies, various assessment techniques have been researched, such as success probability estimation [24, 32] or biofeedback [22, 25, 35]. However, When it comes to adjusting this difficulty, most approaches focus on HPT (apart from procedural world generation [26, 57, 50]), even in the most recent advancements [3, 4, 8, 17, 18, 40].

Opponents that imitate the player character exist in numerous commercial games, perhaps most notably the recurring *Dark Link* in the *The Legend of Zelda* series [15], the *Guild Wars Doppelganger* [5], *Renegade Shepard* from *Mass Effect 3* [6] and *SA-X* in *Metroid Fusion* [42]. Yet, so far these have only been realized as crude approximations of the original player, as they mimic appearance, equipment, basic moves and/or skill sets by relying on heuristic, strategically rigid decision-making.

At the same time, machine learning approaches in video games that harness continuous improvement through simulated play have become popular, e.g. the deep reinforcement learning of *Atari* games [31], temporal difference learning of *Backgammon* [54] or the surpassing of human player performance in the board game that has been rated as not solvable by artificial intelligence methods for a long time; *Go* [48]. The field of player modeling has seen explorations of machine learning techniques for multiple purposes, prominently featuring prediction, classification or analysis [13, 14, 27, 53], to facilitate individual game interpretations, testing or for providing believable opponents. Still, the incorporation of machine learning approaches for generative player modeling for DDA and the resulting player experiences remain under-investigated. Holmgård et al. studied *personas* for player decision modeling [19, 20] that continually observe and adapt to human behavior in order to produce agents with different decision-making styles. These *personas* were realized via evolutionary linear perceptrons and compared to heuristic agents in a test bed 2D dungeon crawler game, resulting in a higher player-rated human-likeness. They also assessed player models when defined as “deviations from theoretically rational actions” in a study of *Super Mario Bros.* [1, 21] and clustered these by means of feature extraction. Using the same game, Ortega et al. [36] imitated human playing styles by means of Neuroevolution and Dynamic Scripting and reached higher scores of human-likeness than performance-directed AI agents, based on subjective judgments. Missura and Gärtner utilized Player Modeling in a 2D test bed shooter via Support Vector Machines acting as predictors for difficulty mismatches and enabling classical DDA parameter tuning based on the results [30]. In previous work, we were successful in showing that player model agents can yield significantly higher motivation compared with heuristic opponents in a short-term online study using the 2D platform fighter *Korona:Nemesis* [9, 39]. Based on these insights, player awareness about substituting individual players with DPBM agents in online multi-player matches was also assessed. In contrast to heuristic bots, DPBM agents turned out to be indistinguishable from their human precursors [37]. In addition, we contrasted different machine learning techniques in a player modeling study of the MMORPG *Lineage 2* [33, 38]. Deep learning offered the best individual prediction accuracy, facilitating the production of playing sessions that closely resemble the original behavior, as well as for differentiating between players. Consequently, we discussed the broader implications for the application of DPBM in: DDA (offering adaptation beyond parameter tuning; training players by exposing them to own strengths and weaknesses), player substitution (bridging online match disruption due to dropouts; providing more individually representative agents), automated game testing (enhancing the estimation of balancing issues by incorporating realistic human player behavior) and cheating detection (revealing behavior that is more likely to stem from undesirable third-party bots rather than players; yielding objective evidence based on behavior in cases of identity theft).

To the best of our knowledge, there is no prior work assessing the experience of players who continuously challenge themselves, where generative player modeling facilitates proficiency progress.

APPROACH

In contrast to the aforementioned studies, the approach presented in this work provides a medium- to long-term situated evaluation. We compare the deployment of player modeling through DPBM with traditional HPT and assess feasibility, approval and motivation in a complex AAA game through a highly ecologically valid field study. To facilitate realistic and generalizable results that avoid artificial laboratory study setting biases [44], we aimed for the deployment of our approach in a real-world setting in a fully fledged game with an existing community of players. In the following, we explain the construction of the recorded training data format, how it was informed by expert interviews, the DPBM architecture, and the study environment.

Expert Interview

In order to gain a more elaborate understanding of viable strategies, decision making and what behavior might lead to different play styles in *Aion*, one of the authors consulted 3 expert players of the game (each with 7-8 years of prior experience) and extracted the most important aspects qualitatively, using brief 1 hour semi-structured interviews over the course of one day. Apart from a less-structured introduction and follow-up discussion after each item, we asked the following questions:

- In a one-on-one situation against a (computer controlled opponent / human player), based on which factors do you decide which skill to use?
- How do you react when you are not able to execute your strategy?
- How would you approach an opponent of the same class that is equally proficient as yourself?

We analyzed the interview using an outcome-oriented structuring content analysis after Mayring [28] and consolidated the most expressive statements about factors that qualify as good indicators for decision making. The most descriptive factors as indicated by the experts are adherence to skill *rotations* and situational *responsive decisions*. Within *rotations*, expert players predominantly apply a specific set of preferred sequences of actions, e.g. ramping up damage by combinations of enhancing and weakening skills or controlling the opponent by a succession of restricting skills. Due to the large amount of possible skills or items to use (cf. Figure 2), these rotations can include complex chains of consecutive skills and/or contain *sub-rotations*. *Responsive decisions* denote the reaction to certain states that the player character or an opponent is in, e.g. healing oneself when hit points are low, removing restricting conditions on the character, increasing or reducing distance between characters or exploiting temporary conditions the opponent is in. They can also trigger more complex situation-specific *rotations*. The description of playstyle aspects by means of *rotations* and *responsive decisions* is not limited to this specific game, but broadly generalizes to the genre of action RPG games. We defined the DPBM state-action architecture on the basis of these factors, including player and target state information as crucial indicators for *responsive decisions* and previous skill information for positioning within *rotations*.

In combination, these factors compose the game state that is fed into the DPBM input layer, while the output layer is trained according to the respective skill that the player used in this situation (cf. Figure 3).

Adaptive Instance Dungeon

Instance dungeons are a major part of MMORPGs, as they can be entered numerous times, solo or in a group, to acquire experience points, equipment, currencies and/or other desirable items. As such, they provide a fertile testing ground for evaluating long-term motivation [52], since most often repeated or even continuous entries are required to reach higher-level goals. To gather expressive evidence of the motivational potential of DPBM, we developed the single-player adaptive instance dungeon *Eternal Challenge* that incorporates both traditional DDA aspects via HPT as well as DPBM. Within the instance, the players encountered various opponents that were clearly distinguishable by their visual appearance (cf Figure 1) and were adjusted through distinct parameters tuned by HPT (cf. Table 1). The underlying proficiency variable λ approximated the player's performance level by being increased whenever he successfully completed *Eternal Challenge* and decreased at the characters death or temporal expiry of the countdown. This way, HPT produces a typical "rubberbanding" effect between player-specific thresholds of lack of challenge and excessive challenge, which is one of the most common ways to enable flow-states to arise in traditional DDA [47] and was constructed by following the inspiration of these approaches [17, 23, 30, 47] in combination with fine-tuning by the developers operating the server to find a range covering too easy, too hard and enough configurability in between for every observed player.

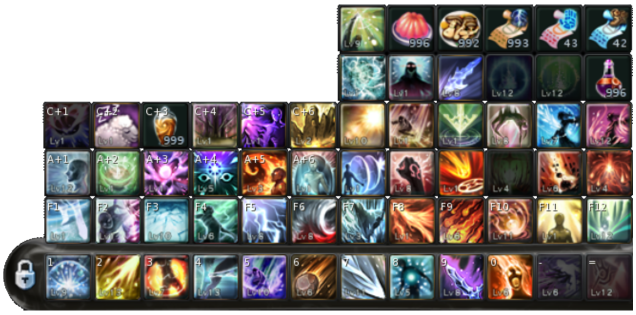


Figure 2. Exemplary arrangement of a subset of skills available to the *Sorcerer* class in *Aion*. Additionally, context-dependent skills (when the player or a target opponent is in a particular condition) and a multitude of items can be activated.

λ	**difficulty level**	Increased whenever *Eternal Challenge* was completed successfully, decreased upon death or timeout.
α	**frequency**	With increased λ, the temporal spawn delay of α opponents was decreased, resulting in an exponential increase of difficulty.
β	**perseverance**	With increased λ, hit points (HP) of β opponents increased, making them harder / more time-consuming to defeat.
γ	**disturbance**	With increased λ, γ opponents used more actions that weaken the player, which decreases survivability, damage performance and increases the tension.
	DPBM	No explicit parameter tuning was used, since network proficiency approximates the player's skill implicitly.

Table 1. DDA mechanisms of *Eternal Challenge*, mapping λ to the difficulty of α, β, γ, while DPBM seeks to emulate the player's behavior.

To avoid incentivizing players to perform badly on purpose, rewards (in the form of experience gained and the level-range of items dropped) were adjusted to be proportional to the difficulty level λ. Upon entering the instance, a 15-minute countdown started that expelled the player if it was not finished after expiration. Within this time limit, the player was expected to destroy a sturdy, non-responsive opponent (β) that spawned additional, hostile enemies over time (α, β, γ) which had to be endured or defeated as well. As soon as the main opponent was defeated, an additional foe that utilizes DPBM appeared in an adjacent room. If the player managed to beat this opponent as well, rewards were distributed and the internal λ level was raised accordingly. λ had no theoretical, but a practical upper limit, since the game inherently restricts reaching damage per second values beyond a certain threshold.

Deep Player Behavior Modeling

When entering *Eternal challenge*, the recorded behavioral data from all preceding runs of the player was retrieved from the underlying database and fed into a feed-forward multi-layer perceptron with backpropagation and a logistic sigmoid activation function (cf. Figure 3), where input and output layer size varied depending on the player's class, skill set and usage ($M = 98.2, SD = 15.1$) input, 5x10 hidden, ($M = 76.2, SD = 15.1$) output nodes). The network was initialized randomly and trained over 1000 epochs, based on the insights of previous work [38, 39] and benchmarks prior to the study that indicated diminishing returns when further increasing the range of parameters. When encountering the DPBM opponent, the trained model was applied generatively to retrieve a set of action probabilities given the occurring state description at real-time. After a weighted choice, the resulting skill was executed, followed by querying the DPBM for the next situation, effectively approximating the learned behavior from the player's preceding battles. As movement was controlled heuristically, temporal-dynamics of behavior are not explicitly modeled, but behavior over time is modeled by focusing the sequencing of skill *rotations* and *responsive decisions* in each occurring state. In terms of the player modeling taxonomy of Yannakakis et al. [58], this implementation realizes a *model-free* (bottom-up) player modeling approach mapping *gameplay data* to actions via *preference learning* and *classification*. According to the player modeling description framework of Smith et al. [49], DPBM directly utilizes *game actions* **(domain)** to *generate* **(purpose)** *individually* **(scope)** modeled behavior by means of *induced* **(source)** training of machine learning techniques.

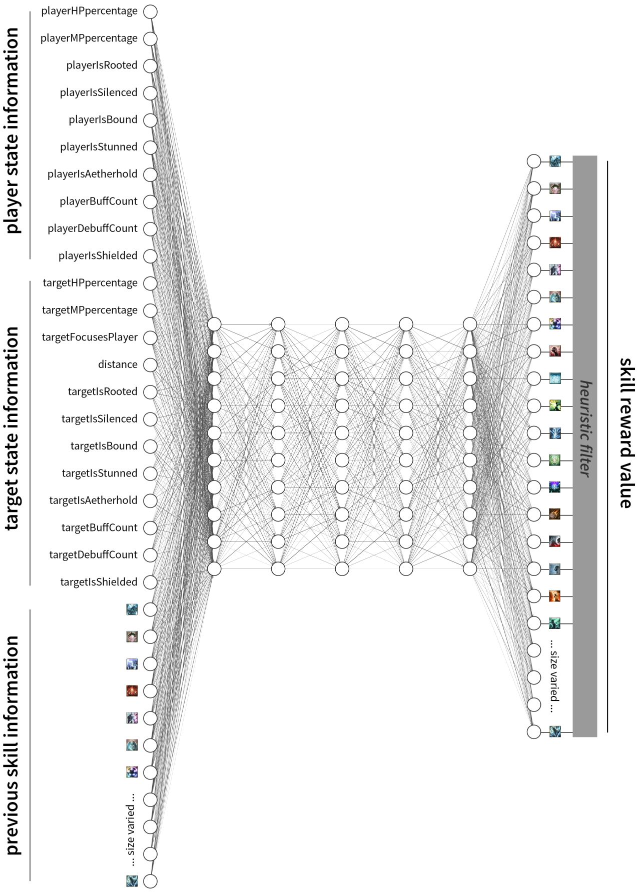


Figure 3. The DPBM architecture mapping game state (information about player, opponent and skill history) to action (skill usage) probabilities. The size of the input and output layers varied depending on the player's class, skill set and usage. The resulting action probability array is filtered heuristically by removing skills that are impossible to execute due to cool-down, MP shortage or other insufficient conditions.

STUDY

In order to evaluate the DDA and intrinsic motivation capabilities of DPBM, we conducted an online field study following a within-subjects design over the course of four weeks. The adaptive instance was published on a private *Aion* game server. To ensure that the measured motivation is attributable to the DPBM approach, we took several precautions. In order to minimize novelty or anticipation bias, we did not announce the existence or concept of the instance prior release and chose a long-term study design. In addition, rewards constitute one of the biggest extrinsic motivators for long-term commitments [56] and thus a potentially high confounding factor when assessing intrinsic motivation. Therefore, the rewards of *Eternal Challenge* were kept conservative and approximated the amount of rewards in other available instances, i.e. players were not able to obtain something that they would not be able to elsewhere and did not acquire a higher reward-to-time-ratio. Findings of Deci et al. [12] also suggest that excessive extrinsic rewards can inhibit intrinsic motivation, the core factor of engagement and enjoyment [45, 41]. Finally, as interplay among players is a major motivating factor of MMORPGs [56], we excluded multi-player situations, leaderboards, high score lists or the publication of ranks and completion times during the study period to avoid complex potential biases in this early situated study. Although MMORPGs are designed to be about multi-player scenarios, occasions of playing solo occur on a regular basis and novel challenge paradigms should arguably be tested in basic, controllable setups before being extended to include additional factors, such as team play or competition.

Measures

For every single *Eternal Challenge* run performed by a player, we recorded state-action data for DPBM training (cf. Figure 3), instance completion times and results (failed or succeeded), as well as the training times and prediction accuracies of the DPBM. In addition, we logged entry counts and timestamps for all available instance dungeons for further activity comparisons. After the study period, a post-study questionnaire asked for player-reported assessments of *perceived competence*, *interest-enjoyment*, *tension-pressure* and *effort-importance*, following the *Intrinsic Motivation Inventory* (IMI) [29], for comparison between the traditional DDA parameters α, β, γ, and the DPBM opponent. Each iteration of the questionnaire was explicitly headed by a display of the appearance of the corresponding opponent in order to assure correctly targeted responses (cf. Figure 1).

Additionally, the survey contained qualitative queries about the appreciation of – and strategies used against – the opponents, the impression of DPBM opponent's behavior in the players' own words, and a free field for additional remarks.

Procedure

The instance dungeon *Eternal Challenge* was introduced and released as part of a regular update to the private server. From then on, players of the community had the opportunity to enter it up to once daily, independent from entering different or additional instances. After four weeks, the recording of in-game data stopped and the post-study questionnaire was advertised on a message board associated with the server.

Participants

During the study period, $(n = 171)$ participants entered *Eternal Challenge* resulting in 776 total instance runs. 30 players (17 men, 13 women (self-identified)) completed the optional post-study questionnaire.

RESULTS

Using a one-way RM ANOVA, we found significant effects for the IMI scores *perceived competence*, *interest-enjoyment*, *tension-pressure* and *effort-importance* between conditions α, β, γ and DPBM (cf. Table 2).

These outcomes were further evaluated using two-tailed paired t-tests (cf. Table 3, Figure 4). Employing conservative Bonferroni correction, *p*-values were multiplied with the amount of repeated comparisons. DPBM received significantly higher scores of *interest-enjoyment* and *effort-importance* compared to all HPT opponents, resulting in mostly large effect sizes after Cohen. It also outperformed α and β significantly in terms of *tension-pressure* with medium effect sizes.

perceived competence	$F(3,26) = 3.59$	$p < .05$	$\eta^2 = 0.29$
interest-enjoyment	$F(3,26) = 8.75$	$p < .01$	$\eta^2 = 0.5$
tension-pressure	$F(3,26) = 3.37$	$p < .05$	$\eta^2 = 0.28$
effort-importance	$F(3,26) = 5.63$	$p < .01$	$\eta^2 = 0.39$

Table 2. ANOVA results ($F(df1,df2)-$, $p-$values, η^2 for effect size) of the *Intrinsic Motivation Inventory* between the different opponents.

On average, players spent $(M = 7.71, SD = 2.49)$ minutes in the instance and used up to 91 $(M = 21.1, SD = 11.6)$ different skills against the DPBM opponent. Model training times lasted $(M = 2018, SD = 3692)$ ms per session with $(M = 8.75, SD = 3.34)$ ms per recorded skill.

To assess the objective quality of the underlying machine learning model and render it comparable to related approaches, 80% of the data recorded until any given time of entry into the instance was used for training, whereas 20% served for a routine initial test, resulting in $(M = 60.64, SD = 22.57)$% prediction accuracy.

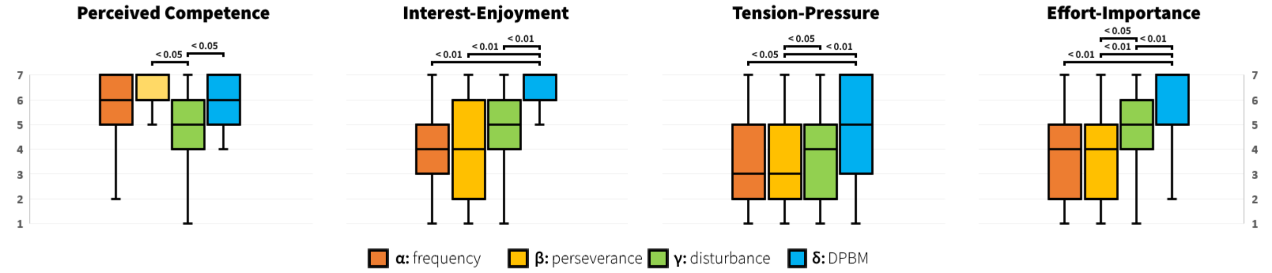


Figure 4. Intrinsic motivation inventory (IMI) results for the compared DDA variables. Includes medians (center marks), standard deviations (boxes), minimal and maximal values (whiskers) and significant difference markers.

Compared to all 35 available instances in the game, *Eternal Challenge* (EC) became the most popular instance by daily numbers of players over the duration of the study (cf. Figure 5), as chi-square goodness of fit tests show (cf. Table 4), assuming equal proportions. Even when omitting the first quarter to counteract a presumable novelty bias in the distinct initial spike, EC still outmatched all alternatives.

interest-enjoyment	*DPBM* ($M = 6.17$, $SD = 1.11$)	**effort-importance**	*DPBM* ($M = 5.87$, $SD = 1.63$)
α ($M = 4$, $SD = 1.51$)	$p = .000$ $d = 1.64$	α ($M = 4.04$, $SD = 2.03$)	$p = .006$ $d = .99$
β ($M = 4.04$, $SD = 2.06$)	$p = .001$ $d = 1.23$	β ($M = 3.91$, $SD = 1.83$)	$p = .000$ $d = 1.13$
γ ($M = 4.78$, $SD = 1.76$)	$p = .01$ $d = .95$	γ ($M = 4.91$, $SD = 1.81$)	$p = .004$ $d = .56$

tension-pressure	*DPBM* ($M = 4.91$, $SD = 2.15$)	**perceived competence**	*DPBM* ($M = 5.83$, $SD = 1.07$)
α ($M = 3.39$, $SD = 1.8$)	$p = .049$ $d = .77$	α ($M = 5.65$, $SD = 1.67$)	$p > .05$
β ($M = 3.39$, $SD = 1.8$)	$p = .007$ $d = .77$	β ($M = 5.91$, $SD = 1.35$)	$p > .05$
γ ($M = 4.17$, $SD = 1.85$)	$p > .05$	γ ($M = 4.65$, $SD = 1.72$)	$p > .05$

Table 3. Means, standard deviations and significant t-test results after Bonferroni correction ($p-$values and Cohen's d for effect size, $df = 29$) of the *Intrinsic Motivation Inventory* between the different opponents.

For the qualitative remarks, we used a structuring content analysis after Mayring [28] to assess the effect of challenging the DPBM opponent. Players were asked to state their general impression and opinion freely, without confounding or influencing questions. From the utilizable statements, 31.8% describe an appropriate challenge (e.g. *"quite easy at first but afterwards I really was busy thinking about how I approach him"*, *"it's almost as good as I am"*), while 9.1% depict it as slightly too high or slightly to low. 13.6% emphasize a notable entertaining factor, whereas 4.4% declare that this encounter did not appeal to them. Although the behavior or decision-making of the DPBM opponent was never explicitly stated or explained during the study, 36.4% of players ascribed the ability to learn from previous battles and the adaptation to the player's own behavior, combos, rotations and/or strategies to their enemy (e.g. *"at first he randomly used skills that I also used, later he added my combos"*, *"tried to replicate my own skills and techniques"*, *"it was hilarious when I played against myself"*).

During complete study period:		
EC vs. #2	$\chi^2(1, n = 1007) = 58.64$	$p < 0.01$
EC vs. #3	$\chi^2(1, n = 902) = 134.26$	$p < 0.01$
EC vs. #4	$\chi^2(1, n = 856) = 181.35$	$p < 0.01$
After first quarter of study period:		
EC vs. #2	$\chi^2(1, n = 627) = 4.48$	$p = 0.03$
EC vs. #3	$\chi^2(1, n = 526) = 45.09$	$p < 0.01$
EC vs. #4	$\chi^2(1, n = 493) = 70.93$	$p < 0.01$

Table 4. Chi-square goodness of fit tests between *Eternal Challenge* and the second, third and fourth most popular instance. Less popular instances have shown similarly significant results, but have been omitted for the sake of readability.

DISCUSSION

Our results indicate distinct effects on approval and intrinsic motivation for DPBM opponents, as well as effects on long-term commitment for the presence of DDA in general. We were successful in evidencing significantly higher motivation for players to enter adaptive instance dungeons compared to static alternatives over considerable duration of successive play sessions and report indications that DPBM attributes significantly more to this preference than traditional DDA parameter tuning.

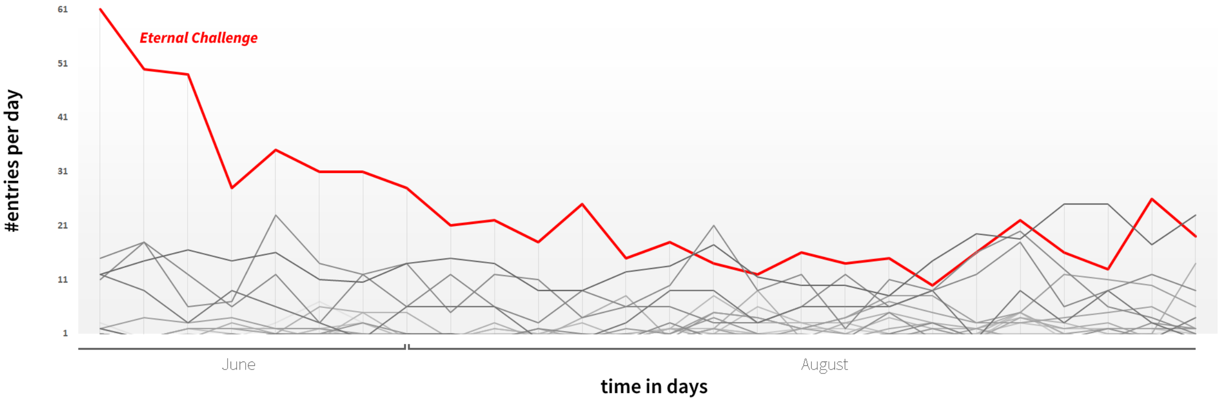


Figure 5. Daily number of unique players entering *Eternal Challenge* compared to all other available instances during the study period.

This insight is based on notably high absolute IMI scores and significant differences compared to conventional DDA techniques. The outcome that DPBM outperformed HPT in terms of *interest-enjoyment* indicates a high "fun factor", while *tension-pressure* and *effort-importance* highlight the considerable challenge, leading to an overall higher intrinsic motivation and linked potential to induce flow. The actual implicit DDA capabilites of DPBM are backed by qualitative statements that reveal an appropriate challenge, a noticeable difficulty adjustment over time and the perception of playing against an equal opponent that facilitates rethinking of habitual behavior. This work also demonstrates the technical applicability of large-scale, long-term generative player modeling with reasonable training times and accuracies.

Overall, our work provides quantitative and qualitative empirical evidence supporting our initial hypotheses about facilitating long-term motivation potential, capabilities for enabling DDA and individual representation, indicating the following responses to the respective research questions:

- **We measured a substantially and consistent motivation achieved with the support of DPBM over the medium- to long-term.**
- **The measured intrinsic motivation of challenging a DPBM-fuelled opponent significantly exceeded traditional HPT.**
- **Players perceive behavior from DPBM as representative of – or comparable to – their own decision-making.**

Limitations and Future Work

The mixed-bag fashion of the instance, which resulted from aiming to maintain an ecologically-valid realistic instance design, results in a combined experience of HPT and DPBM opponents that might influence the participants' assertions. This study design was selected due to the long-term period of the study in a community where players know each other, rendering a between-subjects design confounding, since players would have exchanged views about the different conditions and/or complained about unjust treatments.

To further corroborate evidence and to gain a clear comparison between the different HPT factors and DPBM, the experiment should be replicated to manifest a control group (mutually exclusive from this player base) in which no DPBM (or HPT) is present. Apart from that, the Intrinsic Motivation Inventory was designed to measure single sessions within experiments. While it was not explicitly developed for this study's setup, we found it to be the most appropriate questionnaire to assess motivation, as there is no validated reflective long-term motivation questionnaire that does not have to be raised after every single session (which was omitted in favor of ecological validity).

Based on our achievements and outcomes, we are looking forward to extending the scope of using DPBM in video games to enabling personalized, adaptive challenges that go beyond one-on-one situations to encompass interactions between different players and consider both competitive as well as cooperative interplay. DPBM agents could be deployed in multi-player scenarios where groups are challenged to deal with effects between player modeled opponents or utilized to support team-fights between human players, as equivalent reinforcements. Additionally, we plan to construct a one-dimensional proficiency metric that maps DPBM configurations to estimated competence in a game, in order to offer players more unique DDA encounters stemming from different players with similar proficiency. Using the considerably large data set recorded in this study, we seek to benchmark several alternative machine learning techniques as core mechanisms for the underlying player modeling (e.g. recurrent, deep belief, GAN or context-driven LSTM networks), to be able to give practical statements about applicability concerning temporal requirements and resulting accuracy. Furthermore, we envision DPBM as an effective instrument for elevating autonomous game testing and balancing, since actual, precise player behavior can be simulated, and temporary substitutions or continuations of disconnected players in online matches can be facilitated to minimize game experience disruptions.

CONCLUSION

We presented the design and implementation of an adaptive instance dungeon in the MMORPG *Aion* to evaluate a novel, implicit take on *Dynamic Difficulty Adjustment* that is not dependent on manually composed parameter tuning, but affords a continually adapting challenge through *Deep Player Behavior Modeling*. In an extensive medium- to long-term study ($n = 171$ over the course of four weeks) we contrasted an opponent applying DPBM to traditional DDA parameter tuning and can report significantly higher intrinsic motivation stemming from the unique game experience of being confronted with strategic behaviors that mirror one's own patterns. Qualitative statements reinforce the approval and positive experience of DPBM, while the consistent and dominant usage of the instance throughout the whole study period reflects its potential to elevate long-term motivation and commitment. Regarding the technical applicability of the approach, we report on the DPBM architecture, its accuracy and data structure and give an estimation about the temporal demand, yielding real-time potential.

According to the guidelines of transparent statistics, the collected data of this approach, as well as its implementation, will be made openly available upon publication, using an open-access repository.

ACKNOWLEDGMENTS

We would like to thank all participants, testers, *Beyond Aion* for hosting the instance dungeon and Philipp Krüger for his collaboration, advice and porting. This work was funded by the German Research Foundation (DFG) as part of Collaborative Research Center (SFB) 1320 EASE - Everyday Activity Science and Engineering, University of Bremen (http://www.easecrc.org/), subproject H2.

REFERENCES

[1] Nintendo Research Development 4. 1985. *Super Mario Bros.* Game [NES]. (13 September 1985). Nintendo, Kyōto, Japan.

[2] Ernest Adams. 2002. Balancing Games with Positive Feedback. *Gamasutra. com, January* 4 (2002).

[3] Dennis Ang and Alex Mitchell. 2017. Comparing Effects of Dynamic Difficulty Adjustment Systems on Video Game Experience. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. ACM, 317–327.

[4] Dennis Ang and Alex Mitchell. 2019. Representation and Frequency of Player Choice in Player-Oriented Dynamic Difficulty Adjustment Systems. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. ACM, 589–600.

[5] ArenaNet. 2005. *Guild Wars*. Game [PC]. (28 April 2005). ArenaNet, Bellevue, WA. Played 2019.

[6] BioWare. 2012. *Mass Effect 3*. Game [PC,XBox360,PS3,WiiU]. (6 March 2012). BioWare, Edmonton, Kanada.

[7] Capcom Production Studio 4. 2005. *Resident Evil 4*. Game [Gamecube]. (11 January 2005).

[8] Thomas Constant and Guillaume Levieux. 2019. Dynamic Difficulty Adjustment Impact on Players' Confidence. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, 463.

[9] Nevermind Creations. 2019. *Korona:Nemesis*. Game [PC]. (18 August 2019).

[10] Mihaly Csikszentmihalyi. 2013. *Flow: The psychology of happiness*. Random House.

[11] Thomas Debeauvais. 2016. *Challenge and retention in games*. Ph.D. Dissertation. UC Irvine.

[12] Edward L Deci, Richard Koestner, and Richard M Ryan. 1999. A meta-analytic review of experiments examining the effects of extrinsic rewards on intrinsic motivation. *Psychological bulletin* 125, 6 (1999), 627.

[13] Anders Drachen, Alessandro Canossa, and Georgios N Yannakakis. 2009. Player modeling using self-organization in Tomb Raider: Underworld. In *2009 IEEE symposium on computational intelligence and games*. IEEE, 1–8.

[14] Anders Drachen, Rafet Sifa, Christian Bauckhage, and Christian Thurau. 2012. Guns, swords and data: Clustering of player behavior in computer games in the wild. In *2012 IEEE conference on Computational Intelligence and Games (CIG)*. IEEE, 163–170.

[15] Nintendo EAD. 1987. *Zelda II: The Adventure of Link*. Game [NES]. (14 January 1987). Nintendo EAD, Kyoto, Japan.

[16] Nintendo EAD. 2014. *Mario Kart 8*. Game [WiiU,Switch]. (29 May 2014). Nintendo EAD, Kyoto, Japan. Played 2019.

[17] William Rao Fernandes and Guillaume Levieux. 2019. δ-logit: Dynamic Difficulty Adjustment Using Few Data Points. In *Joint International Conference on Entertainment Computing and Serious Games*. Springer, 158–171.

[18] Julian Frommel, Fabian Fischbach, Katja Rogers, and Michael Weber. 2018. Emotion-based Dynamic Difficulty Adjustment Using Parameterized Difficulty and Self-Reports of Emotion. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*. ACM, 163–171.

[19] Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N Yannakakis. 2014a. Evolving personas for player decision modeling. In *2014 IEEE Conference on Computational Intelligence and Games*. IEEE, 1–8.

[20] Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N Yannakakis. 2014b. Generative agents for player decision modeling in games.. In *FDG*. Citeseer.

[21] Christoffer Holmgård, Julian Togelius, and Georgios N Yannakakis. 2013. Decision making styles as deviation from rational action: A super mario case study. In *Ninth Artificial Intelligence and Interactive Digital Entertainment Conference*.

[22] Dayana Hristova. 2017. Dynamic difficulty adjustment (DDA) in first person shooter (FPS) games. (2017).

[23] Robin Hunicke. 2005. The case for dynamic difficulty adjustment in games. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*. ACM, 429–433.

[24] Robin Hunicke and Vernell Chapman. 2004. AI for Dynamic Difficulty Adjustment in Games.

[25] Changchun Liu, Pramila Agrawal, Nilanjan Sarkar, and Shuo Chen. 2009. Dynamic difficulty adjustment in computer games through real-time anxiety-based affective feedback. *Int. Jrnl. of Human-Computer Interaction* 25, 6 (2009), 506–529.

[26] Ricardo Lopes, Ken Hilf, Luke Jayapalan, and Rafael Bidarra. 2013. Mobile adaptive procedural content generation. In *Proceedings of the fourth workshop on Procedural Content Generation in Games (PCG 2013), Chania, Crete, Greece*.

[27] Tobias Mahlmann, Anders Drachen, Julian Togelius, Alessandro Canossa, and Georgios N Yannakakis. 2010. Predicting player behavior in tomb raider: Underworld. In *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games*. IEEE, 178–185.

[28] Philipp Mayring. 2010. Qualitative inhaltsanalyse. In *Handbuch qualitative Forschung in der Psychologie*. Springer, 601–613.

[29] Edward McAuley, Terry Duncan, and Vance V Tammen. 1989. Psychometric properties of the Intrinsic Motivation Inventory in a competitive sport setting: A confirmatory factor analysis. *Research quarterly for exercise and sport* 60, 1 (1989), 48–58.

[30] Olana Missura and Thomas Gärtner. 2009. Player Modeling for Intelligent Difficulty Adjustment. In *Discovery Science*, João Gama, Vítor Santos Costa, Alípio Mário Jorge, and Pavel B. Brazdil (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 197–211.

[31] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, and others. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (2015), 529.

[32] Fausto Mourato, Fernando Birra, and Manuel Próspero dos Santos. 2014. Difficulty in action based challenges: success prediction, players' strategies and profiling. In *Proceedings of the 11th Conference on Advances in Computer Entertainment Technology*. ACM, 9.

[33] NCsoft. 2003. *Lineage 2*. Game [PC]. (1 October 2003). NCSoft, Seongnam, South Korea.

[34] NCsoft. 2008. *Aion*. Game [PC]. (25 September 2008). NCSoft, Seongnam, South Korea. Played August 2019.

[35] Pedro A Nogueira, Vasco Torres, Rui Rodrigues, Eugénio Oliveira, and Lennart E Nacke. 2016. Vanishing scares: biofeedback modulation of affective player experiences in a procedural horror game. *Journal on Multimodal User Interfaces* 10, 1 (2016), 31–62.

[36] Juan Ortega, Noor Shaker, Julian Togelius, and Georgios N Yannakakis. 2013. Imitating human playing styles in super mario bros. *Entertainment Computing* 4, 2 (2013), 93–104.

[37] Johannes Pfau, Jan David Smeddinck, Ioannis Bikas, and Rainer Malaka. 2020. Bot or not? User Perceptions of Player Substitution with Deep Player Behavior Models. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM.

[38] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2018. Towards Deep Player Behavior Models in MMORPGs. In *Annual Symp. on Computer-Human Interaction in Play Ext. Abstracts (CHI PLAY '18)*. ACM, New York, NY, USA, 381–92.

[39] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2019. Deep Player Behavior Models: Evaluating a Novel Take on Dynamic Difficulty Adjustment. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, LBW0171.

[40] Mike Preuss, Thomas Pfeiffer, Vanessa Volz, and Nicolas Pflanzl. 2018. Integrated Balancing of an RTS Game: Case Study and Toolbox Refinement. In *2018 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 1–8.

[41] Andrew K Przybylski, C Scott Rigby, and Richard M Ryan. 2010. A motivational model of video game engagement. *Review of general psychology* 14, 2 (2010), 154–166.

[42] Nintendo RD1. 2002. *Metroid Fusion*. Game [GBA]. (18 November 2002). Nintendo RD1, Kyoto, Japan.

[43] Andrew Rollings and Ernest Adams. 2003. *Andrew Rollings and Ernest Adams on game design*. New Riders.

[44] Robert Rosenthal and Kermit L Fode. 1963. The effect of experimenter bias on the performance of the albino rat. *Behavioral Science* 8, 3 (1963), 183–189.

[45] Richard M Ryan and Edward L Deci. 2000. Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary educational psychology* 25, 1 (2000), 54–67.

[46] Lingdao Sha, Souju He, Junping Wang, Jiajian Yang, Yuan Gao, Yidan Zhang, and Xinrui Yu. 2010. Creating appropriate challenge level game opponent by the use of dynamic difficulty adjustment. In *2010 Sixth International Conference on Natural Computation*, Vol. 8. IEEE, 3897–3901.

[47] Noor Shaker, Julian Togelius, and Georgios N Yannakakis. 2016. The experience-driven perspective. In *Procedural Content Generation in Games*. Springer, 181–194.

[48] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, and Laurent Sifre et al. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529 (2016), 484–489.

[49] Adam M. Smith, Chris Lewis, Kenneth Hullet, Gillian Smith, and Anne Sullivan. 2011. An Inclusive View of Player Modeling. In *Proceedings of the 6th International Conference on Foundations of Digital Games (FDG '11)*. ACM, New York, NY, USA, 301–303.

[50] David Stammer, Tobias Günther, and Mike Preuss. 2015. Player-adaptive spelunky level generation. In *2015 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 130–137.

[51] Alexander Streicher and Jan D. Smeddinck. 2016. Personalized and Adaptive Serious Games. In *Entertainment Computing and Serious Games*, Ralf Dörner, Stefan Göbel, Michael Kickmeier-Rust, Maic Masuch, and Katharina Zweig (Eds.). Lecture Notes in Computer Science, Vol. 9970. Springer International Publishing, Cham, 332–377.

[52] Mirko Suznjevic and Maja Matijasevic. 2010. Why MMORPG players do what they do: relating motivations to action categories. *International Journal of Advanced Media and Communication* 4, 4 (2010), 405–424.

[53] Marco Tamassia, William Raffe, Rafet Sifa, Anders Drachen, Fabio Zambetta, and Michael Hitchens. 2016. Predicting player churn in destiny: A hidden markov models approach to predicting player departure in a major online game. In *2016 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 1–8.

[54] Gerald Tesauro. 1994. TD-Gammon, a self-teaching backgammon program, achieves master-level play. *Neural computation* 6, 2 (1994), 215–219.

[55] Valve. 2008. *Left 4 Dead*. Game [PC]. (18 November 2008). Valve, Bellevue, WA, USA. Played 2017.

[56] Hao Wang and Chuen-Tsai Sun. 2011. Game reward systems: Gaming experiences and social meanings.. In *DiGRA Conference*, Vol. 114.

[57] Su Xue, Meng Wu, John Kolen, Navid Aghdaie, and Kazi A Zaman. 2017. Dynamic difficulty adjustment for maximized engagement in digital games. In *Proceedings of the 26th International Conference on World Wide Web Companion*. International World Wide Web Conferences Steering Committee, 465–471.

[58] Georgios N Yannakakis, Pieter Spronck, Daniele Loiacono, and Elisabeth André. 2013. Player modeling. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Dungeons & Replicants: Automated Game Balancing via Deep Player Behavior Modeling

Johannes Pfau
Digital Media Lab
University of Bremen
Bremen, Germany
jpfau@tzi.de

Antonios Liapis
Institute of Digital Games
University of Malta
Msida, Malta
antonios.liapis@um.edu.mt

Georg Volkmar
Digital Media Lab
University of Bremen
Bremen, Germany
gvolkmar@tzi.de

Georgios N. Yannakakis
Institute of Digital Games
University of Malta
Msida, Malta
georgios.yannakakis@um.edu.mt

Rainer Malaka
Digital Media Lab
University of Bremen
Bremen, Germany
malaka@tzi.de

Abstract—**Balancing the options available to players in a way that ensures rich variety and viability is a vital factor for the success of any video game, and particularly competitive multiplayer games. Traditionally, this balancing act requires extensive periods of expert analysis, play testing and debates. While automated gameplay is able to predict outcomes of parameter changes, current approaches mainly rely on heuristic or optimal strategies to generate agent behavior. In this paper, we demonstrate the use of *deep player behavior models* to represent a player population ($n = 213$) of the massively multiplayer online role-playing game *Aion*, which are used, in turn, to generate individual agent behaviors. Results demonstrate significant balance differences in opposing enemy encounters and show how these can be regulated. Moreover, the analytic methods proposed are applied to identify the balance relationships between classes when fighting against each other, reflecting the original developers' design.**

Index Terms—**Automated game testing, balancing, deep learning, generative player modeling, imitation learning, video games**

I. Introduction

Due to its steady growth in popularity and accessibility, the video game industry has evolved to a multi-billion dollar branch that surpassed all other entertainment industry sectors including TV, cinema and music[1]. Along with this development, player demands for content and mechanics are ramping up to extents that even large companies struggle to manage [1]. Next to core content production, the majority of computational and labour effort is put on the detection of gameplay and experience bugs (e.g., 80% of the 50 most popular games on the major distribution platform *Steam*[2] require critical updates after launch [2]). While automated routines for the detection and reporting of critical errors and solvability become more popular in the industry [3], [4], balancing remains one of the most difficult and time-consuming phases of the game design process. The availability of versatile in-game units, character classes, factions or roles between which players are able to choose from has become indispensable for many successful titles, yet the balancing of these appears to open up an incessant effort. Even prominent titles of competitive online games that launched years ago still undergo persistent balance patches (e.g. StarCraft II [5], Overwatch [6] or Guild Wars 2 [7]). Following the definition of Sirlin [8], a game is *"balanced if a reasonably large number of options available to the player are viable"* (where viability sets the requirement of having many meaningful choices throughout a game), while *"players of equal skill should have an equal chance at winning"*. Together with frequently desired asymmetrical configuration possibilities of these options, this inherently leads to combinatorial explosions, which can become hazardous for the enjoyability of the game and the satisfaction of its players. Even worse, Hullett et al. highlight that balancing issues most of the time *"only become apparent after many months of play"* [9]. Compared to straightforward fixable bugs, glitches and solvability aspects, the trouble with balancing issues is that they do not only appear during the launch of a newly published game. Instead, balancing is an ongoing, repetitive task that is heavily influenced by the perceptions of the player community: *"after each patch, often the discussion begins again, factoring in new balancing or abilities for each class"* [10]. In the game industry, balancing is most often approached through long-term expert analysis, excessive human play-testing, and persistent debates with the community. Meanwhile, recent applied machine learning techniques have become very successful in outperforming human capabilities of playing, e.g. in Atari games [11], classical board games such as chess, shōgi and Go [12], [13] or real-time strategy games (RTS) such as StarCraft II [14]. While computer-controlled agents employing these approaches might also be suitable for automated game testing, their utility for automated balancing is arguably limited, given that optimal or super-human proficiency is not representative for the population of human players the game should be tailored for [15].

In this paper, we apply *Deep Player Behavior Modeling* (DPBM) [16] to automated game balancing. Within DPBM, individual decision making from game states is mapped to a preference distribution of actions via machine learning, approximating the replication of individual players. In contrast

This work was funded by the German Research Foundation (DFG) as part of Collaborative Research Center (SFB) 1320 EASE - Everyday Activity Science and Engineering, University of Bremen (http://www.easecrc.org/), subproject H2.

[1] https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/

[2] https://store.steampowered.com/

978-1-7281-4533-4/20/$31.00 ©2020 IEEE

to optimal or generalized models, the DPBM approach allows for the consideration of many (potentially viable) playing styles that players can employ instead of reducing it to a global decision making module. In previous work, DPBM showed to be successful in generating agents capable of offering challenges on the same proficiency level [17] and convinced other players that they replicated individual behavior believably [18].

In this work, we used a dataset of the popular massively multiplayer online role-playing game (MMORPG) *Aion* [19] consisting of atomic decision making that was recorded throughout 6 months and 213 players in one-versus-one situations [17]. From this, we generated DPBM-driven agents for all players and benchmarked their proficiency against heuristic NPCs in a two-dimensional study setup that manipulated the offensive and defensive capabilities of the latter. While that study gave initial insights about the basic versatility of classes in player versus environment (PvE) settings, a subsequent investigation examined how all player replicas playing against each other in a player versus player (PvP) situation. For the empirical assessment of the resulting proficiencies, we utilized a metric that approximates the quality of single benchmark performances in terms of effectiveness and efficiency. Evaluating the capabilities for automated game balancing and individual proficiency estimation, we aim to answer the following research questions:

- *Can imbalances between in-game classes be detected through generative player modeling with respect to PvE and PvP?*
- *Can generative player modeling elevate automated game testing to turn design specifications into optimized parameter constellations?*

We hypothesize that agents that are representative of individual players' decision making are able to detect differences in performance between classes and resemble the population closer compared to generalized or random agents. Under these conditions, DPBM should provide a viable technique to map behavioral patterns to proficiency scores and to inform automated game balancing empirically. This work contributes to games user research and game development in academia and industry by introducing a novel technique capable of enhancing game testing processes with the potential of reducing the associated effort. In addition, a proficiency metric is constructed and presented that allows for the comparison of benchmark results. Effectively, this indicates the added value of assessing the replicated player population against generalized or random models.

II. Related Work

The implementation of automatic simulations of video game play has become a viable and efficient alternative or improvement to tedious and non-exhaustive human testing for the purpose of finding critical errors, solvability investigations or parameter tuning. The majority of scientific approaches focuses on detecting logical bugs or game crashes, such as Radomski et al. [20] or Varvaressos et al. [21] who identified violations of manually defined constraints via simulated play. Buhl et al. [3] highlight the utility of autonomous testing routines in everyday continuous integration and continuous delivery pipelines by contrasting the amount of encountered bugs against previous developments without them. Zheng et al. [22] designed a game playing agent utilizing deep reinforcement learning, while Chan et al. [23] made use of a neuroevolution approach that on top of playing was able to report on the constellation and sequence of actions that lead to game malfunctions. Furthermore, Bécares et al. [24] mapped human tester playthrough records to semantic replay models using Petri nets and Iftikhar et al. [25] and Schaefer et al. [26] introduced frameworks for autonomously testing generic games of the platformer or puzzle genre, respectively.

A number of studies tackle solvability, such as those of Powley et al. [27], Shaker et al. [28] or Volkmar et al. [29] that aided the level design of (procedurally generated) games by assuring potential solutions are feasible. Schatten et al. [30] simulated large-scale dynamic agent systems to test quest solvability in MMORPGs. Within the scope of point-and-click adventure games, Pfau et al. [4] established a generic adventure solver traversing these via reinforcement learning and reporting crashes, dead-ends and performance issues. Van Kreveld et al. [31] and Southey et al. [32] assessed difficulty or interestingness approximations of levels or mechanics by machine learning of descriptive in-game metrics.

Regarding balancing, scientific approaches often build on simulations that iteratively assess balance criteria and dynamically tune in-game parameters based on the former. Jaffe et al. [33], García-Sanchez et al. [34] and De Mesentier Silva et al. [35] applied this paradigm to board or card games, which was amplified by Mahlmann et al. [36] by introducing procedurally generated cards on top of these simulations. In other genres, Beau and Bakkes [37] utilized Monte-Carlo Tree Search for balancing units of Tower Defense games, Morosan and Poli [38] tweaked difficulty specifications in RTS and Arcade games after neuroevolution agents assessed these and Leigh et al. [39] dynamically balanced strategies though the coevolution of two competing agents playing a Capture The Flag game.

Closely related to the approach outlined in this paper, Holmgård et al. [40] conflated atomic player behavior into procedural personas to simulate and test different play styles in a Dungeon Crawler game and Gudmundsson et al. [41] utilized atomic choices in order to predict the difficulty of various levels of a Match-3-Puzzle game. Nonetheless, even if some approaches process some kind of human player input, incorporating actual information about *individual* and *atomic* player behavior has not been tackled yet. Generative player modeling has the potential to fuse automatic simulation methods with behavioral information, giving the developers the opportunity to receive practically immediate insights on which player strategies are popular, dominant and/or may require rework. Further generative player modeling is able to inform developers on how parameter tuning will likely alter the outcome of strategies before presenting it to the community, how implemented dynamic difficulty approaches

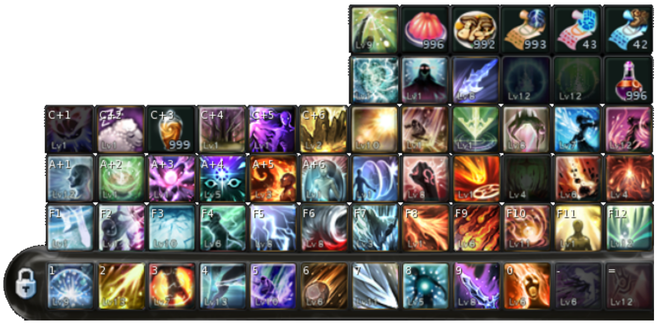


Fig. 1. Exemplary arrangement of a subset of skills available to the *Sorcerer* class in *Aion*. Additionally, context-dependent skills (when the player or a target opponent is in a particular condition) can be activated.

can be informed about parameter thresholds, and how to automatically balance game mechanics after large-scale permutations of classes, setups, parameters and behavior in all stages of development.

III. Approach

This section details our decisions for the selected game environment, the recorded data structure and the modeling approach.

A. Game Environment

To select a representative game within a genre that considerably suffers from the aforementioned balancing issues, we chose the MMORPG *Aion* in which a typical set of in-game classes is available. *Melee* classes (Gladiator, Templar, Assassin) mainly deal close-combat damage, in contrast to *Magic* classes (Sorcerer, Spiritmaster, Gunner) or Rangers. *Heal* classes (Cleric, Bard) deal less damage but offer additional support, while Chanters excel at the latter. Even if many in-game situations involve multi-player constellations, all classes are able to perform on their own in principle. Combat is mainly fought out by activating skill actions that harm the opponent(s) and/or benefit the player character (cf. Fig. 1). Depending on the sequencing of these skills and their contextual usage, individual players execute diverse strategies. Even if these strategies rarely maximize efficiency, they resemble situational preferences that emerge in personal play styles, such as improving own offensive or defensive capabilities or leading to maintained control over the opponent.

B. Dataset and Structure

Publicly accessible datasets that comprise vast proportions of recorded real-world player information are found in several instances, yet all of these third-party data providers offer only publicly available statistical meta-data describing high-level behavioral data. Even with the information about which actions are used in which frequencies, no knowledge is contained about the contextual game state during these action decisions, which, in turn, limits the expressiveness of the eventual generative agent. In contrast, we implement a state-action architecture mapping contextual information to individual player's decision making (indicated in Fig. 2 as input and output). Over the course of 6 months, 213 players with considerable prior expertise of *Aion* were recorded within a daily single-player dungeon instance in considerably challenging one-versus-one combat situations [17]. Table I provides the number of players in the dataset for each class.

C. Deep Player Behavior Modeling

DPBM realizes individual generative player modeling by assessing atomic player behavior in a state-action architecture and establishes a mapping among these via machine learning [16]. For generating a replicative agent that is representative of a single individual, the recorded behavioral data from all relevant observations was retrieved from the underlying database and fed into a feed-forward Multilayer perceptron (MLP) with backpropagation and a logistic sigmoid activation

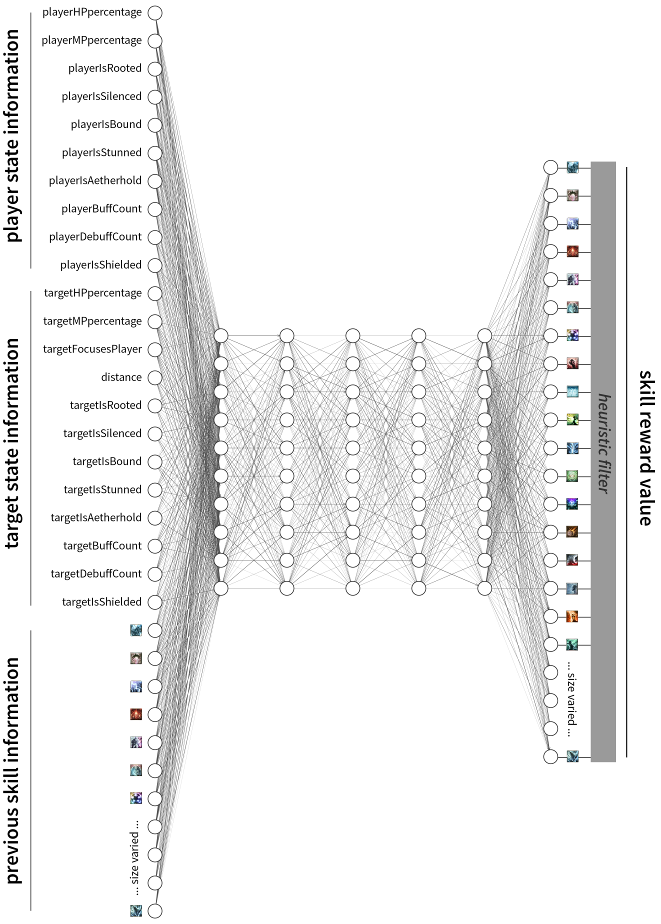


Fig. 2. The DPBM architecture mapping game state (information about player, opponent and preceding skill) to action (skill usage) probabilities. Design decisions can be found in [17]. The size of the input and output layers varied depending on the player's class, skill set and usage. The resulting action probability array is filtered heuristically by removing skills that are impossible to execute due to cool-down, MP shortage or other insufficient conditions.

TABLE I
PLAYER COUNT OF EACH *class* (AND **ARCHETYPE**) IN THE DATASET.

MELEE		**MAGIC**		**RANGED**		**HEAL**	
33	*Gladiator*	20	*Sorcerer*	13	*Ranger*	25	*Cleric*
19	*Templar*	18	*Spiritmaster*	**SUPPORT**		39	*Bard*
17	*Assassin*	10	*Gunner*	19	*Chanter*		

TABLE II
RECORDED NUMBER OF DIFFERENT SKILLS IN EACH CLASS OF *Aion*.

MELEE		**MAGIC**		**RANGED**		**HEAL**	
78	*Gladiator*	52	*Sorcerer*	53	*Ranger*	48	*Cleric*
56	*Templar*	51	*Spiritmaster*	**SUPPORT**		78	*Bard*
57	*Assassin*	42	*Gunner*	57	*Chanter*		

function. The input layer consisted of 22 nodes describing the contextual game state plus a set of nodes representing the preceding skill. Consisting of the same set of skill nodes, the output layer characterizes the probability distribution of action choices with respect to the individual player and the input situation (cf. Fig. 2). The sizes of the skill sets varied per class, as shown in Table II.

The network was initialized randomly and contained 4 hidden layers with equal size to the input layer. It was trained over 1000 epochs, based on insights from previous work [16]–[18], [42]; benchmarks prior to the study also indicated diminishing returns when further increasing the range of parameters.

When exposed to the testing environment, the trained model was applied generatively to retrieve a set of action probabilities given the occurring state description at real-time. After a weighted choice, the resulting skill was executed, followed by querying the DPBM for the next situation, effectively approximating the learned behavior from the original player's battles. Based on the player modeling taxonomy of Yannakakis et al. [15], [43], this implementation realizes a *model-free* (bottom-up) player modeling approach mapping *gameplay data* to actions via *classification*. According to the player modeling description framework of Smith et al. [44], DPBM directly utilizes *game actions* (domain) to *generate* (purpose) *individually* (scope) modeled behavior by means of *induced* (source) training of machine learning techniques.

D. Proficiency Metric

To estimate balance discrepancies between classes we construct a proficiency metric that assesses the quality of an agent's performance during evaluation. For the purpose of measuring a generalizable efficiency factor we consider four variables which are measured after a one-versus-one combat situation:

- The binary value of having won against the opponent (w)
- The normalized temporal duration of the fight (t)
- The agent's remaining health point (HP) percentage (hp_a)
- The opponent's remaining HP percentage (hp_o)

All variables lie between 0 and 1, are multiplied with their respective weight ($\alpha, \beta, \gamma, \delta$; all weights are equal for this study) and normalized over weights and the sum of observations (n), resulting in the final proficiency ϕ that ranges from worst-case (0) to optimal (1) performance:

$$\phi = \sum_{i,j=0}^{n} \frac{\alpha w + \beta(1-t) + \gamma hp_a + \delta(1-hp_o)}{(\alpha+\beta+\gamma+\delta)n^2}$$

IV. EVALUATION

In this section we outline the two evaluation environments (one-on-one PvE and PvP) used to assess the viability of each class. In addition, the section presents the regulation technique we used to mitigate balance discrepancies.

A. Player versus Environment (PvE) Evaluation

Focusing on differences between classes in one-versus-one situations, we chose to investigate performances of DPBM-driven agents encountering 100 opponents that incrementally increase in difficulty (see Fig. 3). To render the analysis visualizable and human-understandable, we only manipulate the offensive and defensive capabilities of each opponent, i.e. its *attack* and *maximal health points (maxHP)* values respectively. Since the proficiency distribution of a player population is likely to entail a great variance, the initial configuration was set to a trivial encounter, whereas the following modulations of the opponent increased *attack* and/or *maxHP* by 25% per iteration, up to a barely defeatable enemy. This led to a two-dimensional benchmark setup of continually increasing challenge with similarly decreasing expected proficiency. Figure 4 demonstrates the proficiency distributions together with the corresponding ϕ values of the best and worst DPBM-driven agent compared to the overall average.

After the evaluation of the 213 DPBM agents across 100 opponent configurations with incrementally increasing difficulty, the resulting proficiency estimations were categorized into the game-specific classes in order to compare their performance.

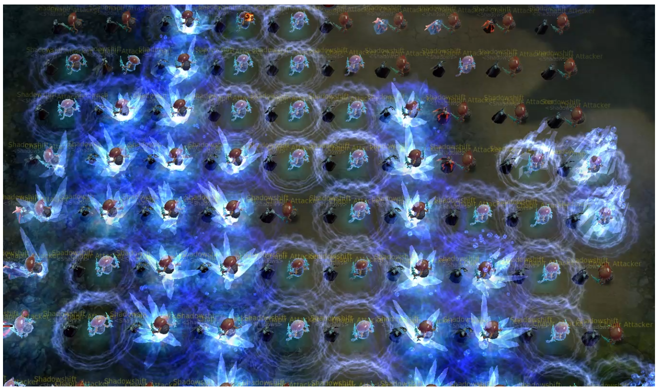


Fig. 3. In-game screenshot of the PvE benchmark (*Aion* [19]). DPBM-driven player replicas encounter 100 heuristic opponents with increasing difficulty in one-on-one situations (attack horizontally, maxHP vertically). Depending on the game state between the agent and its target, emerging behavioral patterns for action preferences and sequences can be monitored. For reasons of observability, entities are spawned with sparse distance to other confrontations. Yet, they are only able to damage and influence their respective counterpart.

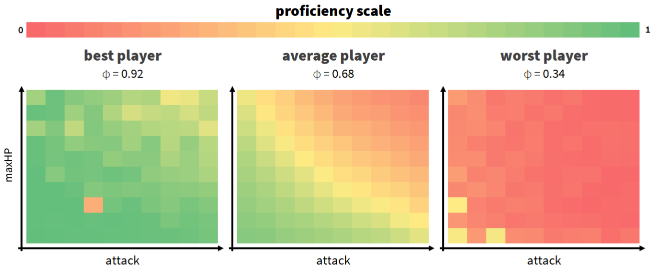


Fig. 4. Proficiency heatmaps of the best, average and worst player replication of the benchmark. The horizontal axis denotes the increasing attack value of the heuristic opponent while the vertical axis describes the increasing HP value of it (+25% per step, respectively).

As baselines, for each respective class, we observed the performance of an agent that modelled generalized (non-individual) behavior and an agent with random decision making.

B. Player versus Player (PvP) Evaluation

While the process of Section IV-A approximates the players' ability to cope with PvE encounters and therefore provides one measure of balance estimation, another dimension worth examining is the balance between the classes themselves. Thus, a subsequent evaluation pitted all player replicas against each other in one-on-one PvP confrontations, leading to $22,578$ unique combinations (including intra-class battles). The proficiency outcomes of these matches were pooled and averaged to measure systematic dominance or inferiority relationships between classes. To prevent never-ending duels (e.g. between two agents using the healer classes and mainly defensive strategies), the maximal duration was capped at five minutes.

C. Regulation

The DPBM approach primarily focuses on *informing* game development about possible imbalances within a player population; however, certain regulation techniques can follow immediately, assuming that all classes should follow a similar proficiency distribution. The most direct approach of regulation would be to tune the environmental parameters such as the offensive and defensive capabilities of opponents (similar to Section IV-A). Thus, we subsequently determine a meaningful target proficiency (in this example, the mean proficiency of all iterations) and computed the mean squared error of measured proficiency values of each player in a class. From this, we reveal the approximate parameter values to tune by calculating the center of mass of these errors per class. Eventually, the proficiency distributions for all classes can be compared, given the PvE benchmark results of the respective players and using the tuned parameters for their opponents.

V. Results

Table III outlines the testing prediction accuracies of the employed DPBMs (using a 80-20 holdout validation method) including conservative heuristic filtering within the most probable 1, 5 and 10 skill choices. In addition, the table includes the training times per player as measured on a NVIDIA GeForce RTX2080 using Keras 2.2.4 with TensorFlow 2.0.0 backend.

TABLE III
Prediction accuracy on the testing set, with corresponding training times for the various DPBMs employed.

	Testing accuracy			**Training time**
	Top-1	Top-5	Top-10	
M:	61.3%	75.3%	81.3%	$7.24s$
SD:	22.4%	11.8%	14.9%	$1.68s$

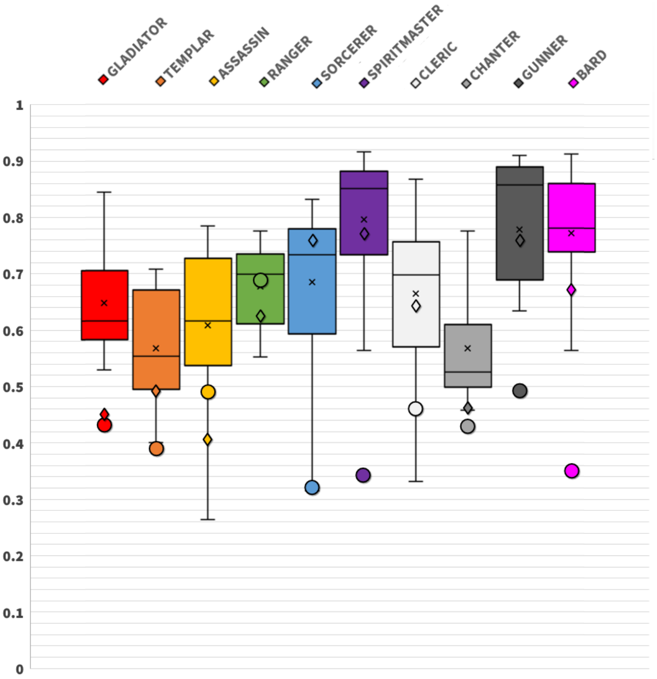


Fig. 5. Proficiency results of player replicas across different classes. The graph depicts mean values (indicated by **x**), median values (–), the proficiency of the generalized model of the class (◇) and random guessing (○).

A. Player versus Environment (PvE) Evaluation

Using a one-way ANOVA, we find a significant difference of DPBM-agent proficiency across classes in the PvE evaluation, displaying a large effect size ($F(9, 203) = 9.63$, $p < .01$; partial $\eta^2 = 0.3$; see Fig. 5). Further we use Bonferroni-corrected two-tailed Welch's t-tests to highlight statistical differences between particular classes. Highlighting notable disparities, players of the Spiritmaster, Gunner or Bard class scored higher proficiency values than most other classes while Chanter and Templar players were almost consistently outperformed by other classes ($p < 0.05$). After further t-tests, significant proficiency differences between individual DPBM and generalized models became apparent ($p < 0.05$, Cohen's $d = 0.61$). This also holds in comparison to the random decision making agent ($p < 0.01, d = 2.45$).

B. Player versus Player (PvP) Evaluation

With respect to the PvP evaluation, Fig. 6 visualizes average proficiency values of player replicas from one class compared to all other classes. While classes within the same archetype (e.g. Gladiator and Templar both being physical Melee classes or Sorcerer and Spiritmaster both being Magical ranged

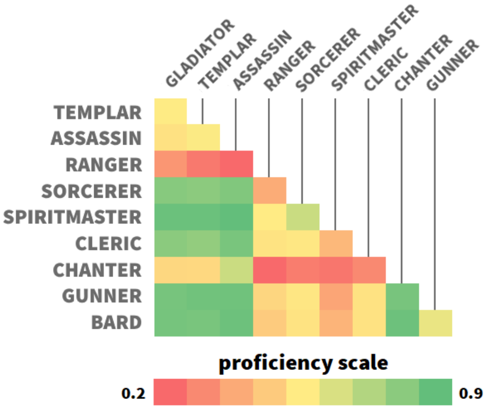


Fig. 6. Mean proficiency results of DPBM-driven player replicas in one class (vertically) when fighting replicas of other classes (horizontally).

classes) had few proficiency differences ($p > 0.05$), distinct superiority relationships emerge when different archetypes are matched up. Rangers scored significantly lower against Melee classes (Gladiator, Templar, Assassin, $p < 0.05$), yet they consistently outperformed the Magic classes (Sorcerer, Spiritmaster, Gunner, $p < 0.05$). The Magic classes were equally and consistently able to dominate Melee classes, effectively representing a rock-paper-scissors-like interaction scheme. The Heal classes (Cleric, Bard) also outperformed Melee, yet succumbed to both Rangers as well as to the Magic classes ($p < 0.05$). Being the game's primary support class, Chanters were dominated by the majority of opposing classes.

C. Regulation

Figure 7 visualizes the regulation with the tuned opponent parameter values for each class and the corresponding proficiency distribution of DPBM-driven player replicas. According to a one-way ANOVA, there are no significant differences remaining between class proficiency values after parameter tuning ($F(9, 203) = 1.42$, $p > 0.05$).

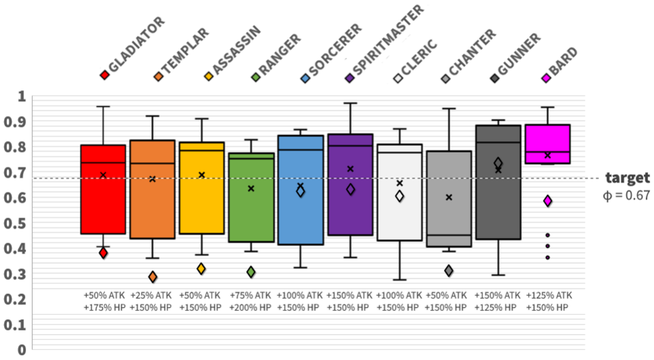


Fig. 7. Proficiency results of player replicas across different classes after parameter tuning for a balanced resulting proficiency ($\phi = 0.67$). The graph depicts mean values (indicated by **x**), medians values (–), outlier values (•) and tuned parameters.
◊ indicates the proficiency value of the generalized class model.

When compared to average adjustment values, opponents' *attack* was regulated weaker for Melee and Support classes, but notably higher for Magic and Heal classes. In contrast, Rangers faced opponents with average *attack*, but considerably increased *maxHP* after regulation.

VI. Discussion

With regards to the PvE evaluation, the ANOVA and subsequent post-hoc tests revealed significant differences in proficiency between player replicas of different classes. This might indicate imbalances of these classes, yet it should be interpreted with respect to the underlying design guidelines. For instance, the relatively low proficiency scores of the Chanter and Templar classes likely stem from their reliance on other players, either to support or to receive support from respectively. Still, under the assumption that primarily damage-dealing classes should be equally viable, certain discrepancies emerge that point to certain magical classes (such as Spiritmaster, Gunner or Bard) outperforming physical damage dealers (such as Gladiator, Assassin) significantly. When interpreting results of the PvP benchmark, it is worth noting that balanced viability does not necessarily have to result in each class having equal chances against all others. Depending on the underlying design agenda, a similarly balanced constellation is a rock-paper-scissors-like interaction scheme between classes, which this evaluation was able to demonstrate approximately (see Fig. 8). Nevertheless, the inferior performance of the Chanter in both one-on-one PvE as well as PvP situations might encourage developers to augment the versatility of this class if its role is not only meant to support other players.

The approach introduced with this paper has no access to the underlying design constellations of the original game and primarily aims to inform developers about imbalances. However, we have already indicated and tested possible procedures to adjust the viability of these classes towards a balanced configuration. The adjustment of opponents for individual classes, based on the proficiency distribution of its players, has proven to detect configurations that end up in a more balanced outcome (see Fig. 7 as compared to Fig. 5). While this

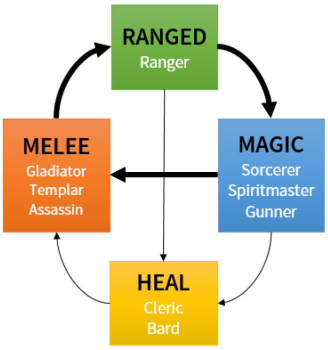


Fig. 8. Illustrated outcome of the DPBM-driven PvP simulation. On average, players of Melee classes outperform Rangers, which themselves counter Magic classes, which eventually beat Melee classes. Heal classes are dominated by Rangers and Magic classes, but can withstand Melee.

method yields promising results in terms of balanced viability for single-player games or solo dungeons, its adjustment is not trivially applicable to group PvE situations, since the opponents are only tailored to a single class and the interaction between classes further confounds the attunement. A more comprehensive regulation method would be the adjustment of in-class parameters, such as their own offensive or defensive values or particular skill values. Yet, this would significantly influence the interaction between the classes and might harm the likely intended rock-paper-scissors scheme within. If aiming for equal proficiency of all classes against each other, an iterative procedure of attunement and re-simulation would be expedient, in that the largest proficiency mismatch between classes is detected, adjusted in favor of the inferior class and affected matchups are re-simulated, reiterated up to a predefined threshold.

Based on the empirical evidence presented, our previously posed research questions can be answered as follows:

- *Significant imbalances between in-game classes can be detected through DPBM within PvE and PvP.*
- *Design specifications can be established via regulation based on DPBM-driven simulation results.*

VII. Limitations and Future Work

During the implementation of this approach, different constraints and assumptions had to be taken into account that eventually lead to a number of limitations. Perhaps most importantly, classes (especially in MMORPGs) are often designed to vary in versatility within different situations or against different classes. This includes classes that benefit greatly from party-play versus classes that are tailored for single-player situations, those who focus on dealing damage to many enemies instead of single targets or not dealing damage at all (being busy with tanking, healing or supporting otherwise). Nevertheless, the presented technique is not constrained to damage dealing, but overall one-on-one versatility. DPBM can quantify these differences to inform game developers whether their intended design aligns with the actual outcomes of a population playing it. Evidently, this requires data from a player population to employ the testing procedures, which limits its versatility before the game's launch. However, it is applicable for never-ending balance observations (and predictions) and for benchmarking novel challenges introduced with later patches or DLCs. Apart from *generalization* or *random play* in the PvE evaluation, an optimally playing agent (e.g. by self-training/reinforcement learning) would be an additional interesting candidate, to compare if this approach is closer to the real population. To filter out the influence of different attribute stats, we normalized equipment and other relevant configurations throughout all characters. A closer (yet very temporary) approximation of the overall population capability could be realized with this approach if the equipment range was taken into consideration. Eventually, player models in the PvP evaluation were driven by the same behavior, independent of their opponent, since this behavior was only trained on data stemming from battles against their own class [17]. This likely distorted the results and should be repeated when enough data of the respective situations are given; however, it does not diminish the potential of DPBM.

For future work, we primarily seek to refine behavior modeling by introducing more variables, such as global movement information (encompassing higher level goals) or the estimation of individual players' precision and their temporal cognitive computation demand. Apart from the constraint of two dimensions (attack and defense), the challenge of the PvE encounters can further be examined by altering the skill sets, decision making or movement behavior of enemies. The simulations themselves can likely be sped up by calculating battles without graphical representations. Eventually, the applicability of this approach will be investigated with respect to significantly more complex multi-player situations, such as in adjusting boss battles for a population (PvE) or simulating large-scale competitive sieges (PvP), throughout multiple player experience evaluations. Furthermore, if a mapping from mere behavioral patterns to in-game proficiency can be constructed, this prediction might augment matchmaking (for both PvP and PvE) bringing together players with approximate skill levels more accurately.

VIII. Conclusion

Balancing in-game parameters and classes to ensure diverse viable choices for players is a challenging, time-consuming and toiling expense for game developers. While traditional approaches employ expert analysis, excessive human playtesting and persistent debates with the community, this paper introduces the use of an individual generative player modeling technique (DPBM) for automating game balancing. Using a dataset of 213 players that visited a single-player dungeon of the MMORPG *Aion* over the course of six months, we we generated individual agents replicating human play behavior. Within the context of one-on-one PvE battles, we detected and sufficiently regulated significant effects between classes. For the interaction between classes, a PvP evaluation among all players revealed a rock-paper-scissors-like interaction scheme that is likely to resemble the original developers' design. The proposed approach is able to inform game development about PvE and PvP imbalances quantitatively and provide empirical evidence that player behavior entails a degree of individual proficiency.

References

[1] M. Washburn Jr, P. Sathiyanarayanan, M. Nagappan, T. Zimmermann, and C. Bird, "What went right and what went wrong: an analysis of 155 postmortems from game development," in *Proceedings of the 38th International Conference on Software Engineering Companion*, 2016, pp. 280–289.

[2] D. Lin, C.-P. Bezemer, and A. E. Hassan, "Studying the urgent updates of popular games on the steam platform," *Empirical Software Engineering*, vol. 22, no. 4, pp. 2095–2126, 2017.

[3] C. Buhl and F. Gareeboo, "Automated testing: A key factor for success in video game development. case study and lessons learned," in *proceedings of Pacific NW Software Quality Conferences*, 2012, pp. 1–15.

[4] J. Pfau, J. D. Smeddinck, and R. Malaka, "Automated game testing with icarus: Intelligent completion of adventure riddles via unsupervised solving," in *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play*. ACM, 2017, pp. 153–164.

[5] Blizzard Entertainment, "*StarCraft II*," Game [PC], July 2010, blizzard Entertainment, Irvine, CA, USA. Played 2017.

[6] Blizzard Entertainment, "*Overwatch*," Game [PC,PS4,XboxOne,Switch], May 2016.

[7] ArenaNet, "*Guild Wars*," Game [PC], April 2005, arenaNet, Bellevue, WA.

[8] D. Sirlin, "Balancing multiplayer competitive games," in *Game Developer's Conference*, 2009.

[9] K. Hullett, N. Nagappan, E. Schuh, and J. Hopson, "Empirical analysis of user data in game software development," in *Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*. IEEE, 2012, pp. 89–98.

[10] C. Lewis and N. Wardrip-Fruin, "Mining game statistics from web services: a world of warcraft armory case study," in *Proceedings of the Fifth International Conference on the Foundations of Digital Games*. Citeseer, 2010, pp. 100–107.

[11] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[12] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, p. 484, 2016.

[13] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel *et al.*, "Mastering chess and shogi by self-play with a general reinforcement learning algorithm," *arXiv preprint arXiv:1712.01815*, 2017.

[14] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev *et al.*, "Grandmaster level in starcraft ii using multi-agent reinforcement learning," *Nature*, vol. 575, no. 7782, pp. 350–354, 2019.

[15] G. N. Yannakakis and J. Togelius, *Artificial Intelligence and Games*. Springer Nature, 2018.

[16] J. Pfau, J. D. Smeddinck, and R. Malaka, "Towards deep player behavior models in mmorpgs," in *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*. ACM, 2018, pp. 381–392.

[17] J. Pfau, J. D. Smeddinck, and R. Malaka, "Enemy within: Long-term motivation effects of deep player behavior models for dynamic difficulty adjustment," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, 2020. *In press*.

[18] J. Pfau, J. D. Smeddinck, I. Bikas, and R. Malaka, "Bot or not? user perceptions of player substitution with deep player behavior models," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, 2020. *In press*.

[19] NCsoft, "*Aion*," Game [PC], Seongnam, South Korea, September 2008, nCSoft, Seongnam, South Korea. Played August 2019.

[20] S. Radomski and T. Neubacher, "Formal verification of selected game-logic specifications," *on Engineering Interactive Computer Systems with SCXML*, p. 30, 2015.

[21] S. Varvaressos, K. Lavoie, S. Gaboury, and S. Hallé, "Automated bug finding in video games: A case study for runtime monitoring," *Computers in Entertainment (CIE)*, vol. 15, no. 1, p. 1, 2017.

[22] Y. Zheng, X. Xie, T. Su, L. Ma, J. Hao, Z. Meng, Y. Liu, R. Shen, Y. Chen, and C. Fan, "Wuji: Automatic online combat game testing using evolutionary deep reinforcement learning," in *Proceedings of the 34th ACM/IEEE International Conference on Automated Software Engineering*, 2019.

[23] B. Chan, J. Denzinger, D. Gates, K. Loose, and J. Buchanan, "Evolutionary behavior testing of commercial computer games," in *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*, vol. 1. IEEE, 2004, pp. 125–132.

[24] J. H. Bécares, L. C. Valero, and P. P. G. Martín, "An approach to automated videogame beta testing," *Entertainment Computing*, vol. 18, pp. 79–92, 2017.

[25] S. Iftikhar, M. Z. Iqbal, M. U. Khan, and W. Mahmood, "An automated model based testing approach for platform games," in *2015 ACM/IEEE 18th International Conference on Model Driven Engineering Languages and Systems (MODELS)*. IEEE, 2015, pp. 426–435.

[26] C. Schaefer, H. Do, and B. M. Slator, "Crushinator: A framework towards game-independent testing," in *Proceedings of the 28th IEEE/ACM International Conference on Automated Software Engineering*. IEEE Press, 2013, pp. 726–729.

[27] E. J. Powley, S. Colton, S. Gaudl, R. Saunders, and M. J. Nelson, "Semi-automated level design via auto-playtesting for handheld casual game creation," in *2016 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2016, pp. 1–8.

[28] M. Shaker, M. H. Sarhan, O. Al Naameh, N. Shaker, and J. Togelius, "Automatic generation and analysis of physics-based puzzle games," in *2013 IEEE Conference on Computational Inteligence in Games (CIG)*. IEEE, 2013, pp. 1–8.

[29] G. Volkmar, N. Mählmann, and R. Malaka, "Procedural content generation in competitive multiplayer platform games," in *Joint International Conference on Entertainment Computing and Serious Games*. Springer, 2019, pp. 228–234.

[30] M. Schatten, B. O. urić, I. Tomičič, and N. Ivkovič, "Automated mmorpg testing–an agent-based approach," in *International conference on practical applications of agents and multi-agent systems*. Springer, 2017, pp. 359–363.

[31] M. Van Kreveld, M. Löffler, and P. Mutser, "Automated puzzle difficulty estimation," in *2015 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2015, pp. 415–422.

[32] F. Southey, G. Xiao, R. C. Holte, M. Trommelen, and J. W. Buchanan, "Semi-automated gameplay analysis by machine learning." in *AIIDE*, 2005, pp. 123–128.

[33] A. Jaffe, A. Miller, E. Andersen, Y.-E. Liu, A. Karlin, and Z. Popovic, "Evaluating competitive game balance with restricted play," in *Eighth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2012.

[34] P. García-Sánchez, A. Tonda, A. M. Mora, G. Squillero, and J. J. Merelo, "Automated playtesting in collectible card games using evolutionary algorithms: A case study in hearthstone," *Knowledge-Based Systems*, vol. 153, pp. 133–146, 2018.

[35] F. de Mesentier Silva, S. Lee, J. Togelius, and A. Nealen, "Ai as evaluator: Search driven playtesting of modern board games." in *AAAI Workshops*, 2017.

[36] T. Mahlmann, J. Togelius, and G. N. Yannakakis, "Evolving card sets towards balancing dominion," in *2012 IEEE Congress on Evolutionary Computation*. IEEE, 2012, pp. 1–8.

[37] P. Beau and S. Bakkes, "Automated game balancing of asymmetric video games," in *2016 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2016, pp. 1–8.

[38] M. Morosan and R. Poli, "Automated game balancing in ms pacman and starcraft using evolutionary algorithms," in *European Conference on the Applications of Evolutionary Computation*. Springer, 2017, pp. 377–392.

[39] R. Leigh, J. Schonfeld, and S. J. Louis, "Using coevolution to understand and validate game balance in continuous games," in *Proceedings of the 10th annual conference on Genetic and evolutionary computation*. ACM, 2008, pp. 1563–1570.

[40] C. Holmgard, M. C. Green, A. Liapis, and J. Togelius, "Automated playtesting with procedural personas with evolved heuristics," *IEEE Transactions on Games*, 2018.

[41] S. F. Gudmundsson, P. Eisen, E. Poromaa, A. Nodet, S. Purmonen, B. Kozakowski, R. Meurling, and L. Cao, "Human-like playtesting with deep learning," in *2018 IEEE Conference on Computational Intelligence and Games (CIG)*. IEEE, 2018, pp. 1–8.

[42] J. Pfau, J. D. Smeddinck, and R. Malaka, "Deep player behavior models: Evaluating a novel take on dynamic difficulty adjustment," in *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, 2019, p. LBW0171.

[43] G. N. Yannakakis, P. Spronck, D. Loiacono, and E. André, "Player modeling." Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013.

[44] A. M. Smith, C. Lewis, K. Hullet, G. Smith, and A. Sullivan, "An Inclusive View of Player Modeling," in *Proceedings of the 6th International Conference on Foundations of Digital Games*, ser. FDG '11. New York, NY, USA: ACM, 2011, pp. 301–303, event-place: Bordeaux, France. [Online]. Available: http://doi.acm.org/10.1145/2159365.2159419

Automated Game Testing with ICARUS: Intelligent Completion of Adventure Riddles via Unsupervised Solving

“For human beings, testing the same game for a longer period of time can be quite demanding of both their creativity and concentration. Since projects require different styles of testing at different times, such as simply playing through the game as quickly as possible or in-depth bug testing of various parts of the game, the testers often have to actively force themselves to leave the path their brains are used to and to come up with new creative ways of breaking the game. Additionally, even for a linear game, the number of possible combinations as well as the order they are made in during a play session can become extremely large.”

- Maik Hildebrandt, Head of QA at Daedalic Entertainment [12]

Johannes Pfau
University of Bremen
Bibliothekstraße 1,
28359 Bremen, Germany

Daedalic Entertainment
Papenreye 51,
22453 Hamburg, Germany
jpfau@tzi.de

Jan David Smeddinck
ICSI, University of California, Berkeley
1947 Center St, Berkeley, CA 94704
jandavid@icsi.berkeley.edu

Rainer Malaka
University of Bremen
Bibliothekstraße 1,
28359 Bremen, Germany
malaka@tzi.de

CHI PLAY'17 Extended Abstracts, October 15–18, 2017, Amsterdam, Netherlands © 2017 Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-5111-9/17/10…$15.00
https://doi.org/10.1145/3130859.3131439

Abstract

With *ICARUS*, we introduce a framework for autonomous video game playing, testing, and bug reporting. We report on the design rationale, the practical implementation, and its use in game development industry projects. The underlying solving mechanic is based on *discrete reinforcement learning* in a dualistic fashion, encompassing volatile short-term memory as well as persistent long-term memory that spans across distinct game iterations. In combination with heuristics that reduce the search space and the possibility to employ pre-defined situation-dependent action choices, the system manages to traverse complete playthrough iterations in roughly the same amount of time that a professional game tester requires for a speedrun. The *ICARUS project* was developed at Daedalic Entertainment. The software can be used to generically run all adventure games built with the popular *Visionaire Engine* [6] and is currently used for evaluating daily builds, for large-scale hardware compatibility and performance tests, as well as for semi-supervised quality assurance playthroughs.

The supplementary video depicts real-time solving with active control and observation via a *web control panel.*

Author Keywords

Automated game testing; quality assurance; reinforcement learning; automated bug reporting; continuous performance analysis; continuous integration testing

ACM Classification Keywords

D.2.5 [Software Engineering]: Testing and Debugging;
I.2.1 [Artificial Intelligence]: Applications and Expert Systems – Games;
K.8 [Personal Computing]: Games

INTRODUCTION

Continuous and extensive quality assurance (QA) plays an important role in the video game industry. Modern games are often immensely complex software systems that offer a broad range of possible game experiences and are often immediately used by a large number of consumers. At the same time, bugs or game glitches can considerably harm the immersion, fun, and endanger the overall game experience. Thus, a large portion (typically ~10-20 %) [5] of the budget for a particular video game production is spent solely on finding and reporting bugs, testing traversability, compatibility, performance, and aesthetics. Such issues are usually broken down into three major categories of severity (**A**: Crashes/Freezes, **B**: Blocker and **C**: General. See: Table 1). While the order of severity is descending, the probability to miss a bug of the particular type is simultaneously ascending. Furthermore, the majority of missed bugs stems from error blindness (due to the habituation to the game procedures and the sticking to established action choice patterns), a specific form of change blindness [20], that testers grow more likely to fall victim to the more often and frequently they play-test the same game.

A	**Crashes/Freezes** Shutting down the game unexpectedly or preventing the screen from rendering any further.
B	**Blocker** Resulting in a game state from which no further game progress can be made.
C	**General** Graphical flaws, animation issues, typos, glitches.

Table 1. Common categories of bugs in video games [1, p. 178].

In this light, the introduction of ICARUS in professional video game development does not only aim at reducing labor costs for QA, but also at improving the bug tracking performance and at decreasing the cognitive load for human testers, assisting in all of the bug categories named above. Following a discussion of the current state of the art in game testing, automated testing, and the application of techniques from artificial intelligence / machine learning in these contexts, we present the rationale and architecture of ICARUS in detail, together with exemplary use cases in the form of an industry case study and a discussion that reflects on the value that such systems can currently provide in game development processes, as well as an outlook on future developments in the area of intelligent automated game testing. This technical framework description and the according case study provide a report on a novel system for automated game testing with adventure games. Readers from the scientific community will gain a better understanding of the extent to which the game industry is embracing applied artificial intelligence and machine learning in contexts beyond classic game AI, while readers with a background in the game industry can gain a better understanding of how similar approaches might benefit their own projects.

RELATED WORK

So far, automated frameworks for testing software or specifically video games have been developed. Automated approaches exist, for example for selected, discrete performance measurements, such as determining the FPS at which a game can run on a given system, or the CPU and memory load when starting or running the game using new games or

Left clicks	for each available target object
Right clicks	for each available target object
’Use’ each item	with each available target object
’Use’ each item	with each available target item
’Look at’ item	for each available target item

Table 2. Common generic action categories for adventure games. Dialog options are handled separately, see: Dialog.

saved game states [9]. While such systems can frequently detect issues in category A, blockers and especially more general flaws of non-technical nature, like unsolvable conditions in complex quests, remain undetected and require manual involvement. Other approaches simulate playthroughs, using manually predetermined [3, 8, 11] or recorded [4, 10] action sequences. These systems can help with detecting many potential blockers and some more general issues. However, they require manual adaptation or re-recording of the action sequences whenever the procedure changes, which typically happens on a daily basis during the active game development of modern games. Furthermore, most of the time video games do not strictly constrain the player regarding the order in which a sequence of actions needs to be executed. Actions are not always mandatory to perform in order to progress in a game and often the player is given several choices on how to proceed. The former deterministic approaches thus require different manually defined (or recorded) action sequences. Even in games with just a few optional branches or decision points, the resulting combinatorial explosion clearly illustrates the limitations of such manually guided automatic testing. For some specific games, targeted automatic solvers exist that iterate over the whole possible action space of the game (e.g. in a brute force breadth-first search fashion [7]). However, these examples include a large number of actions that are repeated over and over again, although they often do not require validation in each iteration. Non-deterministic approaches were successful in spotting unwanted NPC behavior and glitches [13, 14], parameter tuning [15], testing formal core mechanics of multi-agent systems [16] or detecting every bug expressible in a proposed language [17], but in rather strictly limited situations, whereas our approach is tailored to the needs of traversing complete games. For a number of board games, complete, AI-guided play testing approaches exist [17,18,19], which clearly identified loopholes and design flaws, yet lack industrial application.

As the following section will show in further detail, the ICARUS system tackles a number of shortcomings of the systems that were discussed in this section. With an active and guided machine learning approach, it narrows the playthrough down to the most relevant actions, after having explored the complete game action set, highlighting potential yet less common blockers as well as general blockers, that - unlike crashes or freezes - could have easily gone undetected using more traditional automated testing. As Figure 7 shows, this can notably speed up the progress of QA evaluations.

ICARUS

The system for *intelligent completion of adventure riddles via unsupervised solving* (ICARUS) is a generic, platformindependent game solver written in *Lua* [2] and optimized for the *Visionaire Game Engine* [6]. ICARUS was developed at Daedalic Entertainment, a leading company in the development and publishing of adventure games. Hence, it is primarily focused on solving the main functionality and riddles of adventure games. However, the solver follows a more generic design rationale, allowing for the integration of many meaningful types of game actions that can be adapted to any similarly traversable game genre, since the solver system interacts with the game environment using the same commands as a human player would. In order to facilitate human supervision, the ability to start, stop and play in the meantime, as well as for the

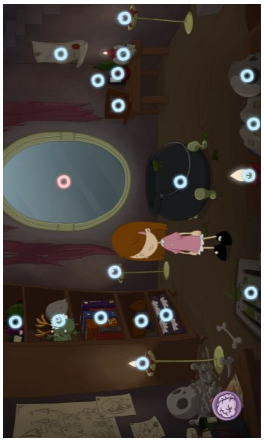


Figure 2. Example scene of the game *Anna's Quest*, containing 18 target objects, indicated by blue rings (for illustration purposes only). On each target, the actions of Table 2 can be applied.

most accurate game representation and bug reproducibility, the games are played in real-time. However, soft acceleration methods, such as character speed modification or skipping dialog texts, menus, videos, etc. can be turned on and off at run-time via the *web control panel*. In comparison to the existing approaches mentioned before, ICARUS can not only record performance metrics (FPS, RAM, CPU usage etc.) at single points of time, but it can track these measurements continuously over the whole span of a game iteration, recognizing crucial performance issues and pinning them down to concrete game situations and hardware constellations. For these iterations, it is not constrained to pre-determined sequences or recorded playthroughs, but it will dynamically explore the game state regardless if knowledge about the current situation is already given or not, using the solving process explained in the following section. The persistently learning nature of this setup allows ICARUS to combine the advantages of complete action testing and fast playthroughs, since it will start with a broad, explorative search over all possibilities of the game state and improve itself (in terms of number of executions per playthrough, thus also speed) with each further game iteration it traverses.

Solving Process

In most adventures, the actions that lead to progress are well-defined, generally consisting of (a) interacting with objects or characters, (b) collecting items, (c) combining items with other items, objects or characters, and (d) choosing from dialog options. Thus, as long as the acting character is not busy executing an action, ICARUS comes up with a representation of the game state by collecting the set of possible actions (Table 2) and stores it temporarily in a list of *currentActions*.

On these current actions, ICARUS remembers possible reward outcomes from previous choices that are stored in *currentRewards* $\in Z^{a\times4}$, the matrix mapping observed actions to reward values (where 0 is assigned to unobserved actions), which is a subset of *allRewards* $\in Z^{b\times4}$, containing short-term as well as long-term reward information (b being the amount of all actions observed in this and all previous game iterations in total, in 4 information dimensions about the action type, target, used item and reward).

Action selection

To choose an action, ICARUS performs a (random if configured to function probabilistically, consecutive if configured to use complete action iteration) selection of *maxCurrentRewards* ⊂ *currentRewards*, which contains only the actions that yield the highest rewards among currentRewards. After the selection, ICARUS executes the corresponding action (e.g. a left click on a target T), waits until the completion of the action and evaluates the reward.

Reward learning

If the chosen action led to game progress (e.g. the inventory state changed, a quest progressed, targets appeared/disappeared, access to new areas is opened, etc.), a configurable, positive number is remembered persistently for this action in the long- and short-term memory. In general, a given state change can be considered positive if it is unrepeatable and leads to the enabling of formerly unavailable game actions. If no observable change happened, ICARUS punishes the last action by setting the action reward in the short-term

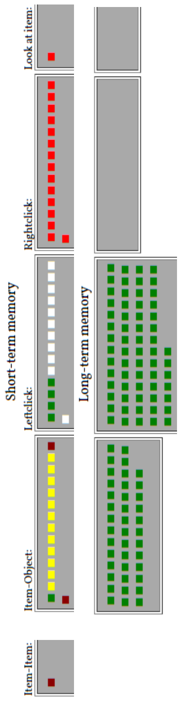


Figure 3. Reward map in a game state containing 1 item and 15 possible action targets, visualized in the *web control panel*.

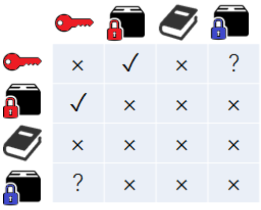
memory to the respective punishment parameter that can equally be configured per action category. These configurations can take place in the script itself or via the *web control panel* at run-time, with +1 as the default for every positive game state change and -1 as the default for every action type in any other case. In that way, actions that rarely contribute to progress (e.g. looking at items) can be punished harder than important actions (e.g. left click). Once the set of current actions is iterated, i.e. every possible action has a negative temporary reward, the short-term reward map is *soft-reset (see: Soft-resetting the reward map)*.

The segmentation into short- and long-term memory is important since the completion performance increases with *short-term memory action selection* and the *final ICARUS action selection* as illustrated in Figure 7, which is mainly caused by the inclusion of long-term rewards. Entries of the long-term reward map are loaded whenever the respective action is currently available and thus strongly determine the sequence of the action selection. Nevertheless, the short-term memory is still needed, since the rewards for the actions in the long-term memory are learned from a very specific game state that has to be the same (or similar) to the current game state in order to yield actual game progress. If an action is remembered positively from the long-term reward, but the current game state yields no reward for this action, the respective punishment does not overwrite the positive reward in the long-term memory, but it will store a negative reward in the short-term memory, which ICARUS will use in the end for the action selection.

For example, ICARUS might record a positive reward if the game object *door* is opened, where the underlying game state had a precedent action that unlocked the door. In the next game iteration, ICARUS will remember the positive reward and prioritize attempting to open the door, even if it is still locked. The punishment reward will be recorded in the short-term memory and ICARUS will proceed with other actions (most likely the collection of a *key* and the unlocking action *key-with-door*) before it will reconsider opening the door. This reconsideration process is realized by the following *soft-resetting*.

Soft-resetting the reward map

In this process, every negative value of the map is increased by 1, so that -1 rewards result in 0 ("unobserved") and even lower values are coming one step closer to a possible re-observation. That means that an action from an action type that is configured to be punished with -5 requires 5 soft-resets to be considered for execution again. If entries that have a positive long-term reward turn to 0 ("unobserved") in this process, they are set to the respective original instead to ensure that ICARUS prefers the execution of them again, after the soft-reset. Figure 3 visualizes the reward map right after a soft-reset, where several actions containing long-term rewards are reset into positive values (green), some actions were punished harder and thus yielded a high negative reward (red), and some actions had a low negative reward which were reset to 0 in this step (white). In total, the technique of soft-resetting results in a normalization of the reward space, so that negatively rewarded actions have a chance to be executed again, but strictly after positively and new actions are tested.



	🔑	📦	📖	📦
🔑	×	✓	×	?
📦	✓	×	×	×
📖	×	×	×	×
📦	?	×	×	×

Figure 4. Example inventory containing 4 items, resulting in 16 combinations. Two actions are actually beneficial for game progress (X), two have also to be tested (?) and 12 can be discarded via educated guessing (X).

Figure 6. Inventory in the example game state of *Anna's Quest*. 11 items are held that can be combined with each other or used on the target objects of the scene (Figure 2).

Educated Guessing

The majority of adventure games are composed using puzzles that challenge the human power of deduction and creative combination by demanding the correct usage of items with object targets or other items. In theory, every possible *item-item* and *item-object* combination has to be considered in the process of action collection. However, most of these combinations are evoking only standard responses and the set of items that lead to progress in combination with each other or with objects is most often small in comparison to the possible set of all combinations.

For example, an inventory containing a *red key*, a *red box*, a *book* and a *blue box* already has 16 possible item-item combinations, where only the combination *red key-with-red box* (or vice versa) leads to actual game progress (see: Figure 4). Educated guessing (comparable to pruning a search space in classical AI search) is the process of discarding options that do not make sense to evaluate in the first place, e.g. combining each item with itself, combining the *book* to any of the items or trying to open one of the boxes with another box. It will leave *red key-with-red box* and *red key-with-blue box* as possible actions, even if the latter won't have any positive effect.

The decision why *red key-with-blue box* is still considered an action that could yield reward and e.g. book-with-blue box is not, is made on the type of response that the respective action evokes. *red key-with-blue box* will trigger an evaluation about whether the key fits the box (or a comment that is precisely defined for this situation, e.g. "This key doesn't fit."), where book-with-blue box will only trigger a generic standard response, e.g. "This doesn't work". Once this generic response is triggered and it's execution therefore part of a test run, it does not need to be triggered by the remaining zero-effect item combinations again, and the solving process can thus be sped up using educated guessing without sacrificing the validity and reliability of a test run. This distinction between action types works by assessing engine information about the particular action, which cannot explicitly tell the reward or the outcome, but is able to discard purely cosmetic or commentary actions that often trigger random default phrases. Every action that falls under certain categories, invokes standard functions, is tagged with default codes, or can be parsed for yielding no meaningful game progress can thus be strictly excluded from repeated executions in order to drastically reduce the amount of combinations for active execution checks.

In a real world example, the combinatorial complexity can become very severe. Figure 5 displays the extent of combinations of a game state containing 11 available items (Figure 6) and 18 available targets (as in Figure 2). Having many items in the inventory leads to exponential growth of the number of possible actions, which becomes even worse in scenes with many objects. This is where educated guesses come into play in order to avoid exponential growth in execution times that can cause considerable costs even with automated testing. As in the examples mentioned before, red cubes in the figures represent actions that were already tested and only yielded negative reward. However, a great portion of these actions (marked in dark red) will never be chosen, since ICARUS discards them using educated guessing. The resulting subset of item-item combinations still contains a number of combinations

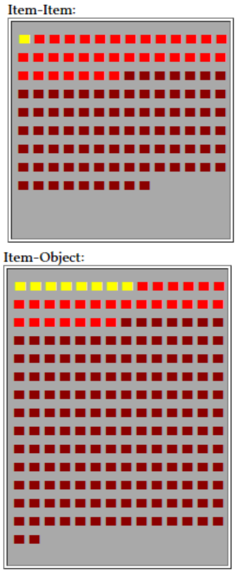


Figure 5. Example reward map subset in a scene of the game *Anna's Quest*. Yellow/white actions are unobserved, red actions were tried out and yielded negative reward. The majority of actions however are discarded by educated guessing (dark red), since they are classified as yielding no reward beforehand.

that do not result in immediate game progress or only display (specifically chosen) comments. However, in this example the method of educated guessing reduced the search space from 121 possible combinations to 35, i.e. by over 70%. The same 11 items could also be used on 18 different object targets in this scene, but over 80% do not require active execution checking for game progress, since they are already filtered out by educated guessing. However, when fully exhaustive exploratory game testing is required, developers and testers can disable this filter at run-time.

Dialog
The dialog choice system differs systematically from the previously mentioned actions. Dialogs are temporary, usually occur only in situations in which no other actions are possible and the set of options is very limited, namely to the number of dialog alternatives that are available at any give step. In each frame on the main loop, ICARUS assesses whether the game state is in a dialog or not before choosing an action or a dialog part. If the game is in a dialog state, a dialog option is chosen at random or in a traversing fashion, depending on the configuration.

Hints
Before ICARUS selects an action, but after observing all scene targets and inventory items, it will check if the situation fits to one of the hints (scripted actions) that can be manually defined in the configuration. Each of the two action archetypes (dialogs and actions) has its own hint table. E.g., *MGHints* contains walkthrough-like actions for puzzles that have an extremely low probability of being solved without context-sensitive, graphical, or time-dependent comprehension. ICARUS will favor an action above all else if the situation at hand matches the situation specified in a *MGHint* with the following general structure:

The current scene name is exactly "*SCENE*".
The target "*TARGET_OBJECT*" exists.
All conditions in the table "*CONDIS*" are met. No condition in the table "*NEG_CONDIS*" is true.
All values of "*LIST OF VALUE-CONSTRAINTS*" are met.
All items of "*LIST OF NECESSARY ITEMS*" are held.
All items of "*LIST OF FORBIDDEN ITEMS*" are not held.

Using this mechanism, in extreme cases, a complete game iteration can be executed deterministically by a hard-coded sequence of hints, since ICARUS will not explore the remaining game actions as long as the current situation fits to the execution of a hint. This can help with troubleshooting fixed processes as every game iteration has the same action sequence (e.g. in system compatibility or performance testing).

Web Control Panel
Since ICARUS is written completely in the script language *Lua* that is implemented by the engine, it can be applied to games running from the game engine editor as well as to complete builds, without the need of external programs or tools. However, to achieve more comfortable control over the most important parameters at run-time, to visualize the technical view of target objects, items, and possible item-item or item-object combinations, to provide a current snapshot of the debug log, and to provide access to the complete shortand long-term memory reward map, the system comes with a *web control panel* (see Figures 3, 5, and the supplementary video figure). It is implemented running a local web server on the testing machine, which can be assessed while the same

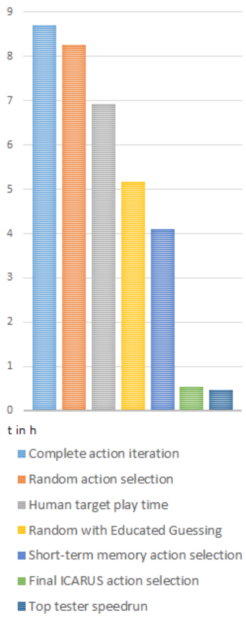


Figure 7. Completion times of the game *The Pillars of the Earth* by ICARUS employing several different features and approaches, compared to human playthroughs.

computer is actively testing or remotely, to simplify the observation, control, and management of multiple testing iterations on several machines.

Completion time

Figure 7 depicts the time different agents required for one game iteration of the game *The Pillars of the Earth* that is currently under development. The complete and random action selection versions of ICARUS which do not include *educated guessing* or *reinforcement learning* serve as a baseline for the more elaborate solving algorithms. They complete the game in about the time a human player needs who has no prior experience with the game, its puzzles, and is exposed to the content for the first time. The heuristic filtering of *educated guessing* cuts away about 40 % of the time required, whereas using *educated guessing* together with *short-term reinforcement learning* cuts the time required for completing the game by more than than half. Combining all of the introduced features (*educated guessing*, both *short-term* and *long-term reinforcement learning*, and *hints*), ICARUS can achieve a playthrough of the game in about 30 minutes, which is on the same level as the fastest speedruns of expert QA testers with prior experience of the game. Given that the game actions are supposed to be carried out at real-time, in an in-game, situated manner, these completion times can be considered near optimal and they are well suited for regular application.

Performance Tracking

In order to assess performance data about the game while playing, ICARUS can be configured to log the usage of RAM, VRAM, the time required to render the last frame (see: Figure 8), and further information of arbitrary kind into a persistent csv file. This tracking can take place simultaneously or independently from the basic solving process. If enabled, ICARUS will record one entry of performance data per frame, but save only the most extreme values of a given time window. In practice, each of the entries contains a time stamp, the name of the current scene, the current chapter, the last action and target that were chosen from ICARUS, and performance data: the time needed to render this frame in ms, as well as the amount of RAM and VRAM used in this frame in MB, before they get chunked to only the most extreme values, segmenting in steps of 1000ms. This implementation of performance tracking is novel in the sense that it is integrated in the process of automated solving, as well as being able to detect and report critical performance issues immediately (e.g. significantly high frame load time, or exhaust of available RAM/VRAM) in continuous comparison of the same game situation over many different hardware constellations and development versions, which proved to be of great use in its first application during the development of *The Pillars of the Earth*.

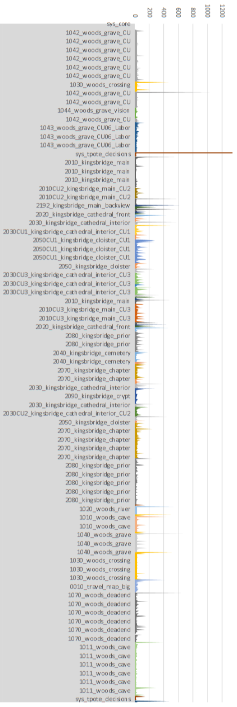


Figure 8. An example result listing of frame time tracking of *The Pillars of the Earth*. Different scenes are distinguished by coloring. Frame time is recorded in ms.

Bug detection

Returning to the initial issue of bug detection, the ICARUS system can support the detection and reporting in all of the major bug categories:

A Crashes/Freezes
Fully autonomous:
The tracking component of ICARUS will report immediately when the game crashes or stops rendering, thus it won't record any further data entries, displaying the exact time, scene and action that lead to the defect.

B Blockers
Fully autonomous:
When a predefined progress timeout is defined (e.g. 5 minutes) or the action space is empty at a non-busy point of time, ICARUS can detect if it is stuck in a game state that can not proceed any further.

C General
Semi-autonomous:
Aesthetic graphical, animation, sound, or spelling issues cannot be detected automatically using a logical solving algorithm. However the generic setup can be employed to play any adventure game, while human testers no longer have to concentrate on executing game actions. They can instead focus on spotting bugs of all categories more closely, monitoring multiple game sessions that are being played automatically at the same time. Furthermore, the explorative nature of ICARUS leads to the execution of actions that are potentially undiscovered by the regular testing procedures, often because they seem to be not intuitive or promising to lead to game progress, while still potentially containing or causing bugs.

DISCUSSION AND SITUATED USE

ICARUS is currently used for continuous integration by daily build validation at Daedalic Entertainment for all new adventure titles. The application of ICARUS supports the development teams in staying up to date with recent game alterations and content implementation through integrated testing. If any complete feature updates are committed to the shared repository, ICARUS will automatically test the build provided from the internal game build server, reporting issues if necessary. ICARUS is even more frequently applied in the continuous QA processes and test runs, where it aids testers by reducing workload, using the semi-autonomous approach. Furthermore, ICARUS is employed in the gold mastering of finalized games to check for changes of traversability and performance after games are completed to a shippable version. Finally, even large-scale hardware compatibility tests that are using remote hardware can be executed through ICARUS, as the first, external test of the game *The Pillars of the Earth* on 61 different hardware constellations and platforms demonstrated successfully. This does not only help with determining minimum hardware requirements, but also provides general insights into the impact of game mechanics, graphics and scene staging across a large number of systems.

Limitations and Future Work

In order to widen the field of applications for ICARUS and to be no longer constrained to a single game engine, the system is currenlty being extended to support further game engine environments, namely Unreal Engine and Unity.

> “Having a script like ICARUS that will simply try out different combinations and paths without getting exhausted or used to a specific way of playing the game helps focus QA resources on more complex tasks (testing of visuals and audio, game logic, etc.), making QA testing of the project more efficient and effective. Obviously, automated QA of parts or the entirety of a game can never completely replace human testers […]. However, it is definitely a valuable addition to our QA methods and helps to improve the overall quality of our games.”
>
> *- Maik Hildebrandt,*
> *Head of QA at Daedalic Entertainment* *[12]*

CONCLUSION

The ICARUS solver for adventure games has a proven track record as a significant enhancement for the quality assurance at Daedalic Entertainment. It can support developers and QA staff with tedious workflows, simplifying daily tasks and enabling performance comparisons across game iterations, game versions, and hardware constellations that might otherwise be prohibitively costly to execute. It can detect or aid the detection of all major bug categories, by either fully autonomous reporting or by allowing testers to focus closely on occurring bugs instead of being busy with executing game action sequences. The time needed for a complete game iteration is on the same level as professional game testers, thus no delays compared to prior development and QA processes are caused when using the system, while the time for implementing ICARUS in a completely new game project is also reasonable. Although some related work on automated non-technical testing solutions in games exists for clearly defined, template or macro-based scenarios, the generic nature, the ability to iterate through complete game iterations reliably, and the manifold features of tracking, visualizing and reporting, allow ICARUS to support game studios with establishing novel standards of QA, providing benefits to developers, publishers, and gamers alike.

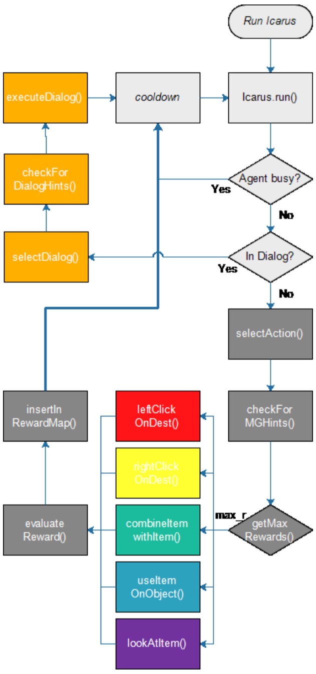


Figure 9. Flowchart of ICARUS

REFERENCES

1. Bob Bates. 2004. "Game Design".
2. Retrieved February. 2017. "The programming language Lua". https://www.lua.org/
3. Fazeel Gareeboo and Christian Buhl. 2012. "Automated Testing: A Key Factor For Success In Video Game Development. Case Study And Lessons Learned". http://www.uploads.pnsqc.org/2012/papers/t-26_ Gareeboo_paper.pdf EA Sports.
4. W.P. Judd and W.L. Heinz. 1997. "Universal automated training and testing software system". https://www.google.com/patents/US5602982 US Patent 5,602,982.
5. Mathieu Lachance. 2016. "How much people, time and money should QA take?". Retrieved March 29, 2017 from http://www.gamasutra.com/blogs/ MathieuLachance/20160113/263446/How_much_people_time_ and_money_should_QA_take_Part1.php.
6. Retrieved March. 2017. "Visionaire Studio". http://www.visionaire-studio.net/
7. Cyril Marlin. 2011. "Automated Testing: Building A Flexible Game Solver". http://www.gamasutra.com/view/feature/134893/ automated_testing_building_a_.php
8. Jim Merrill. 2016. "Automated testing for League of Legends". https://engineering. riotgames.com/news/automated-testing-league-legends Riot Games.
9. K. Peterson, S. Behunin, and F. Graham. 2012. "Automated testing on multiple video game platforms". https://www.google.com/patents/US20120204153 US Patent App. 13/020,959.
10. G.M. Pope, J.F. Stone, and J.A. Gregory. 1994. "Automated software testing system". https://www.google.com/patents/US5335342 US Patent 5,335,342.
11. U.H.H. Wild and M.I. Jabri. 1997. "System and method for automated testing and monitoring of software applications". https://www.google.com/patents/US5671351 US Patent 5,671,351.
12. Johannes Pfau. 2017. "Personal interview with Maik Hildebrandt"
13. B. Chan, J. Denzinger, D. Gates and K. Loose. 2004. "Evolutionary behavior testing of commercial computer games" In Evolutionary Computation, 2004. CEC2004. Congress on (Vol. 1, pp. 125-132). IEEE.
14. 14. Southey, Finnegan, Gang Xiao, Robert C. Holte, Mark Trommelen, and John W. Buchanan. 2005. "Semi-Automated Gameplay Analysis by Machine Learning." In AIIDE, pp. 123-128.
15. Alexander Zook et al.. 2014. "Automatic playtesting for game parameter tuning via active learning". Proceedings of the International Conference on the Foundations of Digital Games.
16. Martens, Chris. 2015. "Ceptre: A language for modeling generative interactive systems." In

Eleventh Artificial Intelligence and Interactive Digital Entertainment Conference.

17. Smith, Adam M., Mark J. Nelson, and Michael Mateas. 2009. "Computational Support for Play Testing Game Sketches." In AIIDE.
18. Osborn, Joseph Carter, April Grow, and Michael Mateas. 2013. "Modular Computational Critics for Games." In AIIDE.
19. Fernando de Mesentier Silva et al.. 2107. “AI as Evaluator: Search Driven Playtesting of Modern Board Games”. Proceedings of the AAAI 2017 Workshop on What's Next for AI in Games.
20. Daniel J.Simons and Ronald A.Rensink. 2005. “Change blindness: past, present, and future”. Trends in Cognitive Sciences, Volume 9, Issue 1 [p.16-20].

Deep Player Behavior Models: Evaluating a Novel Take on Dynamic Difficulty Adjustment

Johannes Pfau
Digital Media Lab, TZI,
University of Bremen
28359, Bremen, Germany
jpfau@tzi.de

Jan David Smeddinck
Open Lab, School of Computing,
Newcastle University
Newcastle upon Tyne, NE1 7RU, UK
Jan.Smeddinck@newcastle.ac.uk

Rainer Malaka
Digital Media Lab, TZI,
University of Bremen
28359, Bremen, Germany
malaka@tzi.de

ABSTRACT

Finding and maintaining the right level of challenge with respect to the individual abilities of players has long been in the focus of game user research (GUR) and game development (GD). The right difficulty balance is usually considered a prerequisite for motivation and a good player experience. Dynamic difficulty adjustment (DDA) aims to tailor difficulty balance to individual players, but most deployments are limited to heuristically adjusting a small number of high-level difficulty parameters and require manual tuning over iterative development steps. Informing both GUR and GD, we compare

CHI'19 Extended Abstracts, May 4–9, 2019, Glasgow, Scotland UK

ACM ISBN 978-1-4503-5971-9/19/05.
https://doi.org/10.1145/3290607.3312899

CCS CONCEPTS
Human-centered computing → User models
Computing methodologies → Neural networks
Applied computing → Computer Games

KEYWORDS
Dynamic difficulty adjustment;
Player Modeling;
Neural Networks;
Deep Learning;
Games

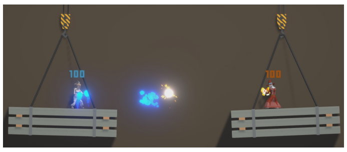


Figure 1: Screenshot of *Korona:Nemesis*. The player (on the left) utilizes Water to counter a Fire projectile.

Fire	Cancels **Restoration** Critically hits **Restoration/Steel** Destroys **Steel** projectiles Applies **burning** damage over time
Water	Immunity against **burning** Critically hits **Fire/Steel** Destroys **Fire** projectiles
Lightning	Immunity against **suffering** Critically hits **Water/Death** Destroys **Water** projectiles
Restoration	Restores 10LP Converts **Water** projectiles into 10LP Immunity against **Pain**
Steel	Reflects **Lightning** projectiles Reflects **Pain** projectiles Critically hits **Lightning/Pain**
Death	Inverts **Restoration** Critically hits **Restoration/Pain** Applies **suffering** damage over time
Pain	Self-ignites **Fire** Critically hits **Fire/Lightning** Applies 0.4 seconds stun

Table 1: Element interactions in the game.

an approach based on deep player behavior models which are trained automatically to match a given player and can encode complex behaviors to more traditional strategies for determining non-player character actions. Our findings indicate that deep learning has great potential in DDA.

INTRODUCTION

Dynamic difficulty adjustment (DDA) addresses potential mismatch between player proficiency and level of challenge in video games by balancing game parameters that increase or decrease the latter. Traditional approaches that manipulate core game variables (such as speed, damage or hit ratio), have been successfully evaluated and integrated in scientific [3] and industrial (e.g. Resident Evil 4)[1] usage. For practical reasons, DDA is usually hidden, since it yields incentives to perform badly on purpose [8]. Current DDA systems are typically limited to a small number of high-level parameters and require careful tuning of threshold-heuristics [10]. Here, we utilize *Deep Player Behavior Models* (DPBM) [6] to introduce a distinct adaptation module that incorporates player proficiency implicitly instead of explicitly and represents and generates game proficiency on a multi-dimensional level, allowing for complex emergent dynamics. In order to investigate the player experience with DPBM for DDA, we designed *Korona:Nemesis*, a platform fighter focused on prediction, learning and decision making. In an exploratory study, we compared player experience when playing against opponents with different decision making strategies including *basic* heuristics, *random* actions, near-*optimal* heuristics, and DPBM. Based on *self-determination theory* [7], we hypothesize that opponents deploying DPBM-guided strategies yield high results in *interest-enjoyment*, due to displaying convincing, but not rigidly perfect strategies, while *tension-pressure* might be increased and *perceived-competence* might be decreased when facing near-optimal opponents. Both are expected to lead to higher motivation and better player experience than traditional, trivial or unadjusted opponents. Our results provide first evidence that DPBM for generating opponent behavior confirms our hypotheses and offers a valuable subject to study within the field of DDA. We contribute to game user research in the form of a novel take on DDA and promote the applicability and value of machine learning techniques in video games.

RELATED WORK

DDA has developed from flow maximization [3] over multi-player balancing [11] up to a tool for proficiency estimation [2]. In order to estimate the discrepancy between challenge and skill, various assessment techniques have been researched, such as success probability estimation [3] or biofeedback [4]. For the adjustment however, most approaches focus on adjusting game difficulty parameters. In the meantime, machine learning approaches in video game playing that harness continual improvement through simulated play [9] have become popular. Bringing these developments together, we assess the experience of players that provide behavior samples feeding a continuous learning process, facing opponents driven by DPBM on the same proficiency level.

When facing an incoming **Fire** projectile, there are multiple viable choices. The player might react with a **Water** attack, since **Water** projectiles destroy **Fire** projectiles (cf. Figure 1). A more offensive choice would be to counter this attack with a **Pain** attack, which will not stop the incoming projectile (and thus cost 10LP), but critically hit and self-ignite the opponent. At the same time, the opponent has the opportunity to re-counter this counter-attack, depending on making good predictions (e.g. if (s)he predicts the counter-attack to be a **Water** attack and wants to counter it with **Lightning**, but in fact it is a **Pain** attack, it will incur a critical hit).

Sidebar 1: Decision making example.

variable	value
timestamp	12/27/2018 5:16:29
mapID	Map_Steel6
playerCurrentEnergy	WATER
playerChosenAction	ATTACK_WATER
playerHPpercentage	100
playerIsBurning	0
playerIsSuffering	0
targetCurrentEnergy	FIRE
targetHPpercentage	100
targetIsBurning	0
targetIsSuffering	0
absoluteXdistance	12.194
absoluteYdistance	0.211
fireProjectileAhead	1
waterProjectileAhead	0
lightningProjectileAhead	0
steelProjectileAhead	0
deathProjectileAhead	0
painProjectileAhead	0

Table 2: Sample player model entry for the situation given in Figure 1.

GAME DESIGN

In order to construct a setting for studying crucial decision making in real-time, we designed a fast-paced physic-based platform fighter called *Korona:Nemesis* that extends the classic rock-paper-scissors scheme to 7 types of element projectiles (cf. Table 1). In each level, players are placed in a 2D environment, start with 100 life points (LP) and have the objective to eliminate their opponents LP (last player standing wins). Players can *move* (left or right), *jump*, *attack* or *switch* actions. *Switching* changes the current stance to one of the 7 elements. *Attack* will launch an elemental projectile depending on the current stance. Getting hit by a hostile projectile deals 10 damage. Since damage is doubled on a critical hit and projectiles can be destroyed, reflected or influenced by other projectiles (cf. Table 1), players constantly have to be aware of present projectiles, their own and enemies' stances and adapt quickly to the situation. As in rock-paper-scissors, predicting the opponent is key to success and since players adapt and react constantly, there is no single dominant strategy (e.g., cf. Sidebar 1). Players need to learn not only the in-game element-interactions, but also their preferred way to counter attacks and maximize their chances, depending on the current situation. The presence of multiple viable choices, preferences and dislikes makes for a fertile ground for player modeling and decision making studies.

ENEMY TYPES

To focus on the players' experience of the opponents' decision making, enemies differed only in terms of their action selection behavior (and appearance), so possible action and movement choices, damage calculation, elemental interactions etc. were equal between all opponent types. The following categories of opponents were pseudonymized in-game to prevent revealing their strategies.

Basic. The basic opponent choses (and stays with) a single elemental stance per level. It is designed to be the easiest to counter since all actions are trivially predictable and serves as a baseline.

Random. The most balanced enemy in rock-paper-scissors is a completely random one. We decided to include this strategy as it is impossible to predict and thus hard to counter, since every action is independent from the preceding behavior or the current situation. Due to the symmetrical setup of the game, it will make both advantageous and disadvantageous decisions and should therefore not be (near-)impossible to beat.

Optimal. In order to provide an upper bound of performance, the optimal opponent reacts to each player action with one of the optimal counter-attacks and tries to maximize the damage applied to the player.

Player model. Utilizing every player action (together with game state context) executed in the preceding levels, the DPBM opponent will learn from the player's behavior and calculate weights for each possible action, whenever it chooses an action. Depending on the situation, it will make

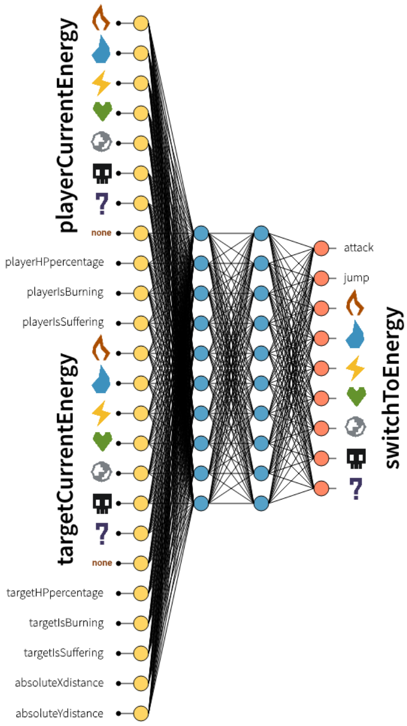


Figure 2: Network used for a single player. Real valued variables are mapped to the range from 0 to 1, energies and conditions are binary.

decisions similar to the player from whom the behavior originated. In this novel DDA approach, the opponent will continually develop while the player learns and advances in proficiency and it will make similar mistakes to the player. In contrast to traditional approaches, players simultaneously have to overcome and exploit their own flaws to win, potentially leading to an upward spiral of learning in both the player and the opponent. The ideal win/lose outcome would be an even split, demonstrating the closeness to the player's skill.

PLAYER MODELING

Based on insights about expressive data and suitable modeling techniques from our earlier work [6], we recorded all crucial player action decisions (*attacking* or *switching* with or to the respective element and *jumping*) together with contextual data from the current situation (cf. Table 2). After each level, this data was fed at run-time into a 24x10x10x9 multilayer perceptron with backpropagation and a logistic sigmoid activation function (cf. Figure 2). The network was initialized randomly and its architecture was determined beforehand, selecting for an efficient trade-off between training time (< 1 second on tested machines) and prediction accuracy (70-90% on testing set).

PILOT STUDY

Over the course of 2 weeks, we conducted a within-subjects study online. Subsequently to the tutorial, the experiment manipulated one independent variable (opponent behavior) with four conditions in randomized order: *basic*, *random*, *optimal* and *player model*. Data was gathered through game protocols and a post-study questionnaire.

Measures. In-game, we logged winning scores of all enemies, all of the players' actions and the resulting deep learning accuracies. Through the questionnaire, demographics and experience in video games were recorded. With respect to each specific enemy type, we asked for subjective assessments how strong and how balanced the particular opponent appeared, captured the player experience using the *Intrinsic Motivation Inventory* [5] (all 7-point Likert scales) and asked players to explain the opponent behaviors in their own words. Conclusive comments, questions and registering an email address for further studies were optional.

Procedure. Following informed consent and a quick tutorial that explained the controls and interactions of the game, participants encountered all four enemy types in permuted order. Each enemy was faced in the first 10 levels of the game, which were kept simple in order to focus on the opponent. Pausing the game was possible at all times and happened whenever the enemy type changed. After facing all of the opponents (in $M = 12.5$ minutes), the subject was redirected to the web questionnaire and unlocked the multiplayer mode (not part of the study).

	basic	random	optimal	player model
Score	4.8 ± 1.9	4.4 ± 2	7.8 ± 1.2	3.3 ± 1.9
strength	3.7 ± 1.9	4.5 ± 1.4	6.1 ± 1.1	4.3 ± 1.7
balance	2.8 ± 1.6	3.5 ± 1.4	3.7 ± 1.8	4.2 ± 1.9
IMI:				
INT	3.1 ± 1.6	3.7 ± 1.3	3.5 ± 1.8	5 ± 1.6
COMP	3.9 ± 1.9	3.6 ± 1.2	2.8 ± 1.6	4.9 ± 1.9
EFF	4.6 ± 2.2	5.5 ± 1.3	6.2 ± 1	5.6 ± 1.9
TEN	3.7 ± 2	4.8 ± 1.7	5.8 ± 1.1	4.6 ± 1.9

Table 3: Mean statistics ± standard deviations for the four enemy types. Score depicts the number of wins of the opponent. Strength and Balance were subjectively reported. INT: interest-enjoyment, COM: perceived-competence, EFF: effort-importance, TEN: tension-pressure of the IMI.

basic	*"repetitive"* *"some kind of predictable"*
random	*"changing his strategy/weapon very often"* *"unpredictable"*
optimal	*"very strong and fast"* *"always one upping me"* *"i had no chance and i hate him"* *"too OP"* (overpowered)
player model	*"kinda like the* [optimal opponent], *but not as OP"* *"a mixture of the other opponents"* *"my favorite so far, he was smart and fast but not too powerful"*

Table 4: Qualitative statements by players about the different opponents.

Participants. (n = 98) participants submitted behavioral data and 16 completed the optional questionnaire (75% male, 18% female, aged 18-37 ($M = 26.6, SD = 4.86$)). 75% described themselves as active, 19% as casual or occasional gamers and 6% said that they do not really play video games.

RESULTS

Using a one-way RM ANOVA, we found significant effects for the IMI scores *interest-enjoyment* ($F = 3.88, p < .05$), *perceived competence* ($F = 3.74, p < .05$), *tension-pressure* ($F = 3.47, p < .05$), as well as perceived *strength* ($F = 5.66, p < .01$), between opponents. These outcomes were further evaluated using two-tailed paired t-tests. Regarding the perceived strength, the optimal opponent significantly outmatched all other types ($p < .01, d_{basic} = .96; d_{random} = 0.76; d_{dpbm} = 1.01$). In terms of *perceived competence*, the *player model* resulted in higher values than the random ($p < .05, d = .72$) or optimal ($p < .05, d = .93$) opponent. For *interest-enjoyment*, DPBM significantly outperformed all of the other approaches ($p < .01, d = .75$ for basic, $p < .05, d = .57$ for random and $p < .05, d = .54$ for optimal). Means and deviations are depicted in table 3. When asked to explain the enemies' behavior in their own words, participant statements reflected these sentiments (cf. Table 4). Split into 80/20 training/test sets individually, neural network accuracy scored 49.1% to 100% ($M = 70.3\%, SD = 13.5\%$).

DISCUSSION AND FUTURE WORK

As hypothesized, the mean subjective strength with DPBM lies between basic and random/optimal, though no significant difference was found. The same holds for the mean subjective balance, exceeding all other enemy types. It did not lead to significantly increased *tension-pressure* and *effort-importance* compared to the other strategies, which might be due to the already high temporal pressure of the general gameplay and the short session duration. Nevertheless, the significantly higher score for *interest-enjoyment* indicates a distinct advantage of playing against the player model. It also apparently avoids frustrating players by displaying overly strong (and rigid) behavior, which is reflected in the significantly decreased *perceived competence* when facing the near-optimal opponent. These interpretations are supported by the qualitative statements, in which all positive comments relate to the DPBM approach and all negative ones to the remaining enemy types. This exploratory study is limited by a small sample size and the low conversion rate from participants who played the game to actually submitting the questionnaire. In further ongoing work, we will lay more emphasis on the questionnaire to consolidate the findings concerning player experience. We are also planning to extend the insights of this short-term study to a prolonged period of time to evaluate the long-term consistency of the approach. With this early study, we provided an experimental comparison between DPBM opponents and heuristic ones, yielding evidence for potential to improve DDA capabilities in general. A comparison to alternative traditional DDA approaches remains future work. We plan to investigate the difference in player experience between player modeling and threshold-based parameter adjustments.

In addition, assuming that every player desires a continually learning opponent is a simplification. Further studies that differentiate between player types might yield additional insights.

CONCLUSION

We compared the player experience of facing a continually learning enemy based on *Deep Player Behavior Models* to three classic heuristic game opponent variants. To provide an adequate study environment, we designed the platform fighting game *Korona:Nemesis*. First quantitative and qualitative results indicate significant improvements in player experience when interacting with the DPBM opponent. Thus, this approach successfully demonstrates a novel, implicit take on DDA and corroborates the potential application of DPBM in complex and fast-paced real-time game environments.

ACKNOWLEDGMENTS

We would like to thank all participants. This work was funded by the German Research Foundation (DFG) as part of Collaborative Research Center (SFB) 1320 EASE - Everyday Activity Science and Engineering, University of Bremen (http://www.easecrc.org/), subproject H2.

REFERENCES

[1] Capcom Production Studio 4. 2005. *Resident Evil 4.* Game [Gamecube].

[2] Simon Demediuk, Marco Tamassia, William L. Raffe, Fabio Zambetta, Florian "Floyd" Mueller, and Xiaodong Li. 2018. Measuring Player Skill Using Dynamic Difficulty Adjustment. In *Proceedings of the Australasian Computer Science Week Multiconference (ACSW '18).* ACM, New York, NY, USA, Article 41, 7 pages. https://doi.org/10.1145/3167918.3167939

[3] Robin Hunicke and Vernell Chapman. 2004. AI for Dynamic Difficulty Adjustment in Games.

[4] Changchun Liu, Pramila Agrawal, Nilanjan Sarkar, and Shuo Chen. 2009. Dynamic difficulty adjustment in computer games through real-time anxiety-based affective feedback. *Int. Jrnl. of Human-Computer Interaction* 25, 6 (2009), 506–529.

[5] E McAuley, T Duncan, and V V Tammen. 1989. Psychometric properties of the Intrinsic Motivation Inventory in a competitive sport setting: a confirmatory factor analysis. *Research quarterly for exercise and sport* 60, 1 (1989), 48–58.

[6] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2018. Towards Deep Player Behavior Models in MMORPGs. In *Annual Symp. on Computer-Human Interaction in Play Ext. Abstracts (CHI PLAY '18).* ACM, New York, NY, USA, 381–92.

[7] Scott Rigby and Richard M Ryan. 2011. *Glued to games: How video games draw us in and hold us spellbound: How video games draw us in and hold us spellbound.* ABC-CLIO.

[8] Andrew Rollings and Ernest Adams. 2003. *Andrew Rollings and Ernest Adams on game design.* New Riders.

[9] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, and Laurent Sifre et al. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529 (2016), 484–489.

[10] Alexander Streicher and Jan D. Smeddinck. 2016. Personalized and Adaptive Serious Games. In *Entertainment Computing and Serious Games*, Ralf Dörner, Stefan Göbel, Michael Kickmeier-Rust, Maic Masuch, and Katharina Zweig (Eds.). Lecture Notes in Computer Science, Vol. 9970. Springer International Publishing, Cham, 332–377.

[11] Rodrigo Vicencio-Moreira, Regan L. Mandryk, and Carl Gutwin. 2015. Now You Can Compete With Anyone: Balancing Players of Different Skill Levels in a First-Person Shooter Game. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15).* ACM, New York, NY, USA, 2255–2264. https://doi.org/10.1145/2702123.2702242

The Case for Usable AI: What Industry Professionals Make of Academic AI in Video Games

Johannes Pfau
University of Bremen
Germany
jpfau@tzi.de

Jan David Smeddinck
Newcastle University
Newcastle upon Tyne, UK
jan.smeddinck@newcastle.ac.uk

Rainer Malaka
University of Bremen
Germany
malaka@tzi.de

ABSTRACT

Artificial intelligence (AI) is a frequently used term – and has seen decades of use – in the video games industry. Yet, while academic AI research recently produced notable advances both in different methods and in real-world applications, the use of modern AI techniques, such as deep learning remains curiously sparse in commercial video games. Related work has shown that there is a notable separation between AI in games and academic AI, down to the level of the definitions of what AI is and means. To address the practical barriers that sustain this gap, we conducted a series of interviews with industry professionals. The outcomes underline requirements that are often overlooked: While academic (games) AI research tends to focus on problem-solving capacity, industry professionals highlight the importance of "usability aspects of AI": the ability to produce plausible outputs (effectiveness), computational performance (efficiency) and ease of implementation (ease of use).

CCS CONCEPTS

• **Computing methodologies → Artificial intelligence**.

KEYWORDS

Video games; industry; survey

ACM Reference Format:
Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2020. The Case for Usable AI: What Industry Professionals Make of Academic AI in Video Games. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Due to its steady growth in popularity and accessibility, the video game industry has evolved into a multi-billion dollar sector that surpassed all other entertainment industry areas, including TV, cinema and music[1]. Along with this development, the industry is constantly advancing, harnessing progress – or even trying to build a competitive edge based on innovations – in various fields, e.g. visual rendering, player experience, network stability or hardware progression. However, when it comes to *artificial intelligence* (AI) methods, only a small minority of shipped games harnesses recent scientific advancements [31, 37]. Large proportions of video games involve strategic decision making, optimization processes or competition, which make up fertile exploration grounds for AI, while many games even accumulate the vast amounts of data required for e.g. deep learning techniques. This is constantly demonstrated by successful integration examples, where – in the reverse direction – scientific research on AI for games frequently builds on industrial games, as in game playing [1, 17, 35], automated testing [22, 26, 34] or balancing [9, 20], world-building [27, 36], dynamic difficulty adjustment (DDA) [8, 24, 25] or player modeling [7, 19, 21, 23]. Yet, few cases of scientific AI have been applied in commercially successful video games, most of the time only when the AI itself constitutes part of the game's core mechanics [37], such as in the reinforcement learning of the companion animal in *Black and White* [12], the DDA features in *Halo*[2] [2] or *Left 4 Dead* [33] or the imitation learning (*Drivatar*) of *Forza Motorsport* [32]. This notable disparity can be related to the significantly differing definitions between scientific and industrial game AI. While a definition for scientific AI may be expressed as *"the theory and development of computer systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages"*[3], in the context of video games AI is more frequently seen along the lines of: *"artificial intelligence consists of emulating the behavior of other players or the entities [...] they represent. The key concept is that the behavior is simulated. In other words, AI for games is more artificial and less intelligence. The system can be as simple as a rules-based system or as complex as a system designed to challenge a player as the commander of an opposing army"* [10]. While this may partially be the result of different evolutions of understandings and the applied development of AI, it can arguably also contribute as a potential cause to forming and maintaining considerably isolated spheres of different understandings and schools of AI.

Additionally, due to market-forces in industry, information about utilized algorithms and techniques is not explicitly detectable and often not published for reasons of intellectual property and exploitation avoidance. To investigate the stance of AAA-developers on integrating promising AI techniques into video games, we contacted 105 of the currently most successful game companies, asking to conduct a semi-structured interview concerning their current use and requirements for applicable AI. This paper contributes to game research and development in academia and industry by providing

[1] https://newzoo.com/insights/trend-reports/newzoo-global-games-market-report-2019-light-version/ . Accessed 3.7.2020

[2] http://www.gamasutra.com/gdc2005/features/20050311/isla_01.shtml . Accessed 3.7.2020.

[3] https://en.oxforddictionaries.com/definition/artificial_intelligence. Accessed 3.7.2020

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

a qualitative analysis of (n = 9) industrial developers as well as derived guidelines that AI techniques can build on in order to be considered applicable.

2 RELATED WORK

Makridakis [13], as well as Skilton [28], predict that video games businesses will be highly impacted by the rise of deep learning and the embedding of AI in more and more facets of daily routine. Frutos et al. review the implementation of AI techniques within the scope of the serious game genre and point out a lack of applied AI methods, even if most of the applications originated in academia [4]. Togelius highlights the value of video games as ideal testbeds for academic AI, while on the other hand, this AI could significantly improve game mechanics and player experience [31]. According to Yannakakis, the sparse deployment within industrial productions is mainly caused by *"the lack of constructive communication between academia and industry in the early days of academic game AI, and the inability of academic game AI to propose methods that would significantly advance existing development processes or provide scalable solutions to real world problems"* [37]. Lara-Cabrera et al. note that the industry is beginning to *"adopt the techniques and recommendations academia offers"*, based on public reports from the respective companies [11].

All of these accounts reflect missed opportunities of the industry and incorporate or introduce methods that could fit into industrial implementation, yet no official statements from the actual target group (i.e. industry representatives) have been included so far. We argue that including explicit industry voice can contribute to achieving less one-sided discussions. Therefore this work aims to start contributing to closing this unresolved gap by providing qualitative insights from video game industry professionals based on a semi-structured interview concerning academic game AI.

3 STUDY

In order to obtain a broad impression of the industry's stance on scientific AI, 105 of the currently most successful game companies were contacted for a digital, semi-structured interview. After a period of six weeks and two additional reminders, (n = 9) responses could be collected that served for an outcome-oriented structuring content analysis [15].

3.1 Measures

Initially, participants stated their affiliated game project(s) and company. Subsequently, they were asked about the AI methods that are frequently utilized for development or game mechanics within their projects, followed by details about the type of algorithm or implementation. Beyond this, they stated their personal opinion with respect to the use of further AI techniques and outlined reasons why these are not (yet) incorporated.

3.2 Procedure

Companies were approached through publicly available contact points such as general inquiry mail addresses or instant messengers of community contacts. To avoid demanding personal information or addresses of the developers in order to allow less constrained reflection, the participation requests contained a request to be forwarded internally to representatives for game AI, machine learning, data analysis or development in general. Following informed consent, participants were able to express themselves freely within an online survey. Eventually, they were free to submit name and affiliation or to anonymize their participation.

3.3 Participants

In total, (n = 9) participants of at least seven different companies (including *Croteam*, *Crytek*, *Obsidian Entertainment*, *Paradox Development Studio*, *Harebrained Schemes*) completed the digital survey. Three of these decided to submit their data anonymously.

4 OUTCOMES

All of the surveyed participants agreed that the successful integration of pathfinding in more or less every modern video game since algorithms like A* [5] are cheap in computation, reliable and compelling, conditions which were relayed as clear requirements for consumer environments. Furthermore, unlike many of the other fields of AI, pathfinding is essential for video games to prevent totally idiosyncratic behavior, which led to a very early establishment in the industry. Another frequently mentioned technique are *Finite State Automata* (FSA) [16], for their robustness and observability, despite lacking any higher level capability of reasoning. Developers state that they use them for *"Movement state machines, etc." (P6)*, *"Character action sequences and combat" (P4)* or *"a lot of tasks not considered AI, like managing states of User Interface widgets" (P3)*, fulfilling predictable tasks far removed from the potential of more elaborate AI approaches. *Dynamic difficulty adjustment* [8, 30] is reportedly roughly applied with heuristics like *"[opponents] will start to miss more after managing to hit the player too rapidly" (P6)*, while the same holds also for reasoning systems, which are mostly reduced to frugal decision making about movement (*"e.g. to find out what a good position to shoot from will be, considering things like line-of-fire, distance to target, minimal distance from current position, closeness to allies, etc." (P7)*, *"Most of our AI is still reactive, but we have systems that 'sample' positions in the world for things like: get good attack position, cover spot, etc" (P6)*. Knowledge bases for NPC are elementary but common, incorporating known versus unknown facts, e.g. in *"computer player's knowledge of the game state (where other units are on the map)" (P3). Procedural Content Generation* (PCG) has found it's place in the game industry, not least because of games that are completely centered around it (e.g. *Minecraft* [18], *Spore* [14] or *No Man's Sky* [6], but also in regular games that are not completely focused on PCG, mostly for *"Worldbuilding" (P2)* or *"[generating] in-game content, like making trees at design time" (P3)*. Multi-agent interaction is stated to be a discipline that can improve game quality in a thorough manner, which is why many companies try to come up with good solutions, e.g. *"NPCs can decide to perform a complex attack together" (P4)*, *"One AI charges a player, while the team members give covering fire" (P6)*, albeit drawing on FSA for these decisions. The reasons for the sparse and conservative use of academic AI are shared among the industry:

> *"So far, our AI systems are mostly reactive and driven by behavior trees [3] that receive signals from events that happen in the world. The reason for this is that we need to model explicit rules in their behaviors to make the AI readable and* **"fun"** *for the player. Also, we need*

> *to do this using **limited CPU bandwidth** and in a way that these systems are **debuggable**" (P6).*

When asked about their personal position with respect to academic AI, they agreed that it bears a considerable potential and is of interest (for both developers and players). They also attribute capabilities for making the environment more believable, yet the surveyed experts are weary that academic AI comes with a notable implementation and configuration effort that typically result in the industry focusing on heuristic workarounds. According to our sample, the underlying mindset is best summarised in the terms of the surveyed professionals, implicitly reflecting requirements of the industry:

> *"What we call "AI" in games is vastly different than what's used in academia, or in business/ engineering/ apps/ ... Due to specific requirements like suspension of disbelief, games need a tighter control of possible outcomes and cannot afford the situation to be wildly misinterpreted. [...] Using decision trees, goal oriented action planning[4], and similar is found in some games, but we still largely rely on hand-tuned conditions controlled by hard-coded ifs, state machines etc. If you care more about **"plausibility"** than "intelligence", experience shows that hand-tuned solutions go a long way further than emergent ones. Also, consider the fact that **performance** budget is severely limited especially if there's a large number of actors. E.g we once experimented with a very elaborate goal-oriented action planning algorithm heuristic for gunfight tactics (choosing cover, targets, ...) where things like e.g. flanking were emergent results of the simple base logic resting on data like cover positions, precision estimation, etc... The results were impressive, but way too expensive. [They] could still produce unexpected results in some cases. When you consider that most games in that genre do away with prescripted actions for each possible scene, saving an order of magnitude on performance - and guaranteeing no unexpected behavior, you realize that there's still a long way to go for "real AI" in games." (P1)*

Apart from that, several participants highlighted the considerable labor effort that comes with implementation, adjustment and quality assurance:

> *"In order to make AI a noticeable feature where towns are full of interacting NPCs or where enemies are executing complex strategy, a company has to dedicate probably a dozen or more programmers/designers for over a year to set it all up, which is **very expensive**. Also, the more complex the AI, the more bugs that are created which reduces the polish of the game. We would love to have awesome villager AI with life like daily routines, but it's just too cost prohibitive." (P4)*

P5 brings up that industry and academic AI pursue different goals and that scientific advances do not necessarily lead to improved player experiences:

> *"As game AI is focused on creating entertainment rather than primarily solve problems (which academic AI typically does), and usually has much stricter constraints on **performance** than academic AI, it is often faster to custom-build solutions rather than use academic approaches. It also appears to be largely cheaper to produce a solution that fits the game and is "correct enough" than actually implement a method that produces a correct result. I think for most game AI developers, the interest in using academically developed AI goes as far as it can improve specifics in AI behaviour **reliably** and within budget (both development resources as well as CPU and memory)." (P5)*

Eventually, in order to actually ensure an improved player experience, developers conclude that this works best when AI techniques constitute central game mechanics, so that players actually perceive the added value:

> *"I think there are some opportunities to do more "advanced" AI in video games, but, it probably means that these games needs to be build and designed "around" these systems to make them really shine." (P6)*

5 DISCUSSION

Overall, the responses to the survey gave uniform insights on which AI techniques are popular, suitable or even necessary for modern games (e.g. pathfinding, FSA, PCG) and why other academic advancements are not trivial to adapt for the industry yet (e.g. machine learning, multi-agent reasoning systems, natural language processing). Summarized, the recorded statements can inform the development of more applicable academic AI techniques, e.g. through adding purpose-build middle-ware / services, by providing design guidelines that expect **plausibility/believability**, **computational performance** and **ease of implementation** in order to be applicable and recognized by the industry. These factors notably relate to the foundations of usability in efficiency, effectiveness and ease of use [29]. Ideally, these approaches should also be evaluated for **player experience** to justify the considerable effort and estimate the impact and implications on the game and its players. In effect, scientific submissions that contribute or benchmark novel AI techniques and aim to provide solutions that are applicable in industry, or translational research that aims to investigate the applicability of existing AI techniques in real-world contexts should evaluate and report their applicability with reference to these design requirements beyond the more common focus on successful problem-solving.

6 LIMITATIONS AND FUTURE WORK

The most notable limitation of this work remains the small number of interviewed experts, due to a considerably sparse response rate. Following from this, the interviewed companies will likely not cover all video game genres, and based on the publicly visible profiles are constrained to represent (offline) first-person shooters (FPS), role playing games (RPGs) and real-time as well as turn-based strategy games. While it can be argued that most of the mentioned issues and requirements can also be found in e.g. online FPS, massively multiplayer online RPGs or multiplayer online battle

[4] http://alumni.media.mit.edu/ jorkin/goap.html . Accessed 3.7.2020

arenas (MOBAs), we aim to extend this study to a larger group. An additional bias might have been the recruitment method of the participants, as developers only answered if they had the temporal capacity, the company policy allowed the publication of inside knowledge and they were able to follow and answer the English language of the study. In the follow-up evaluation, we are looking forward to working with a group of experts that is large enough to amount to a meaningful sample that is more representative for the industry and the diverse requirements of different genres, as well as to include quantitative measures and supplementary sources of information, such as public industry reports (e.g. *Gamasutra*[5], blog entries, or the Game AI Summit from the *Game Developers Conference*[6]).

7 CONCLUSION

Academic (game) AI researchers agree that video games provide expedient testbeds for algorithms, benchmarks and data aggregation, while video games could simultaneously benefit from the considerable advancements academic (game) AI continues to establish. Nevertheless, the use of modern and advanced AI techniques by video game companies remains limited, if the resulting games are not explicitly centered around these techniques. Using qualitative semi-structured interviews, this paper reveals the most crucial reasons cited by industry professionals and subsequently extracts requirements that novel AI approaches should meet in order to be applicable for industrial use. Developers expect that AI does not harm the **plausibility/believability** of NPCs (but ideally elevates it), the techniques have to be **easy to implement**, debug and adjust, they should not increase the game's **computational performance** requirements significantly and the added value of **player experience** should be proven. This work contributes to game AI research and development in academia and industry in pursuit of a closer integration of both areas.

ACKNOWLEDGMENTS

We would like to thank *Croteam*, *Crytek*, *Obsidian Entertainment*, *Paradox Development Studio*, *Harebrained Schemes*, *Black Shell Media* as well as all anonymous participants. This work was funded by the German Research Foundation (DFG) as part of Collaborative Research Center (SFB) 1320 EASE - Everyday Activity Science and Engineering, University of Bremen (http://www.easecrc.org/), subproject H2.

REFERENCES

[1] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. 2019. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680* (2019).

[2] Bungie. 2001. *Halo: Combat Evolved.* Game [XBox, PC]. Bungie, Chicago, IL, USA.

[3] Alex J. Champandard and Philip Dunstan. 2019. The Behavior Tree Starter Kit.

[4] Maite Frutos-Pascual and Begoñya García Zapirain. 2015. Review of the use of AI techniques in serious games: Decision making and machine learning. *IEEE Transactions on Computational Intelligence and AI in Games* 9, 2 (2015), 133–152.

[5] Peter E Hart, Nils J Nilsson, and Bertram Raphael. 1968. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics* 4, 2 (1968), 100–107.

[6] Hello Games. 2016. *No Man's Sky.* Game [PC, PS4, XBoxOne]. Hello Games, Guildford, UK.

[7] Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis. 2016. Evolving models of player decision making: Personas versus clones. *Entertainment Computing* 16 (July 2016), 95–104. https://doi.org/10.1016/j.entcom.2015.09.002

[8] Robin Hunicke. 2005. The case for dynamic difficulty adjustment in games. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology.* ACM, 429–433.

[9] Alexander Jaffe, Alex Miller, Erik Andersen, Yun-En Liu, Anna Karlin, and Zoran Popovic. 2012. Evaluating competitive game balance with restricted play. In *Eighth Artificial Intelligence and Interactive Digital Entertainment Conference.*

[10] Donald Kehoe. 2009. Designing artificial intelligence for games. *URL https://software. intel. com/en-us/articles/designing-artificial-intelligence-for-gamespart-1* (2009).

[11] Raúl Lara-Cabrera, Mariela Nogueira-Collazo, Carlos Cotta, Antonio J Fernández-Leiva, et al. 2015. Game artificial intelligence: challenges for the scientific community. *Proceedings 2st Congreso de la Sociedad Española para las Ciencias del Videojuego, Barcelona, Spain, June 24, 2015.* (2015), 1–12.

[12] Lionhead Studios. 2001. *Black & White.* Game [PC]. Lionhead Studios, Guildford, UK.

[13] Spyros Makridakis. 2017. The forthcoming Artificial Intelligence (AI) revolution: Its impact on society and firms. *Futures* 90 (2017), 46–60.

[14] Maxis. 2008. *Spore.* Game [PC]. Maxis, Redwood Shores, CA, USA.

[15] Philipp Mayring. 2010. Qualitative inhaltsanalyse. In *Handbuch qualitative Forschung in der Psychologie.* Springer, 601–613.

[16] Warren S McCulloch and Walter Pitts. 1943. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics* 5, 4 (1943), 115–133.

[17] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (2015), 529.

[18] Mojang. 2011. *Minecraft.* Game [PC]. Mojang, Stockholm, Sweden.

[19] Nathan Partlan, Abdelrahman Madkour, Chaima Jemmali, Josh Aaron Miller, Christoffer Holmgård, and Magy Seif El-Nasr. 2019. Player Imitation for Build Actions in a Real-Time Strategy Game. *AIIDE workshop on Artificial Intelligence for Strategy Games.* (2019).

[20] Johannes Pfau, Antonios Liapis, Georg Volkmar, Georgios Yannakakis, and Rainer Malaka. 2020. Dungeons & Replicants: Automated Game Balancing via Deep Player Behavior Modeling. In *2020 IEEE Conference on Games (CoG).* IEEE.

[21] Johannes Pfau, Jan David Smeddinck, Ioannis Bikas, and Rainer Malaka. 2020. Bot or not? User Perceptions of Player Substitution with Deep Player Behavior Models. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems.* ACM.

[22] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2017. Automated game testing with icarus: Intelligent completion of adventure riddles via unsupervised solving. In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play.* ACM, 153–164.

[23] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2018. Towards Deep Player Behavior Models in MMORPGs. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play.* ACM, 381–392.

[24] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2019. Deep Player Behavior Models: Evaluating a Novel Take on Dynamic Difficulty Adjustment. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems.* ACM, LBW0171.

[25] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2020. Enemy Within: Long-term Motivation Effects of Deep Player Behavior Models for Dynamic Difficulty Adjustment. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems.* ACM.

[26] Stefan Radomski and Tim Neubacher. 2015. Formal Verification of Selected Game-Logic Specifications. *On Engineering Interactive Computer Systems with SCXML* (2015), 30.

[27] Noor Shaker, Julian Togelius, and Mark J Nelson. 2016. *Procedural content generation in games* (1st ed.). Springer Publishing Company, Incorporated.

[28] Mark Skilton and Felix Hovsepian. 2017. *The 4th industrial revolution: Responding to the impact of artificial intelligence on business.* Springer.

[29] International Organization For Standardization. 1998. *ISO 9241-11 - Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs): Part 11: Guidance on Usability.* ISO, Berlin.

[30] Alexander Streicher and Jan D Smeddinck. 2016. Personalized and adaptive serious games. In *Entertainment Computing and Serious Games.* Springer, 332–377.

[31] Julian Togelius. 2015. AI researchers, Video Games are your friends!. In *International Joint Conference on Computational Intelligence.* Springer, 3–18.

[32] Turn 10 Studios. 2005. *Forza Motorsport.* Game [XBox]. Turn 10 Studios, Redmond, WA, USA.

[33] Valve. 2008. *Left 4 Dead.* Game [PC]. Valve, Bellevue, WA, USA. Played 2017.

[5] https://www.gamasutra.com/ . Accessed 3.7.2020.

[6] https://www.gdconf.com/ . Accessed 3.7.2020.

[34] Simon Varvaressos, Kim Lavoie, Sébastien Gaboury, and Sylvain Hallé. 2017. Automated bug finding in video games: A case study for runtime monitoring. *Computers in Entertainment (CIE)* 15, 1 (2017), 1.

[35] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* 575, 7782 (2019), 350–354.

[36] Georg Volkmar, Nikolas Mählmann, and Rainer Malaka. 2019. Procedural Content Generation in Competitive Multiplayer Platform Games. In *Joint International Conference on Entertainment Computing and Serious Games*. Springer, 228–234.

[37] Geogios N Yannakakis. 2012. Game AI revisited. In *Proceedings of the 9th conference on Computing Frontiers*. 285–292.

Do You Think This is a Game? Contrasting a Serious Game with a Gamified Application for Health

Johannes Pfau
Digital Media Lab, TZI
University of Bremen
Bibliothekstraße 1,
28359 Bremen, Germany
jpfau@tzi.de

Georg Volkmar
Digital Media Lab, TZI
University of Bremen
Bibliothekstraße 1,
28359 Bremen, Germany
gvolkmar@uni-bremen.de

Rainer Malaka
Digital Media Lab, TZI
University of Bremen
Bibliothekstraße 1,
28359 Bremen, Germany
nwenig@tzi.de

Jan David Smeddinck
ICSI, UC Berkeley
1947 Center St,
Berkeley, CA 94704
jandavid@icsi.berkeley.edu

Nina Wenig
Digital Media Lab, TZI
University of Bremen
Bibliothekstraße 1,
28359 Bremen, Germany
malaka@tzi.de

CHI'18 Extended Abstracts, April 21–26, 2018, Montreal, QC, Canada
ACM 978-1-4503-5621-3/18/04.
https://doi.org/10.1145/3170427.3188651

Abstract

The general promise of employing the motivational power of games for serious purposes, such as performing physiotherapy exercises, is well-established. However, game user research discusses both the approach of *gamification*, i.e. adding game-elements on to a task-focused application and of *serious games*, i.e. injecting task-focused elements into a more fully-fledged game. There is a surprising lack of empirical work that contrasts both approaches. We present both a *casually gamified application* and a *serious game* with purpose-driven mechanics that provide different frontends to the same underlying digital health application. This application aims at supporting physiotherapy sessions for chronic lower-back afflictions. Results from an explorative pre-study contrasting both approaches indicate a clear preference for the *serious game* version, capturing higher perceived motivational components (autonomy and relatedness), as well as higher immersion and flow relative to the *gamified* version.

Author Keywords

Games for Health; Exergames; Serious Games; Gamification; Motivation; Motion-based Games; Gameful Design

ACM Classification Keywords

K.8.0 [Personal Computing]: General - *Games*

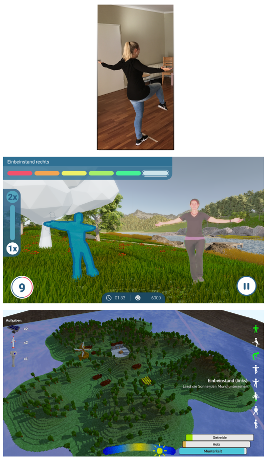


Figure 1: The *stand on one leg* exercise in both versions. **CGA** *(top)* displays the users outline and rewards the exercise execution quality with points. In **MMW** *(bottom)* this exercise shifts the position of sun and moon to control day and night time, building up the *alacrity* resource.

Introduction

Harnessing *the power of games to draw players in and keep them spellbound* [8] to motivate exercising for fitness or therapy has developed from tinkering prototypes to commercial developments and a frequently studied subfield. While other aspects are also receiving attention, motivation is arguably the defining outcome for most research efforts and developments [9]. A central design decision that has to be tackled in every individual approach to utilize this motivational power is then whether to (a) consider a game predominantly and *add exercises in*, or to (b) consider the exercises predominantly and to *add game elements on*. This decision also applies to other contexts (e.g. education) and following the terminology by Deterding et al. [5], would result either in (a) a *serious game*, or (b) *gamification*. While the differences in design approaches and expected outcomes have been discussed in related work, there is a surprising lack of empirical work based on contrasting literature or different implementations.

In this paper we present the motion-based game for health *Move My World (MMW)* that features rich resource-based world-building strategy gameplay with therapy exercises 'added in'. In a pre-study MMW is compared to a more *casually gamified application (CGA)* where players follow movements presented by an instructor figure while placed in an appealing virtual environment and receiving ratings, badges, etc. Based on related literature and *self-determination theory* (SDT) [8] we hypothesize that MMW should lead to higher motivation (especially regarding autonomy as an aspect of intrinsic motivation) and immersion / flow [3].

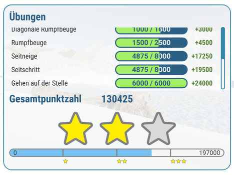
The outcomes provide first evidence that – while participants liked both applications – MMW did indeed result in higher perceived freedom / autonomy and was clearly preferred in a free-to-choose final play session. We contribute both to motion-based games for health, presenting a serious game with purpose-driven game mechanics, as well as to more general *game user research* (GUR), furthering debates on *shallow* vs. *deep* gamification and serious games.

Related Work

Taxonomies have been developed in the larger GUR space to facilitate more nuanced discussion and growing a structured understanding of the different approaches. Deterding et al. [5] define gamification as *the use of game design elements in non-game contexts* and provide a delimitation between gamification (using parts of games) and (serious) games (full-fledged or whole games). The potential and benefits of serious games in general and motion-based games for health in particular are beginning to be more well-understood and researched [9]. However, it is important to notice that the terms are frequently used interchangeably or with a different understanding and it is surprising to see that although many applications with a focus of motivating motion-based exercises are self-proclaimed serious games, or games for health, and although the categorization is not always entirely clear, using the terminology by Deterding et al. many fall under the label of gamification (e.g. [1, 2, 6]). Our *CGA* can be seen as roughly representative of such a common approach to gamified motion-based health applications. While it can be argued that typical gamification elements (such as scores, badges, etc.) are also parts of many full-fledged games and thus they are not mutually exclusive from serious games, or could simply be understood as different *levels of gamification*, we argue that there are important differences in the approaches, as discussed by Deterding et al. and that it is important to establish, whether these theoretical differences result in measurable and predictable outcomes. Notably, there can indeed be different levels or intensities with which gameful

Exercise	Game Impact
rotation	Raises the height of a helicopter to get an overview of the island.
boxing	Commands the villagers to gather resources by chopping wood.
bend to toes	Trigger rain (water fields).
stand on one leg	Raises/sets the sun/moon.
circle hips	Brings up wind (run windmills).
side step	Operates crane (erect buildings).
walking in place	Raise the speed of all villagers.

Table 1: In-game effects triggered when executing the respective exercises in **MMW**.



Übungen

Diagonale Rumpfbeuge	1000 / 1600	+3000
Rumpfbeuge	1500 / 2500	+4500
Seitneige	4875 / 8000	+17250
Seitschritt	4875 / 8000	+19500
Gehen auf der Stelle	6000 / 6000	+24000

Gesamtpunktzahl 130425

Figure 2: Final screen of **CGA**. The user gets feedback about every individual exercise performance, as well as the total score, represented by stars.

elements are applied to non-gaming contexts (shallow or deeper gameful design), including borderline cases such as 'framification' [7], and with which elements of serious purpose are applied to games (serious game design) that warrant further study but are not subject to comparative study in this work. Since games necessarily need to be understood as inseparable wholes [4] the specific choices regarding these design aspects have to be treated as fixed independent variables in the study design. Similarly, related work also discusses delimitations between *gameful* (or ludic) and *playful* (or paidic) approaches that are not primary subjects of this research while standing in complex interaction with the use of games in action [4].

Gamification and Serious Game Design

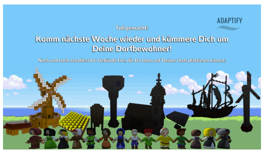
As indicated above, many current exergame applications are either focused on the proper execution of exercises in a rather casually gamified setting, or on the other extreme designed entirely as games for entertainment (e.g. Wii Fit/Sports, Dance Dance Revolution, Kinect Sports, EyeToy Games, etc.), lacking the incorporation of actual therapeutic exercises. To enable the comparative study we employ a gamification and a serious games version of an application for the support of physiotherapy and the application use-case of chronic lower back afflictions. Both applications implement the same configurable as well as exchangeable set of exercises (cf. Figure 1) that represents a subset of a lower back treatment plan developed in cooperation with physiotherapists in the context of the project *Adaptify*.

CGA represents a predominantly exercise focused gamification approach. Exercises are presented in a linear order, preceded by a tutorial video. Users have to perform the correct gestures in a given time window in order to proceed, following the guidance of an instructor character that is presented next to the their real-time body outline. Depending on the quality of the execution (i.e., the proximity to the ideal set of movements that constitute the exercise), detected repetitions are displayed in a color-coded fashion, from red (worst performance) to green (best performance). Continuous good executions can increase a multiplier that is used to calculate a total score. Starting with a hidden background, increasing the score unlocks parts of a pleasant virtual scenery. An end-screen rewards users with a number of stars, depending on the performance (cf. Fig. 2).

MMW resembles an economy simulation god-game where the user has to take care of the population of a procedurally generated island. Per session, one main mission has to be completed, which is achieved through subtasks, such as *"Provide food for your villagers"* by constructing a set of houses, fields and windmills and make them work. In an embedded, interactive tutorial the mayor of the town presents the exercises required in this session and how they influence the world when executed (cf. Figure 1). The user, however, is free to choose the order in which the subtasks are completed. He can, for example, choose to construct all required buildings first and then perform different exercises subsequently or he can postpone the residual execution of an exercise to a later point of time to focus on other subtasks. The sum of these subtasks corresponds to the underlying set of exercises and is dynamically adapted to changing difficulties/repetitions/holding periods. For example, if the generated mission requires four buildings to be constructed and each building needs wood to be built, then the wood collection time for a single building is calculated by dividing the total time that was defined as required for *standing punches* by the number of buildings in the mission. In that way, the user can e.g. choose to finish *standing punches* halfway, then spend time on other exercises, and finally return to the residual wood for the remaining two buildings. This approach enables freedom of choice while

Resource	Usage/Source
wood (accumulates)	Needed to construct buildings. Gained by chopping wood.
alacrity (acc.)	Needed to enable villagers to work. Gained by sleep (trigger night-time).
wind (temporary)	Effect to actuate mills/turbines. Produced via *circle hips*.
water (temp.)	Effect to grow fields. Produced via *circle hips*.
grain/power (acc.)	Needed to complete the respective main mission.

Table 2: Resources which accumulate in reservoirs and effects that are triggered temporarily in **MMW**.



Toll gemacht!

Komm nächste Woche wieder und kümmere Dich um Deine Dorfbewohner!

Nach und nach schaltest Du Gebäude frei, die Du dann auf Deiner Insel platzieren kannst:

Figure 3: Final screen of **MMW**. Users are incentivized to stick with an exercise plan consistently through unlocks for new buildings, missions, and the possibility of developing an individual island.

still ensuring that the minimal amount of time/repetitions for all exercises is satisfied.

Instead of a score system, *MMW* uses a resource management approach (see Figure 2). The required resources are automatically adjusted to reflect a given set and repetitions / durations of exercises, but players can freely determine the order. The resources are displayed at all times, as well as the progress of each individual exercise, the remaining subtasks and the main goal. If the latter is completed, all villagers come together in the village center to celebrate and thank the player. Afterwards, a final screen is presented, showing the success of the current session and further unexplored content that can be unlocked (see Figure 3). This deep integration between game elements and the serious purpose can be described as purpose-driven (or purposeful) mechanics and aims at producing a predominantly gameplay-driven experience.

Both *CGA* and *MMW* feature a complete sound design and were tested and developed to comparable standards, employing iterative testing for quality assurance, as well as the same underlying technology stack for player tracking, exercise detection, and audiovisual rendering.

Comparative Exploratory Pre-Study

To compare both approaches in terms of motivational effects and flow, a within-subjects study was conducted in a laboratory setting. The experiment manipulated one independent variable with two conditions: *gamified application (CGA)* and *serious game (MMW)*. Data was gathered through questionnaires and a post-study semi-structured interview with an emphasis on qualitative methods to facilitate capturing unforeseen aspects.

Measures
An initial questionnaire asked for demographics and experience in video games and sports. A post-trial questionnaire after each game aimed to capture appreciation, motivation through items based on SDT [8] (asking for perceived competence [perceived performance], autonomy [freedom], relatedness [relatable characters]), as well as flow and immersion [3], all indicated through 7-point-Likert scale statement agreement. In the end, a semi-structured interview invited free responses along the same categories, asking participants to contrast both gameplay sessions. Observational notes about problems, remarks and execution flaws were taken throughout the sessions, indicating no notable technical problems or difficulties executing exercises.

Setup and Procedure
Following informed-consent and the pre-study questionnaire participants interacted with both *CGA* and *MMW* in permuted order. In both cases subjects were asked to stand in front of a screen on a marked spot. After completing each regimen that was scheduled to last about 10 minutes and featured the same exercises, they were asked to respond to the post-trial questionnaire. Following the comparative interview after the second trial, where participants were free to add any ideas and thoughts, they were told that a final play session was required. This time they were able to choose whether they wanted to play *CGA* or *MMW*.

Participants
The study included 7 convenient subjects (4f, 3m), 20 to 62 years of age ($M{=}39.14$, $SD{=}16.96$). They indicated ($M{=}7.60$, $SD{=}7.77$) hours of playing video games in a typical week on average. Prior experience with games, sports and physiotherapy is displayed in Table 3.

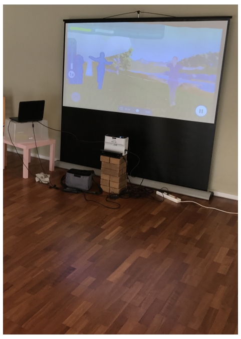


Figure 4: The hardware setup was consistent between both versions. Users faced a 240x135cm screen driven by an ultra-short distance projector. A Microsoft Kinect V2 tracked the users.

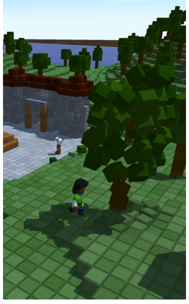


Figure 5: A villager chopping wood in *MMW*. This behavior is triggered when the user performs the associated exercise *boxing*.

Results

We report means and standard deviation but omit inferential statistics (low sample size) to avoid misinterpretation, although some results did indicate statistical significance in t-tests. Participants indicated that they liked both games overall (*CGA*: $M{=}6.57, SD{=}.53$; *MMW*: $M{=}6.86, SD{=}.38$) indicating that both were well-produced and received. Similar positive ($M >= 6$) ratings were also observed for perceived competence, physical wellbeing during exercise execution, and motivation. In *CGA* ($M{=}4.14, SD{=}2.04$) participants were less *"able to relate to the virtual characters"* than in *MMW* ($M{=}6.14, SD{=}.38$). Perceived *"freedom do as I please"* was notably lower in *CGA* ($M{=}3.57, SD{=}2.37$) than in *MMW* ($M{=}6.43, SD{=}.79$). Together these results indicate that SDT motivation differed based on aspects of relatedness and autonomy, but not competence.

Regarding how *"appropriate the challenge through the game"* was *CGA* ($M{=}4.43, SD{=}2.15$) received lower scores than *MMW* ($M{=}5.86, SD{=}.69$). *MMW* was also rated to feel more immersive ($M{=}6.00, SD{=}.82$) than *CGA* ($M{=}4.43, SD{=}2.23$). Accordingly, since balance between challenge and skill, as well as feeling immersed, are important facilitators of flow experiences, the overall experience of having *"a feeling of being in the game flow"* showed a lower mean for *CGA* ($M{=}3.71, SD{=}1.80$) than for *MMW* ($M{=}5.86, SD{=}1.07$).

Interview

Using their own wording, five participants stated they liked *MMW* more because of the *"deeper game mechanics"*, the *"time spent was perceived shorter"*, the *"nice setting"*, and the *"aspect of free choice"*. Only one participant preferred *CGA* because of the *"clear and linear task representation"*. 6/7 reported a higher level of competence in their exercise execution in *CGA*, because of the constant feedback in form of their silhouette. A sense of making decisions, playing at will, mentally appropriate challenge, immersion and flow appeared predominantly, or even solely, in *MMW*. All subjects stated that both prototypes certainly motivate them to perform physical exercises (in comparison to traditional physical therapy without digital assistance), but they strongly preferred *MMW* (5/7) in terms of expected long-term motivation (2/7 indicated no preference), because of the *"variation"*, *"unlockable game elements"* and the *"individual continuation of the game"*. Following the interview, subjects were asked to pick one of the versions to play a third session. 6/7 picked *MMW*, indicating they did so mostly *"out the curiosity for new buildings"* and the *"opportunity to advance their individual villages"*.

Discussion and Future Work

The interview responses clearly express an overall preference towards *MMW*, underlining indications from the questionnaires. Both perceived motivation based in SDT and flow / immersion appear increased compared to *CGA*. Since the setup and exercise selection was not varied this indicates a positive impact of a serious game approach with purpose-driven mechanics and exercises 'added in', compared to an exercise sequence presented by an instructor figure with game elements 'added on'. The higher perceived freedom and flow in *MMW* are likely driven by the more free nature of this game version. Players felt like they could choose which tasks they wanted to address and thus, which exercises they would perform. *CGA* provided a clear order of exercises, leading to a lower sense of freedom. Similarly, the fact that players did not have to perform a single exercise for a prolonged time in *MMW* can arguably not only contribute to higher perceived autonomy, but also support self-regulated balancing between the level of challenge and one's own situated skill. When feeling tired or bored, participants could simply choose a different task to pursue.

Gaming experience	
Non-gamer	1
Casual gamer	6
Advanced gamer	0
Exergame experience	
No prior experience	5
Prior experience	2
Sport habits	
0h sports per week	3
0-2h sports per week	1
2-4h sports per week	2
>4h sports per week	1
Membership in a sports group	
Currently not	6
Currently engaged in a sports group	1
Never been	3
Have been in the past	4
Experience with following an instructor	
No experience	4
Prior experience	3
Experience with physiotherapy	
Never received physiotherapy	3
Received physiotherapy before	4

Table 3: Participants' prior gaming and sports experience

The results warrant a follow-up study with larger participant numbers, an extended duration, employing the full psychometric questionnaires. Including a more radically open-ended / player-driven variant of *MMW* might also be promising, as it could extend the scope of the work to encompass more playful approaches. Furthermore, the situated use and the potential influence of player type will be considered in future work.

Conclusion

We compared a *serious game* and a *gamification* approach for the same underlying purpose of supporting physiotherapy exercises. Regarding motivation, immersion, and flow in a study contrasting the two representative prototypes. The *gameplay-focused resource managing strategy game* was clearly preferred over the alternative with common gamification elements (e.g. points, badges, etc.). Given the specific implementations this may be mainly attributable to the influence of meaningful elements such as making perceived own decisions constantly, relating to the game characters, an increased feeling of flow, and the individual and continuous development of the game world across sessions.

Acknowledgments

We would like to thank all participants. This work was funded by the *German Federal Ministry of Education and Research* (BMBF) through the project *Adaptify*. This work was partially funded through the *DAAD FITweltweit* program.

REFERENCES

1. Aimee L. Betker, Tony Szturm, Zahra K. Moussavi, and Cristabel Nett. 2018. Video Game-Based Exercises for Balance Rehabilitation: A Single-Subject Design. *Arch. of Phys. Medicine and Rehab.* 87, 8 (2018), 1141–49.
2. J. W. Burke, M. D. J. McNeill, D. K. Charles, P. J. Morrow, J. H. Crosbie, and S. M. McDonough. 2009. Optimising Engagement for Stroke Rehabilitation Using Serious Games. *Vis. Comput.* 25, 12 (2009), 1085–1099.
3. Jenova Chen. 2007. Flow in Games (and Everything Else). *Commun. ACM* 50, 4 (April 2007), 31–34.
4. Sebastian Deterding. 2014. Eudaimonic Design, or: Six Invitations to Rethink Gamification. (2014).
5. Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. 2011. From Game Design Elements to Gamefulness: Defining "Gamification". In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments (MindTrek '11)*. ACM, New York, NY, USA, 9–15.
6. Stefan Göbel, Sandro Hardy, Viktor Wendel, Florian Mehm, and Ralf Steinmetz. 2010. Serious Games for Health: Personalized Exergames. In *Proceedings of the 18th ACM International Conference on Multimedia (MM '10)*. ACM, New York, NY, USA, 1663–1666.
7. A. Lieberoth. 2015. Shallow Gamification: Testing Psychological Effects of Framing an Activity as a Game. *Games and Culture* 10, 3 (May 2015), 229–248.
8. Scott Rigby and Richard Ryan. 2011. *Glued to Games: How Video Games Draw Us In and Hold Us Spellbound*. Praeger.
9. Jan D. Smeddinck. 2016. Games for Health. In *Entertainment Computing and Serious Games*, Ralf Dörner, Stefan Göbel, Michael Kickmeier-Rust, Maic Masuch, and Katharina Zweig (Eds.). Lecture Notes in Computer Science, Vol. 9970. Springer International Publishing, Cham, 212–264.

Can You Rely on Human Computation? A Large-scale Analysis of Disruptive Behavior in Games With A Purpose

Johannes Pfau
University of Bremen
28359 Bremen, Germany
jpfau@uni-bremen.de

Rainer Malaka
University of Bremen
28359 Bremen, Germany
malaka@tzi.de

Abstract

Outsourcing effortful problems as microtasks has been successfully implemented by various human computation serious games or GWAP. Still, most of the academic approaches validate their results by conducting laboratory studies. While these have the potential to assess proposed techniques thoroughly with respect to quantitative and qualitative measures, they are prone to the often underestimated experimenter bias. In a large-scale field study $(n = 713)$, we collect practically relevant empirical data about the quality of player behavior in a human computation serious game and classify the results as **useful**, **insufficient** or deliberately **disruptive** executions. Due to the drastic proportion of disruptive behavior (20.2%), we particularize explanations for this kind of behavior and discuss counteracting measurements.

Author Keywords

Human Computation; Games With A Purpose; Disruptive Behavior

CCS Concepts

•**Human-centered computing** → **User studies;**

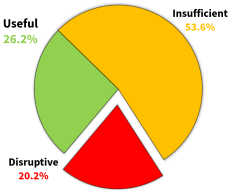


Figure 1: Useful, **insufficient** and **disruptive** behavior proportions.

CHI PLAY EA '19, October 22–25, 2019, Barcelona, Spain.
© 2019 Copyright is held by the author/owner(s).
ACM ISBN 978-1-4503-6871-1/19/10.
https://doi.org/10.1145/3341215.3356297

Introduction

Human computation bears great capabilities in distributing time-consuming and/or effortful tasks or calculations among a potentially vast audience. In this context, the deployment of games with a purpose (GWAP), serious games or gamification has proven to be able to wrap up the original task with game elements in order to harness the emerging motivation and to overshadow the actual effort. Nevertheless, the core tasks often remain laborious and the actual intention of most players is to maximize own enjoyment. Many studies give evidence about the usefulness of human computation serious games in laboratory studies, yet often lack scalability to real-world relevant magnitudes. In a broad study ($n = 713$) we let participants play a GWAP in the domain of everyday household activities and categorized the sessions into **useful**, **insufficient** and **disruptive** performances (cf. Figure 1). Compared to a laboratory pre-study, we found significantly higher amounts of **disruptive** performances and argue why this particular behavior is hard to inhibit. Concluding, we contribute to the research field of games user research, GWAP and human computation by stressing the influence of the experimenter bias on the validity and expressiveness of laboratory studies in this domain. Readers of this paper will receive an estimate about how seriously players actually take GWAP and guidelines about important factors in designing and studying human computation.

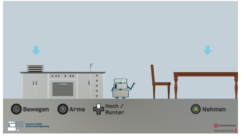


Figure 2: Screenshot of the human computation game *Learning with EASEY*, shortly before the execution of the *table setting* task.

Context

The most popular approach to human computation is undoubtedly *reCAPTCHA* [16], which nowadays simultaneously learns computer vision samples while serving protection from malicious automated web requests. Yet, since the beginning of its widespread deployment, users constantly came up with ways to bypass it (e.g. using image recognition for *reCAPTCHA's* Image Challenge, OCR for the Text Challenge or most successfully exploiting Google's own speech recognition API to solve the Audio Challenge). Assessing one of the most prominent GWAP for human computation, *Peekaboom* [15], Von Ahn et al. state that they do not suffer from malicious players or bots, since they are not disguising the underlying human computation microtasks as game elements but presenting the raw problem of visual object classification to the players, which is not solved by autonomous algorithms yet. However, when observing GWAP with deeper game mechanics, the opportunities for disruptive behavior increase. Curtis [2] examined the motivation of players of *Foldit*, an online GWAP for solving highly complex protein foldings. No disruptive behavior was mentioned in the mixed-methods evaluation, which might stem from the fact that *Foldit* bears a high threshold of participation through difficult tutorial levels that can take up to weeks to solve. Above that, 59% of the players stated that *Foldit* is the sole video game they play and the majority revealed that their intrinsic motivation is to make a contribution in science, not game playing. Siu and Riedl [12] gave evidence that the lack of intrinsic motivation can alter the quality of human computation games significantly, based on a high percentage of boredom (80%) in their cooking-based GWAP *Cafe Flour Sack*. They try to counteract this by introducing different reward mechanisms. In *OnToGalaxy* [4], players populate an ontology via a 2D space shooter in which correctly rescued ships are mapped to ontological subsumption concepts. The authors report trust issues, which is why they introduced a trust function invisible for players that consequently weights their score according to useful or undesirable behavior. This trust assessment was limited to two-player scenarios. For annotating image metadata in their game *PexAce* [11], Simko et al. also report on disruptive behavior in the form of a user that wrote an application that essentially rushed through their proposed game and finished it without any meaningful input in the solution.

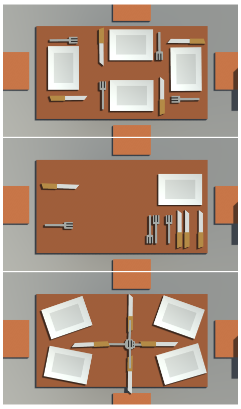


Figure 3: Sample solutions for the task *table setting*. The formal goal required players to move cutlery and dishes for 4 persons from the start position to a suitable configuration. Constraints were given in which each piece had to be moved at least once and all of them have to stay on the table. In effect, **useful** (top), **insufficient** (center) and **disruptive** (bottom) solutions were given.

Altogether, the existence and danger of disruptive behavior is recognized and party tackled. With the help of the everyday activity household GWAP *Learning with EASEY*, we assess the severity of this problem empirically and discuss countermeasures and explanations.

Overall, a common way of motivating GWAP players is to induce extrinsic motivation by rewarding them with points, levels, badges, leaderboards or avatar decorations [11, 2, 4, 15]. Yet, intrinsic motivation has been established as the core factor of engagement and enjoyment [10, 8], which actually can be inhibited by overmuch usage of extrinsic motivators [3, 7]. The underlying causes of intrinsic motivation are nevertheless highly subjective and can be hard to accomplish on a large scale. Thus, in every reasonably sized player base, disrupting behavior can be found as one of the major actuators of intrinsic motivation. This observation is backed by every major theory of player types that aim to explain subjective differences of motivation in players. Bartle's expanded categories of player types [1] includes the *Griefer*, who is most importantly engaged by spoiling progress of other players, Tondello et al. specifically mention the *Disruptor* in their Hexad scale [13] and Tseng [14] ascribes this kind of behavior to *aggressive gamers*.

Glitches, bug abuse and potential for undesirable behavior can reach severe magnitudes in the complex emergent systems of video games [5], where GWAP are not exceptional and not necessarily spared from disruptive players. Yet, many studies regarding GWAP do not ascertain or highlight the fact that disruptive behavior can confound the aggregated solutions significantly and that this effect can enfold large proportions of the player base. We argue that this neglection might stem from the substantively higher amount of laboratory studies in games user research and a significant impact of Rosenthal's experimenter bias [9] that exhibits the drastic alteration of participant behavior towards the experimenters hypothesis under laboratory settings.

Evaluation

In order to get an empirical estimate about the proportion of undesirable behavior in human computation games, we conducted a large-scale study in the domain of GWAP for everyday household activities. To prevent the aforementioned experimenter bias, we had to conceal the appearance of a scientific study and installed it into an open exhibition about robotics and artificial intelligence that took place in a publicly available exposition building. Over the course of 4 months, $(n = 713)$ participants took part in the evaluation. To contrast these findings to an experimenter biased group, we recruited $(n = 21)$ participants to execute everyday household human computation in a laboratory setting.

Procedure

Players of the game were introduced to the stylized robot character *EASEY* (cf. Figure 2) that asks for help in the domain of everyday household activities. They could choose from four different tasks including *table setting* (cf. Figure 3), *tidying up* (cf. Figure 4), *cooking* and *pouring water*. All of the given scenarios follow classical everyday household problems that address robotic manipulation parameter tuning, context adaptation, spatial optimization and personalization to individual preferences. Controls and task descriptions were visible at all times. Players could restart the level at any given time and only the actual final state that they confirmed was saved as the result.

Material

Subjects played on a 22" screen using an XBox One controller. After the respective level, the execution performance was assessed and a screenshot of the final result was

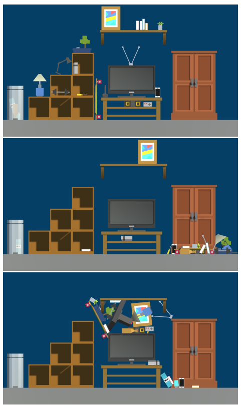


Figure 4: Sample solutions for the task *tidying up*. The formal goal required players to tidy up the cluttered room and differentiate trash from decorations. **Useful** (top) solutions managed to present a desirable execution, **insufficient** (center) were most likely not motivated enough and **disruptive** (bottom) solutions purposely arranged the room in undesirable configurations.

saved. Sessions that did not reach the goal state of the particular task or that were aborted prematurely were assigned to the **insufficient** category. In the case of completed goal states, a manual differentiation between **useful** and **disruptive** performances was selected by the experimenter due to the premise of the GWAP that no feasible autonomous way of telling these apart exists. For the purpose of minimizing the experimenter bias, we refrained from recording additional demographic information or quantitative questionnaires, since their appearance could have influenced bystanders even after the session of a single participant.

Results

Completion times for each level ranged between approximately 1 to 5 minutes. 26.2% of the sessions were completed in a **useful** manner for human computation, 53.6% resulted in **insufficient** outcomes and 20.2% of them were classified as showing **disruptive** behavior. In the laboratory control group, no **disruptive** behavior was shown at all. Participants retried each level until the formal goal requirements were met.

Discussion

Useful performances stem from players acting according to the intention of the underlying approach. They are instrumental for the success of human computation, but rely on the conscientiousness of the players to not fall into the remaining categories.

Insufficient completions are sessions that did not reach the required goal state or failed to fulfill other task constraints. Depending on the context and implementation of the serious game or application, they are rather easily recognizable and measures against this behavior are reasonably achievable. We argue that this incompleteness mainly stems from a lack of motivation to engage in the task and can be addressed by polishing game mechanics, content and quality. Following that, games resulting in a large proportion of **insufficient** results tend to be rather immature, not fully fledged and/or lack to make their players spellbound.

Disruptive behavior stems from the intrinsic motivation to go beyond the game's boundaries, explore weaknesses and flaws within the mechanics and/or to be successful in unconventional ways for this very reason. In conventional video games, this leads mostly to the exploitation of glitches or bugs and can harm the games' internal balancing, especially in the context of multiplayer games. In human computation games however, the results of this behavior can confound the validity of the whole application, distort the aggregated data and ultimately lead to an undesirable and potentially dangerous real-world interpretation. Compared to **insufficient** performances, they are also much harder to spot, since fraudulent behavior can nevertheless lead to a technically correct goal state, and even harder to inhibit, since these players intentionally want to screw the game up. The observation that **disruptive** behavior is not unusual, but rather common (20.2%) even in a large sample emphasizes the underlying harm. In order to counteract this, human computation games have to be vastly heuristically constrained to render commonly undesirable behavior invalid. Yet, this procedure confines the potential of human computation overall, since it might also restrict solutions that are novel but valid. Above that, if the complete procedure of solving a task correctly would exist in the first place, there would be little to no input left that could be aggregated through human computation.

In conclusion, we derive the following guidelines from our experiment: **Useful** behavior is highly desired in human computation and should be facilitated in a primarily intrinsically motivating fashion. To keep the proportion of this at a high level, **insufficient** behavior can be reduced by polishing the motivational aspects of the game in terms of quality and mechanics. However, **disruptive** behavior has to be approached differently. GWAP for human computation should optimize for a sweet spot between constrained heuristics and action freedom in order to keep the validity as well as the expressiveness at a high level. Moreover, cross-validation between different players are advisable, to assess the actual quality of the solution from another perspective. This procedure might also suffer from **disruptive** ratings but bears the capability to reduce the number of undesirable solutions in this two-step process. Additional approaches are discussed in the Future Work section.

Future Work

In order to get a valid estimate of the categories' proportions, we optimized the study design for the amount of participants and the strict avoidance of the experimenter bias. Nevertheless, we were not able to link these findings to particular players. Therefore, we seek to conduct a follow-up field study that however includes post-assessments about the player's type, his intrinsic motivation and qualitatively reflective statements about the own behavior. Within this study, we pay attention to evaluate groups of similar sizes, which was infeasible in the current setup. We are aiming at finding the actual correlation between personality and the quality of human computation solutions. Additionally, we are looking forward to find parameters and techniques for the autonomous classification of undesirable behavior using machine learning player model approaches [6]. Together with peer-reviewed cross-validations, these could establish a standardized *validity measure* that would augment the field of human computation.

Conclusion

The goal of this approach is to demonstrate the drastic influence of the experimenter bias on laboratory studies assessing human computation. This bias might render the utility of human computation approaches as valid or viable under the guise of laboratory study settings. However, real-world factors such as anonymity, actual intrinsic motivation, different player types and the absence of observation might lead to **disruptive** behavior that can distort the validity of human computation results significantly. We strengthen these hypotheses with the result of a large-scale field evaluation that resulted in a significant portion of **disruptive** behavior, while the laboratory control group was spared from it.

Acknowledgments

We would like to thank all participants of the study. This work was funded by the German Research Foundation (DFG) as part of Collaborative Research Center (SFB) 1320 EASE - Everyday Activity Science and Engineering, University of Bremen (http://www.easecrc.org/), subproject H2.

REFERENCES

1. Richard Bartle. 2003. A self of sense. *Available on April* 9 (2003), 2005.

2. Vickie Curtis. 2015. Motivation to participate in an online citizen science game: A study of Foldit. *Science Communication* 37, 6 (2015), 723–746.

3. Edward L Deci, Richard Koestner, and Richard M Ryan. 1999. A meta-analytic review of experiments examining the effects of extrinsic rewards on intrinsic motivation. *Psychological bulletin* 125, 6 (1999), 627.

4. Markus Krause, Aneta Takhtamysheva, Marion Wittstock, and Rainer Malaka. 2010. Frontiers of a

paradigm: exploring human computation with digital games. In *Proceedings of the acm sigkdd workshop on human computation*. ACM, 22–25.

5. Chris Lewis, Jim Whitehead, and Noah Wardrip-Fruin. 2010. What Went Wrong: A Taxonomy of Video Game Bugs. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games (FDG '10)*. ACM, New York, NY, USA, 108–115. DOI: http://dx.doi.org/10.1145/1822348.1822363

6. Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2018. Towards Deep Player Behavior Models in MMORPGs. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '18)*. ACM, New York, NY, USA, 381–392. DOI:http://dx.doi.org/10.1145/3242671.3242706

7. Daniel H Pink. 2011. *Drive: The surprising truth about what motivates us*. Penguin.

8. Andrew K Przybylski, C Scott Rigby, and Richard M Ryan. 2010. A motivational model of video game engagement. *Review of general psychology* 14, 2 (2010), 154–166.

9. Robert Rosenthal and Kermit L Fode. 1963. The effect of experimenter bias on the performance of the albino rat. *Behavioral Science* 8, 3 (1963), 183–189.

10. Richard M Ryan and Edward L Deci. 2000. Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary educational psychology* 25, 1 (2000), 54–67.

11. Jakub Simko, Michal Tvarozek, and Maria Bielikova. 2013. Human computation: Image metadata acquisition based on a single-player annotation game. *International Journal of Human-Computer Studies* 71 (10 2013), 933–945. DOI: http://dx.doi.org/10.1016/j.ijhcs.2013.05.002

12. Kristin Siu and Mark O Riedl. 2016. Reward systems in human computation games. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play*. ACM, 266–275.

13. Gustavo F. Tondello, Rina R. Wehbe, Lisa Diamond, Marc Busch, Andrzej Marczewski, and Lennart E. Nacke. 2016. The Gamification User Types Hexad Scale. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '16)*. ACM, New York, NY, USA, 229–243. DOI: http://dx.doi.org/10.1145/2967934.2968082

14. Fan-Chen Tseng. 2011. Segmenting online gamers by motivation. *Expert Systems with Applications* 38, 6 (2011), 7693–7697.

15. Luis Von Ahn, Ruoran Liu, and Manuel Blum. 2006. Peekaboom: a game for locating objects in images. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, 55–64.

16. Luis Von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. 2008. recaptcha: Human-based character recognition via web security measures. *Science* 321, 5895 (2008), 1465–1468.

Player Types and Achievements - Using Adaptive Game Design to Foster Intrinsic Motivation

Georg Volkmar
Digital Media Lab, TZI
University of Bremen
Bibliothekstraße, 1
28359 Bremen, Germany
gvolkmar@uni-bremen.de

Johannes Pfau
Digital Media Lab, TZI
University of Bremen
Bibliothekstraße, 1
28359 Bremen, Germany
jpfau@uni-bremen.de

Rudolf Teise
Digital Media Lab, TZI
University of Bremen
Bibliothekstraße, 1
28359 Bremen, Germany
teise-rudolf@hotmail.de

Rainer Malaka
Digital Media Lab, TZI
University of Bremen
Bibliothekstraße, 1
28359 Bremen, Germany
malaka@tzi.de

CHI PLAY EA '19, October 22–25, 2019, Barcelona, Spain.
© 2019 Copyright is held by the author/owner(s).
ACM ISBN 978-1-4503-6871-1/19/10.
https://doi.org/10.1145/3341215.3356278

Abstract

Intrinsic motivation is a key factor in facilitating enjoyment and engagement in video games. In recent years, various approaches have been introduced by the industry to raise motivation of players. One of the most prominent methods in modern games comes in the form of *Achievements* which are defined as optional meta-objectives that players can obtain by fulfilling tasks in the game. However, *Achievements* are designed uniformly regardless of the player's personality, play style or general preferences. Therefore, individual differences between players are ignored which can diminish the motivational impact of *Achievements*. To tackle this problem, this paper proposes the design of adaptive *Achievements* based on specific archetypes extracted from the *BrainHex* player typology. For the validation of this approach, we developed a simple Action-RPG which included adaptive *Achievements*. In a comparative study ($n{=}28$) we found that adaptation of *Achievements* leads to an increase of motivational aspects such as perceived effort/importance and sense of reward & individualization.

Author Keywords

Player Types; BrainHex; Achievements; Adaptive Games; Intrinsic Motivation.

CCS Concepts

•Applied computing → Computer games;

Introduction

In modern video games, *Achievements* or *Trophies* serve as symbols of accomplishment which can be unlocked by displaying a specific set of skills of the player and are defined by the game itself, the player/community or the game platform [1]. Introduced in 2005 by Microsoft for their XBox 360 system, *Achievements* have established themselves as small rewards aimed to raise player motivation and engagement as well as a way of socializing with other players [12]. However, in terms of motivation, *Achievements* can provide extrinsic rewards for players [10] which in turn may lead to a decrease of intrinsic motivation [7, 20]. Since intrinsic motivation is strongly related to engagement and enjoyment in any activity [25] including playing video games [22], extrinsically rewarding *Achievements* have the potential to act as a hindrance to a compelling player experience. To tackle this problem and enable *Achievements* to raise intrinsic motivation, we propose an approach to design them in an adaptive way. More precisely, this paper presents a game prototype which adapts *Achievements* according to specific player types. The idea of adapting game content to player types is based on previous research showing that this solution has the potential to create an adequate level of challenge and an overall improved player experience [5]. By displaying a way to implement adaptive *Achievements* based on player types to raise intrinsic motivation successfully, this paper contributes to the research domains of adaptive game design and player typologies.

Player Types and Adaptive Game Design

Categorizing players into types and play styles has been a research topic in the HCI community for years as it helps game designers to understand their target audiences and tailor content to individual players [28]. For this reason, many different models have been developed and evaluated over the years, based on geographic, demographic, psychographic or behavioral characteristics of players [11, 8].

The earliest approach to define a coherent player typology is known as the *Bartle Taxononmy of Player Types* [2]. This model categorizes players based on two axes (Acting - Interacting & World - Players) resulting in four types - *Achievers, Explorers, Socialisers, Killers.* In a refined version, another axis (Implicit - Explicit) was added and thus, eight sub-types emerged - *Planners, Opportunists, Scientists, Hackers, Networkers, Friends, Politicians* and *Griefers* [3].

Based on this fundamental approach and additional long-term data collection, Yee developed an advanced player typology [29]. Instead of categorizing players into distinct types, this model consists of motivational components that reflect player behavior and preferences. Overall, three overarching groups containing ten sub-elements were defined - *Achievement* (*Advancement, Mechanics, Competition*), *Social* (*Socializing, Relationship, Teamwork*) and *Immersion* (*Discovery, Role-Playing, Customization, Escapism*).

By clustering players into groups based on intensity, sociability and the actual games played, the *InSoGa* model defines three *Gamer Mentalities - Social Mentalities* (*Gaming with Kids, Gaming with Mates, Gaming for Company*), *Casual Mentalities* (*Killing Time, Filling Gaps, Relaxing*) and *Committed Mentalities* (*Having Fun, Entertaining, Immersing*) [13].

The *BrainHex* model was developed with the intention to combine previous research regarding player types and neurobiological insights in terms of player satisfaction [17]. In total, *BrainHex* describes seven archetypes - *Seeker* (curious to explore the game world), *Survivor* (enjoys game experiences associated with terror or fear), *Daredevil* (likes to take risks, seeking thrill), *Mastermind* (solving puzzles, identifying efficient decisions & strategies), *Conqueror* (overcoming challenging opponents including other play-

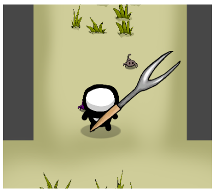


Figure 1: Player character carrying standard equipment at the start of the game.

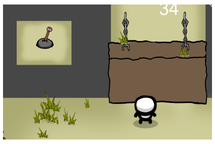


Figure 2: Progressing the level requires players to overcome obstacles such as lever-driven gates.

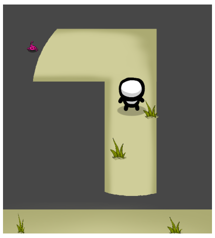


Figure 3: Hidden passageways can be found by moving across the level edges.

ers), *Socialiser* (talking, helping and spending time with people they trust) and *Achiever* (is focused on finishing objectives). In contrast to most other typologies, *BrainHex* doesn't assign a single exclusive type but categorizes players as combinations of these archetypes. More precisely, after undergoing a *BrainHex* test, the model assigns a *primary* and a *secondary* class.

In the domain of gamified learning environments, *BrainHex* was utilized to design and implement game elements according to player types, leading to a significantly higher level of engagement [16]. Moreover, adaptation based on *BrainHex* archetypes was applied in the context of persuasive health games in order to improve eating behavior [19, 18]. To this day, the empirical validation of *BrainHex* is still lacking [28]. However, this model shows great potential regarding the increase of motivation based on player type specific adaptation [4]. Therefore, it will serve as a foundation for the design of adaptive *Achievements* in this paper.

Game Description

For the *Achievements* to be included, we developed a simple Action-RPG with 2D graphics and a top-down perspective called *Forkknight*. Playing the game, players take control of a generic character equipped with a fork-like weapon trying to overcome various challenges included in the levels (see Figure 1).

Mechanics

The main objective of the game is to fight groups of rather weak enemies that appear in the form of hostile tomato-creatures and reach the end of the game where a boss has to be defeated. At the top of the screen, the game displays a health bar which is reduced by stepping on traps spread throughout the level or being hit by an enemy.

Health can be refilled by picking up specific items. If however, the player's health drops to zero, the level is restarted.

Progressing levels is not only achieved by killing off enemies but also by solving simple puzzles like finding switches that need to be activated to open a gate (see Figure 2).

While traversing the game, players can find items to upgrade their abilities and properties. These power-ups are hidden behind illusory walls (see Figure 3) and can increase the character's speed, enhance the health bar or enlarge the weapon which helps to keep enemies at a distance (see Figure 4).

The character is controlled via mouse and keyboard inputs. Using arrow keys or WASD alternatively, players can move in four directions. By moving the mouse in all directions, the fork-weapon results in a circular movement around the character.

In total, 14 levels have to be completed to finish the game. Each mechanic is introduced slowly in tutorial levels giving players the chance to get accustomed to each game element.

Adaptive Achievement Design

As *BrainHex* subdivides its archetypes into a *primary class* and a *secondary class*, the game displays *Achievements* for both categories. Overall, three *Achievements* are available for one playthrough, two of them represent the *primary class*, whereas the last one is based on the *secondary class*. Throughout the entire play session, they are displayed in the upper right corner of the screen (see Figure 5). Multiple *Achievements* have been developed for each type for the game to pick. For each type, one example will be presented in this section.

Seekers are asked to find a way to set the fork on fire by locating a bonfire (see Figure 6). The respective fire can be found behind an illusory wall and requires players to explore each level thoroughly. *Survivors* have to complete a specific challenge run in a level where spikes are moving towards the player who has to evade a number of traps to

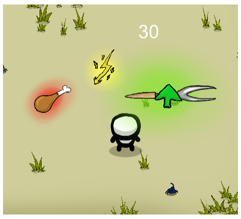


Figure 4: Power-ups to enhance health, speed or weapon size.

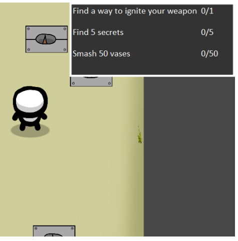

Find a way to ignite your weapon 0/1
Find 5 secrets 0/5
Smash 50 vases 0/50

Figure 5: Example showing how *Achievements* were displayed.

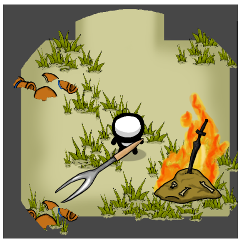


Figure 6: By finding a bonfire, players can light the weapon.

survive (see Figure 7). This section is a part of the game regardless of player types. However, for *Survivors* to unlock the *Achievement*, they have to finish a second, more dangerous iteration of this challenge. *Daredevils* have to complete ten levels in under 45 seconds which requires them to take many risks and should provide a thrilling experience. For the *Mastermind* to be rewarded with the *Achievement* and upgrades, a number of word-based puzzles have to be completed. To solve a single puzzle, three letters have to be entered in the correct order. Hints regarding which letters are to be pressed on the keyboard at which time are distributed at the level edges (see Figure 8). The mechanics of these puzzles are not directly explained but have to be figured out by the player hence providing a challenge for the *Mastermind*.

Since *Conquerors* enjoy the process of overcoming difficult enemies, one of their *Achievements* involves killing the final boss using an alternative method instead of simply attacking with the fork. In order to fulfill the *Achievement* successfully, they need to identify various traps that are placed around its arena as a way of damaging the boss (see Figure 9). As *Achievers* are motivated to complete tasks simply for the sake of completing them, their *Achievements* entail hitting certain objects a couple of times or destroying scene props such as vases scattered all over the game. For the research presented in this paper, we solely focused on singleplayer experiences. Therefore, no adaptions for the *Socialiser* class were implemented in the prototype. For the study depicted in the next section, data obtained from primary class *Socialisers* was planned to be assigned to the control group. However, this reassignment wasn't necessary in the actual experiment.

Evaluation

To examine the impact of player type-specific *Achievements* on players' intrinsic motivation, a between-subjects laboratory study was conducted. In the experimental group, subjects were exposed to adaptive *Achievements* whereas control group members were given random *Achievements* that didn't match their type.

Material

To play the game and answer the set of questionnaires, a standard gaming laptop with built-in keyboard and an external mouse were provided in both groups.

Items for the subsequent questionnaire were extracted from the *Intrinsic Motivation Inventory (IMI)* as it has been deployed successfully in previous studies related to intrinsic motivation [23, 27, 21, 15, 24, 26, 6]. Participants could respond to these questions with a 5-point Likert scale [14]. Additionally, another questionnaire was constructed asking for qualitative feedback and ways to improve the experience.

Participants

Subjects for the study were recruited with the help of social media posts and printed hangouts that were distributed around the university's campus. Therefore, most participants were undergraduate students aged between 20 and 30 years. Of all recruited subjects, three people were asked to conduct a small pilot-test to identify potential flaws in the study design or game-related bugs. The other 29 subjects conducted the actual experiment. No additional demographic data was recorded. Data gathered from one participant had to be excluded from the analysis due to a game-breaking error in the software. Therefore, results reported in this paper are based on the measurements from the remaining subjects $(n=28)$, 15 in the control group and 13 in the experimental condition.

Procedure

Upon giving informed consent by signing an according document, participants were asked to conduct a person-

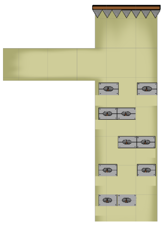


Figure 7: Spike-challenge requiring players to move downwards while evading traps.

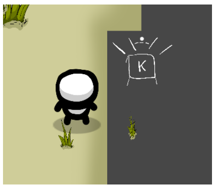


Figure 8: Hint for word puzzles that are spread around the level.

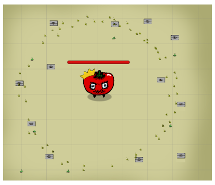


Figure 9: Final boss of the game that can be defeated by attacks or traps.

ality test based on the *BrainHex* questionnaire which can be found online [1]. Following this procedure, the examiner transferred the resulting values from the test to the game prototype. It is important to note that this process was conducted identically for both groups and without notifying participants which type was calculated for them from the test. This prevented people from guessing which group they might belong to and from priming them in terms of their player type which otherwise might have had an impact on their in-game behavior. Before subjects were exposed to the game itself, the examiner briefly introduced them to the game mechanics and gave the opportunity to ask any remaining questions. It was made clear that the objective of playing was to get to the end of the game while the *Achievements* were to be treated as optional sub-goals and not as mandatory end-states.

As the game was started, *Achievements* were automatically assigned to the player dependent on their group. For the experimental group, fitting *Achievements* were displayed whereas for the control group, unfitting ones were chosen randomly. Altogether, three *Achievements* were shown in the upper right corner of the screen, two of them representing the *primary class* and one taken from the *secondary class*. Each *Achievement* was designed as a plain descriptive text carrying a simple progress indication. Participants played the game for a maximum time span of 30 minutes. In case they beat the final boss prior to this limit, they were given the choice to play again (e.g. to complete any remaining *Achievements*) or to finish the test-phase. Ultimately, all subjects were asked to fill out a final questionnaire containing items from the *IMI* and additional questions aimed at qualitative feedback.

Results

For each category of the *IMI* questionnaire, a Student's t-test for independent samples was calculated to analyze differences between the control group and the experimental condition. Regarding *interest-enjoyment*, no significant differences between the control group ($M{=}3.61$, $SD{=}0.45$) and the experimental group ($M{=}3.52$, $SD{=}0.52$) could be identified, $t(26){=}1.01$, $p{=}0.32$. For *perceived competence*, no significant differences between the control group ($M{=}3.28$, $SD{=}0.95$) and the experimental group ($M{=}3.42$, $SD{=}0.84$) were found, $t(26){=}0.95$, $p{=}0.35$.

In terms of *tension-pressure*, the control group ($M{=}3.03$, $SD{=}0.86$) and the experimental group ($M{=}2.75$, $SD{=}0.63$) showed no significant differences either, $t(26){=}1.39$, $p{=}0.18$. However, concerning *effort-importance*, the experimental group ($M{=}3.87$, $SD{=}0.51$) was rated higher than the control group ($M{=}3.37$, $SD{=}0.88$), showing a statistically significant difference, $t(26){=}2.98$, $p{=}0.047$, $d{=}0.68$.

With respect to the questions specifically constructed for this evaluation, Student's t-tests for independent samples were calculated as well. We found significant effects asking for *"It feels like the achievements matched my personal preferences"* ($t(26){=}2.09$, $p{=}0.046$, $d{=}0.66$) and *"I tried hard to fulfill the achievements"* ($t(26){=}2.45$, $p{=}0.02$, $d{=}0.81$), as well as a highly significant difference for *"Fulfilling the achievements felt rewarding"* ($t(26){=}3.05$, $p{=}0.005$, $d{=}1.06$), all in favor for the experimental group. Within a structuring qualitative content analysis, we asked participants what kind of *Achievements* they would have liked to see in the game, without providing information that other *Achievements* could have been in their play-through. Most of the proposed *Achievements* were among or similar to the remaining ones. Seven of the participants stated to want *Achievements* that could be attributed to one of their assigned player types and another seven mentioned *Achievements* that did not fit their *BrainHex* estimate.

[1] http://survey.ihobo.com/BrainHex/

Discussion & Future Work

Results of the *IMI* indicate that *Achievements* adapted to the individual player type do not necessarily render game experiences more enjoyable or exciting, but evidently more engaging, increasing the resulting motivation. More precisely, we have identified a significant difference in *effort-importance* in favor of the experimental condition. This is supported by the strong effects that were measured assessing the impact of the adapted *Achievements* on perceived reward and effort spent solely on completing these, even though their introduced importance within the game was secondary. However, regarding *interest-enjoyment*, we couldn't find any significant differences dependent on the adaptability of *Achievements*. Since *interest-enjoyment* is a strong predictor for intrinsic motivation overall [9], our results indicate that only specific motivational aspects (related to *effort-importance*) could be fostered via the adaptation of *Achievements*.

BrainHex proved to be able to supply suitable classifications of player types to which a multitude of *Achievements* were successfully ascribable. Yet, when it comes to the qualitative assessment of desired *Achievements*, participants were equally likely to request *Achievements* that did not fit their assigned player types as they were to fitting ones. This might stem from the two-stage problem: Firstly, the mapping of *BrainHex* to player types still needs empirical validation and secondly, the translation of particular *Achievements* to player types is also imprecise and one-dimensional. To tackle this problem, we plan to extend our research in the future by investigating current *Achievement* types that are established in the video game industry. Based on our findings, we aim to compile an *Achievement* classification system and examine how certain player types match with specific types of *Achievements*. Above that, we plan to extend this study in order to capture the effects of *Achievement* adaptation between particular player types and investigate if the type itself has an influence on the motivation-increasing potential.

Understanding the effects of matching certain *Achievements* with specific player types can bear great potential for the video game industry. Instead of defining uniform *Achievements* for everyone, developers can address players more individually by conducting a simple and brief typology test beforehand. For this purpose, another topic for future research would be to analyze player behavior and extract their *primary* and *secondary* classes automatically, making a test outside of the game obsolete.

Conclusion

Unlockable *Achievements* have proven to be a major motivator in video games. Yet, the public and often bragging-focused nature of these accomplishments facilitates rather extrinsic instead of intrinsic motivation, while the latter is known to be a substantially better instrument for actual engagement and enjoyment in games. In order to harness the motivational potential while offering more intrinsic appeal, we introduce the adaptation of *Achievements* on the basis of individual player types. Within a comparative user study, we were able to provide evidence for a positive impact regarding certain motivational aspects such as perceived effort and importance as well as sense of reward and individualization.

Acknowledgements

We would like to thank all participants of the study. This work was partially funded by the EU project *first.stage* and by the German Research Foundation (DFG) as part of Collaborative Research Center (SFB) 1320 EASE - Everyday Activity Science and Engineering, University of Bremen (http://www.easecrc.org/), subproject H2.

REFERENCES

1. Henrik Aabom. 2014. Exploring the intrinsic nature of video game achievements. (2014).
2. Richard Bartle. 1996. Hearts, clubs, diamonds, spades: Players who suit MUDs. *Journal of MUD research* 1, 1 (1996), 19.
3. Richard Bartle. 2005. Virtual worlds: Why people play. *Massively multiplayer game development* 2, 1 (2005), 3–18.
4. Max V Birk, Dereck Toker, Regan L Mandryk, and Cristina Conati. 2015. Modeling motivation in a social network game using player-centric traits and personality traits. In *International Conference on User Modeling, Adaptation, and Personalization*. Springer, 18–30.
5. Darryl Charles, A Kerr, M McNeill, M McAlister, M Black, J Kcklich, A Moore, and K Stringer. 2005. Player-centred game design: Player modelling and adaptive digital games. In *Proceedings of the digital games research conference*, Vol. 285. 00100.
6. Edward L Deci, Haleh Eghrari, Brian C Patrick, and Dean R Leone. 1994. Facilitating internalization: The self-determination theory perspective. *Journal of personality* 62, 1 (1994), 119–142.
7. Edward L Deci, Richard Koestner, and Richard M Ryan. 1999. A meta-analytic review of experiments examining the effects of extrinsic rewards on intrinsic motivation. *Psychological bulletin* 125, 6 (1999), 627.
8. Dan Dixon. 2011. Player types and gamification. In *Proceedings of the CHI 2011 Workshop on Gamification*. 7–12.
9. Center for Self-Determination Theory. 2019. Intrinsic Motivation Inventory (IMI). (2019). `https://selfdeterminationtheory.org/intrinsic-motivation-inventory/`.
10. Juho Hamari and Veikko Eranti. 2011. Framework for Designing and Evaluating Game Achievements.. In *Digra conference*. Citeseer.
11. Juho Hamari and Janne Tuunanen. 2014. Player types: A meta-synthesis. (2014).
12. Mikael Jakobsson. 2011. The achievement machine: Understanding Xbox 360 achievements in gaming practices. *Game Studies* 11, 1 (2011), 1–22.
13. Kirsi Pauliina Kallio, Frans Mäyrä, and Kirsikka Kaipainen. 2011. At least nine ways to play: Approaching gamer mentalities. *Games and Culture* 6, 4 (2011), 327–353.
14. Rensis Likert. 1932. A Technique for the Measurement of Attitudes. *Archives of psychology* (1932).
15. Edward McAuley, Terry Duncan, and Vance V Tammen. 1989. Psychometric properties of the Intrinsic Motivation Inventory in a competitive sport setting: A confirmatory factor analysis. *Research quarterly for exercise and sport* 60, 1 (1989), 48–58.
16. Baptiste Monterrat, Michel Desmarais, Elise Lavoué, and Sébastien George. 2015. A player model for adaptive gamification in learning environments. In *International conference on artificial intelligence in education*. Springer, 297–306.
17. Lennart E Nacke, Chris Bateman, and Regan L Mandryk. 2014. BrainHex: A neurobiological gamer typology survey. *Entertainment computing* 5, 1 (2014), 55–62.

18. Rita Orji, Regan L Mandryk, and Julita Vassileva. 2017. Improving the efficacy of games for change using personalization models. *ACM Transactions on Computer-Human Interaction (TOCHI)* 24, 5 (2017), 32.

19. Rita Orji, Regan L Mandryk, Julita Vassileva, and Kathrin M Gerling. 2013. Tailoring persuasive health games to gamer type. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2467–2476.

20. Daniel H Pink. 2011. *Drive: The surprising truth about what motivates us*. Penguin.

21. Robert W Plant and Richard M Ryan. 1985. Intrinsic motivation and the effects of self-consciousness, self-awareness, and ego-involvement: An investigation of internally controlling styles. *Journal of personality* 53, 3 (1985), 435–449.

22. Andrew K Przybylski, C Scott Rigby, and Richard M Ryan. 2010. A motivational model of video game engagement. *Review of general psychology* 14, 2 (2010), 154–166.

23. Richard M Ryan. 1982. Control and information in the intrapersonal sphere: An extension of cognitive evaluation theory. *Journal of personality and social psychology* 43, 3 (1982), 450.

24. Richard M. Ryan, James P. Connell, and Robert W. Plant. 1990. Emotions in nondirected text learning. *Learning and Individual Differences* 2, 1 (1990), 1 – 17. DOI:`http://dx.doi.org/https://doi.org/10.1016/1041-6080(90)90014-8`

25. Richard M Ryan and Edward L Deci. 2000. Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary educational psychology* 25, 1 (2000), 54–67.

26. Richard M Ryan, Richard Koestner, and Edward L Deci. 1991. Ego-involved persistence: When free-choice behavior is not intrinsically motivated. *Motivation and emotion* 15, 3 (1991), 185–205.

27. Richard M Ryan, Valerie Mims, and Richard Koestner. 1983. Relation of reward contingency and interpersonal context to intrinsic motivation: A review and test using cognitive evaluation theory. *Journal of personality and Social Psychology* 45, 4 (1983), 736.

28. Gustavo F Tondello, Rina R Wehbe, Rita Orji, Giovanni Ribeiro, and Lennart E Nacke. 2017. A framework and taxonomy of videogame playing preferences. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*. ACM, 329–340.

29. Nick Yee. 2006. Motivations for play in online games. *CyberPsychology & behavior* 9, 6 (2006), 772–775.

Give MEANinGS to Robots with Kitchen Clash: A VR Human Computation Serious Game for World Knowledge Accumulation

Johannes Pfau[1], Robert Porzel[1], Mihai Pomarlan[1], Vanja Sophie Cangalovic[1], Supara Grudpan[1], Sebastian Höffner[1], John Bateman[1], and Rainer Malaka[1]

[1]University of Bremen, Bibliothekstraße 1, 28359 Bremen, Germany
{jpfau,porzel,pomarlan,vanja,gud,shoeffner,bateman,malaka}@uni-bremen.de

Abstract. In this paper, we introduce the framework of MEANinGS for the semi-autonomous accumulation of world knowledge for robots. Where manual aggregation is inefficient and prone to incompleteness and autonomous approaches suffer from underspecified information, we deploy the human computation game *Kitchen Clash* and give evidence of its efficiency, completeness and motivation potential.

Keywords: Serious Game · Knowledge accumulation · Framework

1 Introduction

Robotic proficiency excels in well-defined tasks and environments [12, 1, 4], but fails in compensating for missing or too generic information. Human-level world knowledge has been shown to close the reasoning gap [4], yet teaching robots this kind of knowledge remains one of the most challenging tasks for robotic AI research, since autonomous approaches end up with underspecified information and manual accumulation results in incalculable effort. In this paper, we introduce *Kitchen Clash*, a VR human computation serious game for the extraction of human world knowledge in the context of everyday activities. Within the framework of MEANinGS (Malleating Everyday Activity Narratives in Games and Simulations), we integrate a combination of information-transforming modules that include finding a proper set of instructions for a given complex task, processing these syntactically as well as semantically to detect underspecified information, autonomously generating testbed scenarios including a variety of decision making affordances and finally solving world knowledge problems by human computation through a serious game aided by physical simulation. In an explorative pilot study, we assessed user experience, appraisal and the overall viability of the presented serious game to report on the findings and demonstrate the feasibility of the approach. To constitute a baseline condition, we evaluated these findings against a control group executing manual knowledge accumulation, resulting in higher efficiency, increased motivation and considerably higher information retrieval. This paper contributes to the community of serious game research, presenting a successful application advantageous to a real-world problem solving field, as well as to the community of robotic research, exemplifying the practicability of a novel framework to overcome underspecified knowledge.

2 Related Work

One of the earliest research programs to study autonomous robots was the Shakey project [12]. Shakey was a mobile robot that used planning to reason about its actions and performed tasks that required planning of paths and actions as well as re-arranging of simple objects. This work was seminal for the fields of classical planning and computer vision. Nevertheless, even in Shakey's simple environment, the limitations of the approach became clear, as the computational complexity of planning problems proved, in general, to be intractable.

Many researchers [4, 12, 11, 13, 18] have worked on providing robotic systems with human-like common sense knowledge so that the robots could, hopefully, avoid costly planning from scratch or trial and error. Dang et al.[3] proposed a method to teach a robot to manipulate everyday objects through human demonstration. The authors asked participants to put on motion capture suits and perform tasks, such as opening a microwave or a slide door, and recorded 3D marker trajectories. These trajectories were used as the input for a chain learning algorithm. Parde et al.[13] developed a method to train robots to learn the world around them by using interactive dialogue and virtual games. The game asked a human player to put some objects in front of the robot and challenge it to guess which object the user has in mind. Through many gameplay sessions, the robot learns about objects and features which describe them and associates these with newly captured training images. Beetz et al.[2, 1] proposed the software toolbox for design, implementation, and the deployment of cognition-enabled autonomous robots to perform everyday manipulation activities. To teach the robot, they use a marker-less motion capture system to record human activity data, which is then stored as experience data for improving manipulation program parameters. Programs, object data, and experience logs are uploaded to the openEASE web server, from which they can be retrieved as needed to extend the task repertoires and objects that a robot can recognize.

The representations needed for action knowledge have also been a topic of research, because the symbolic, highly abstract, "actions as black boxes" representations of the Shakey era do not result in robust behaviors in a realistic environment. In general, action knowledge tends to be subsymbolic, and often takes the form of success/failure probability distributions over an action's parameter space [17, 19]. Note that the experience the robot learns from doesn't have to be from the real world. Simulated episodes, produced either by a human player of a game or a robot simulating itself, can be used for this purpose. Simulation will of course not provide a complete description of a realistic action, but even very coarse simulation can already be useful for a robot that needs to validate its plans and/or pick a better set of parameters [8].

In our work, we propose MEANinGS to use a VR human computation serious game to simulate real-world tasks in realistic environments and situations. Recorded trajectories can be translated to real-world robotic movements which are spatially less constrained than motion capturing approaches and accumulated world knowledge can help overcoming underspecified information, which has the potential to reduce planning computation considerably. Similar to the aforementioned approach, we contribute to the field of cumulative robotic knowledge by adding resulting symbolical and subsymbolical insights to the openEASE repertoire.

3 Implementation

3.1 Framework

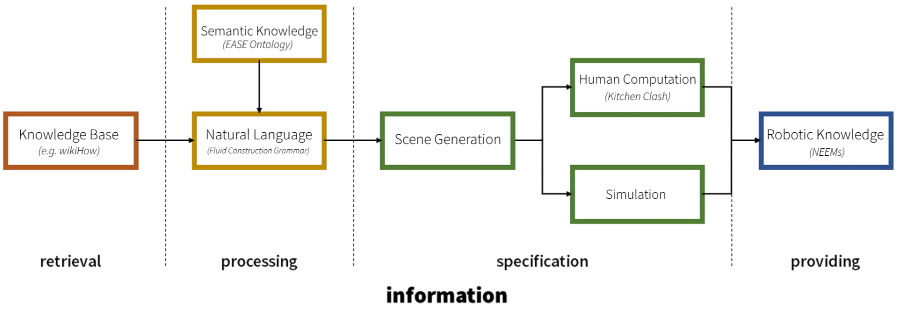


Fig. 1. Flowchart of the introduced MEANinGS framework.

Figure 1 demonstrates the flow of information, as well as the impact of and interaction between the particular modules. MEANinGS originated in the context of everyday household activities and focuses on knowledge accumulation in this area, while its functionality is not limited to the application field. Offering an interface to any natural language based instruction set (**retrieval**), the contained information is **processed** in order to represent subtasks as tuples of manipulative actions and objects acted on or with. When utilizing natural language, the ontological scope of these objects is often heavily underspecified, since humans are used to working with generalized information and specifying these in terms of individual choice, influenced by world knowledge, availability and preference. Yet, this underspecification does not render all possible objects contained in a general term as viable (or usual). Thus, in the **specification** layer, human knowledge is added through *Kitchen Clash*, a serious game presenting a decision making paradigm within this set of objects. Parallel to this, complementing world knowledge is derived from physical properties of the objects and their surroundings via simulation. In both approaches, object choices can be quantified and thus ranked by efficiency and effectiveness. Within the simulation, this assessment can be realized in a fully autonomous manner, while the serious game offers further qualitative insights, since peer-rated quality measurements are included in the rating process, as well as preference and conventionality measures. Eventually, world knowledge is aggregated with trajectorial and contextual information and **provided** as *narrative-enabled episodic memories* (*NEEMs*) according to the KnowRob [1] paradigm.

3.2 Knowledge Base

In order to have a generic, comprehensive framework that is capable of adapting to human data input, our approach does not rely on a single knowledge base, but is designed to handle any set of natural language instructions that are goal-oriented and describe the most crucial subtasks sequentially or hierarchically. In this way, we introduce an interface that manages to grasp verbal commands equally effective as cooking recipes or tutorial websites (e.g. wikiHow). After retrieving the document encompassing the entire task completion, subgoals are derived from the contained sentences or steps and processed in the next module, independently of each other.

3.3 Natural Language Processing

In order to flexibly handle natural language input consisting of abstract and underspecified instructions, a deep semantic parser based on the Fluid Construction Grammar formalism [16] is used. Both the lexicon and the analysis itself make use of ontological knowledge described in Section 3.4, to guide the extensive search process, disambiguate otherwise unclear instructions and evoke unspecified parameters which need to be inferred by later processing steps of MEANinGS. In this way, natural language commands are transformed into a series of desired actions, accompanied by their parameters and respective pre- and post-conditions.

3.4 Ontology

The semantics of the actions and entities involved in the game are defined by a formal ontology. This ontology is designed to provide descriptions for everyday activities in terms of human physiology and human mental concepts, as well as enabling formal reasoning. The ontology supplying the labels for the objects has been designed using the principles proposed by Masolo et al. and is created by using the DOLCE+DnS Ultralite ontology (DUL) as an overarching foundational framework [6, 5]. Specific branches of the KnowRob knowledge model pertaining to everyday activities [1], such as those involved in table setting and cooking, have consequently been aligned to the DUL framework. Additional axiomatization that is beyond the scope of description logics is integrated by means of the Distributed Ontology Language [9]. For the task at hand, however, only the taxonomic model is employed to classify events and objects.

3.5 Scene Generation

Within the scene generation module, we aim to provide a rich contextual world for the following **specification** methods by preparing a scene that contains sufficient interactable objects to ensure *completeness* (i.e. solvability of each contained subgoal) and to facilitate *variety of choices* (in order to retrieve actual world knowledge through humans' decisive solutions or physical properties of the simulation). Since the **processing** layer results in rather generic, underspecified semantic descriptions of objects required to fulfill the task, this module tries to generate as many alternatives for the respective objects as possible. This can be realized either in a bottom-up (empty scene where only necessary objects and alternatives are generated) or top-down (fully fledged household scene where only objects missing for completion and/or their alternatives are generated) approach. Once a scene meets the conditions of the task, it can be used for both human computation as well as simulation.

Placement of objects in a scene is done in a generate and validate fashion. The qualitative constraints on object placements are first used to select and/or modify probability distributions for object positions. These probability distributions can be learned from a set of training scenes– e.g., what it means for a chair to be "near" a table can be represented as a distribution on relative locations of the chair to the table–, or sometimes inferred from an object's shape; for example, the top of an object corresponds to the fragments of its surface with the highest z-coordinate. Probability distributions resulting from different constraints on the same object are combined via point-wise multiplication. Once constructed, a probability distribution accounting for all qualitative constraints on an object is sampled several times to produce candidate poses, and the first candidate that passes a list of tests– e.g. placing the object there would not result in collisions– is used.

3.6 Human Computation

As the primary gap filler for underspecified information, we introduce *Kitchen Clash*, a virtual reality-based, competitive household serious game. Players are challenged with the same set of instructions that stem from the original knowledge base within a virtual household produced by the scene generation module. Each instruction is realized as reaching a subgoal represented by the contained objects and the type of the manipulation (picking up/dropping objects, combining objects with other objects, making use of specific object properties, etc).

VR, compared to offline or non-natural interaction approaches, offers the great potential of tracking complete trajectories of hand, head and body movement, as well as the distinctly classified manipulation actions. Players are asked to execute these tasks with optimal efficiency and quality, which is measured by *time spent on a task*, the *number of recognizable actions* and the *number of undesired events* (e.g. breaking dishes or glitching through physical barriers). Additionally, these sessions are assessed qualitatively by peer-rating individual executions from other players, in an either absolute or relative measurement. Eventually, players are rewarded with a score representing their qualitative and quantitative success.

3.7 Simulation

Within MEANinGS, the simulation branch is employed to estimate concrete parameter setting for the ultimate robotic execution of the activities involved. For example, in the case of transporting liquids in various containers from a source to a target location, the game engine physics can be used to simulate different velocities and trajectories and measure the ensuing spill rate in order to find a suitable setting. Ultimately, we see this as a modern extension of the KARMA system [10], in which the complete understanding of an utterance entails a mental simulation thereof. It is also related to "projection" [8], which is light-weight simulation used by a cognitive robot to try combinations of program parameters and/or change sequences of actions quickly, in a simulated world, before attempting them in reality.

4 Exemplary case

To showcase the functional principle of the framework, we present one of the example tasks used in the **Evaluation**, i.e. *to prepare a portion of cucumber salad.*

Retrieval. When querying wikiHow as a possible source for natural language instructions, *cucumber salad* will result in a multitude of cucumber salad variants, from which the most basic one will be chosen since no further specifications are asked for. Within this module, the overall task will be divided into subtasks (*Slice the cucumber into thin pieces*, *Place the slices into a bowl* and *Pour dressing over the cucumbers*), which will be forwarded to the **processing** layer.

Processing. The natural language parser extracts one action per subtask, each of which should be performed by the discourse addressee - in this case the human player. For the *slicing* action, the undergoer *cucumber* is identified while the obligatory instrument slot is left unspecified. Moreover, the action should result in a goal state that is defined by the changed consistency of the undergoing object. Also, the ontologically equivalent *cutting* action is extracted, to prepare for the case in which only one of these actions is known by the following processing steps. The subsequent *placing* action describes the desired trajectory of the undergoing *slices* to their destination, an undetermined container of type *bowl*. For the final *pouring* action, the poured substance *dressing* and its destination, *the cucumbers*, are identified. Furthermore, the various referring expressions of the main ingredient all resolve to the initial *cucumber* object, in its different configurations.

Specification. In order to prepare a suitable testbed, the scene generation module spawns a *cucumber* (since it doesn't find more specific alternatives to the term) and different variants of *cutting objects* (scissors, a kitchen knife, a butter knife, a butcher's knife, etc).
Within *Kitchen Clash*, a new level is generated that constitutes the challenge and constraints of the overall task. Players entering this level have to find suitable solutions for the presented subtasks and execute these quickly and dexterously, since time, number of actions and the opinion of other players determine the final score. If e.g. a player executes a pickup action on the kitchen knife, triggers a collision between the knife and a cucumber (c.f. Figure 2), collects the resulting slices, causes them to fall into a bowl and initiates the final collision between dressing and cucumbers, all subgoal constraints have been fulfilled and the main task is completed.

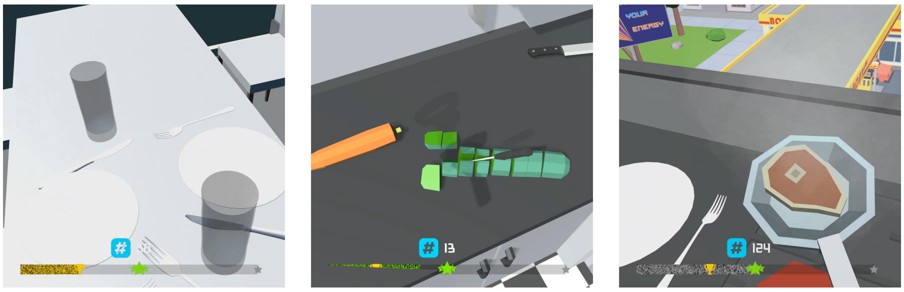


Fig. 2. In-game representation of the three tasks. UI has been kept minimal to prevent distraction, action number is counted and required time outlined on a bar with respect to the best and average time targets. In the second screenshot segment, the *cucumber slicing* task is represented, where the required *cutting object* is specified by taking a *serrated utility knife*.

When it comes to simulating the physical properties, the same scene is populated by a robotic agent instead of a human performer, that evaluates the cutting action between all given alternatives and comes up with a quantitative result of the most appropriate parameters and choices.

Providing. In the end, trajectories and action choices from the **specification** layer are formulated into the standardized *NEEM* description to generalize and publish the insights to the open robotic community.

5 Evaluation

In order to assess the feasibility of the approach, the overall player experience and appraisal, as well as to generate a first data set for further analysis, we conducted an exploratory comparative user study in a laboratory setting. Data was gathered through game protocols, screen capture and a post-study questionnaire. The study was split into two groups in a between-subjects design, where the **VR** group was exposed to *Kitchen Clash* within the associated framework and the **control** group had to accumulate the desired world knowledge manually by depicting the respective tasks in written form.

Measures. In-game, we tracked movements from head and hands every second, as well as all of the players' actions, collision events, time measures and attained scores (quantitative and qualitative). The control group submitted instructional data textually. Through the questionnaire, demographics and prior experience in VR were recorded. Using seven-point Likert scales, we asked for players' motivation (using the Intrinsic Motivation Inventory (IMI) [7]), presence (using the igroup Presence Questionnaire (IPQ) [15]), comprehensibility and perceived usefulness of the game. Additionally, participants elaborated on their decision making processes with respect to world knowledge accumulation.

Procedure. Following informed consent and a temporally unlimited tutorial that explained the controls and interactions of the game, participants were asked to complete three levels containing complex tasks. In the first level, they had to set a table for two persons, deciding on the type of cutlery and tableware and arranging these in their usual composition. Level two consisted of the formerly explained task of turning cucumbers into a salad. Finally, they were asked to prepare a steak by heating the hotplate, choosing a pan, filling it with oil and cooking the steak until the desired degree of doneness was reached. The tasks did not differ between the VR and control group. They were specifically designed to extract world knowledge about solving underspecified information, providing preferred or conventional items, object target constellations and actual execution trajectories. After completing all levels, the subject was redirected to the final questionnaire.

Participants. ($n = 26$) participants took part in the study. (46% male, 54% female, aged 22-58 ($M = 29.9, SD = 8.3$). 72.7% stated having prior experience in VR.

Results. On average, subjects of the VR group spent ($M_1 = 150.9, SD_1 = 51.7; M_2 = 94.9, SD_2 = 42.3; M_3 = 114.9, SD_3 = 34.2$) seconds on the three respective tasks, whereas the control group required ($M_1 = 244, SD_1 = 108.1; M_2 = 350, SD_2 = 196.3; M_3 = 336.3, SD_3 = 181.4$) seconds. Using a Welch's t test, we found significant or even highly significant effects for required time between the groups in all tasks ($p_1 < 0.05, d_1 = 1.1$), ($p_2 < 0.01, d_2 = 1.8$), ($p_3 < 0.01, d_3 = 1.7$, cf. Figure 3).

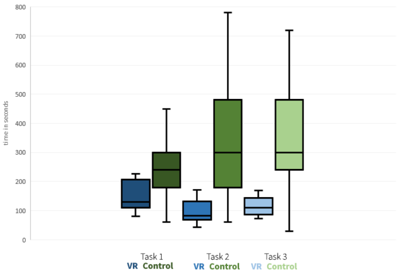


Fig. 3. Time required to fulfill the three tasks between VR (blue) and control (green).

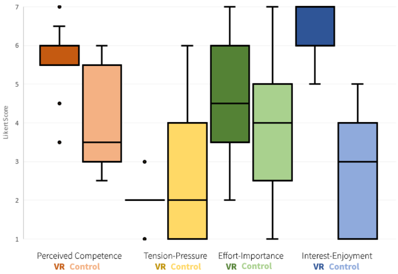


Fig. 4. Results of IMI categories Perceived Competence (red), Tension/Pressure (yellow), Effort-Importance (green) and Interest/Enjoyment (blue) between VR (left) and control (right).

Assessing the IMI, we found no difference for Effort-Importance or Tension-Pressure, but highly significant effects for Perceived Competence ($p < 0.01, d = 1.26$) and Interest-Enjoyment ($p < 0.01, d = 3.13$), showing VR drastically outperforming the control group in terms of motivation (cf. Figure 4). When asked how descriptive the execution in VR (or in written instructions) can be with respect to the real set of actions, 81.2% of the VR group stated that the execution comes close to the real actions, where from the control group only 40% were convinced that real tasks can be sufficiently expressed in written form. Participants had no trouble following the given instructions (indicated by ($M = 6.27, SD = 0.62$) on a comprehensibility scale). According to the IPQ, VR participants reported a mediocre presence ($M = 4.15, SD = 0.81$) for Spatial Presence, ($M = 4.3, SD = 0.42$) for Involvement, ($M = 3.3, SD = 0.54$) for Realness and ($M = 5.63, SD = 1.15$) for General Presence). Regarding simulation sickness, most participants reported no discomfort at all ($M = 2.1, SD = 1.73$). Most of the subjects stated that they would like to play similar games more often ($M = 5.72, SD = 1.6$). Elaborating on the decision making strategy, 45.4% of the participants stated to select the necessary objects based on the respective task or prior experiences, where 54.6% tended to just take the first available thing.

For the qualitative measurements, subjects reported that VR is *"capable of capturing the most crucial aspects of the tasks"* and *"close to reality"*, despite *"lacking haptic feedback [that] decreases grasping accuracy"* and *"not [being] able to perform fine motor functions"*. Participants of the control group stated that it is *"impossible to find the right level of detail"*, *"implicit knowledge is easily overlooked"*, *"it takes way too long to describe all actions in detail"* and *"you cannot really describe cooking since you don't think at details that will come up in the process"*.

We also assessed the amount of information retrievable from the sessions in both groups. Within VR, all executions managed to complete the tasks and filled all occurrences of underspecified information, since these were needed to finish the respective level. Yet, many unnecessary actions were tracked that trace back to the novel experience of the game, accustoming to VR and the controls and the very broad tracking scope. The amount of unnecessary information was significantly smaller in the control group, but in most of the cases they failed to solve the underspecification problem, even when going into detail. Above that, the textual descriptions deviated considerably in their semantics, due to different perceptions of the task, the projection to their individual environment or personal preferences.

6 Discussion and Future Work

Contrasting accumulation of world knowledge manually and in a gamified approach, we have given evidence that human computation can result in significantly higher efficiency, motivation and closeness to the actual execution. Above that, *Kitchen Clash* was able to track complete sequences of actions that describe the fulfillment of tasks both symbolically (registering required operations) as well as subsymbolically (tracking continuous trajectories and contact parameters). Participants enjoyed playing and competing with other players and were interested in continuing the game. Based on these results, we have demonstrated the opportunities and usefulness of human computation for world knowledge aggregation and the feasibility of the overall framework. Yet, this study illustrated that the current implementation suffers from over-collecting unnecessary information and undesirable player choices (e.g. players who take the first object available instead of making an informed decision). Regarding the first issue, we aim to compile large sets of similar task executions using Deep Player Behavior Models [14], offering an optimization paradigm across sessions to extract the necessary core actions needed to fulfill the task probabilistically. When it comes to undesirable player choices, we will evaluate a knockout system of object alternatives that constrains the *variety of choices* of the Scene Generation module in order to force players to overcome obstinate individual preferences and obvious decisions. Furthermore, we are aiming for a narrower interaction between the human computation and the simulation module to generate more elaborate level constellations in *Kitchen Clash* and to make use of the accumulated sequential action knowledge while simulating. Eventually, we are going to open up the game to online multiplayer scenarios where players have to compete against other human players as well as agents representing the aggregated knowledge while learning continually.

7 Conclusion

Learning from natural language instructions is a desirable opportunity for robots, but ends up in underspecified information, even when accessing detailed directions. Introducing MEANinGS, we present a potent framework able to break down these instructions syntactically and semantically, before resolving missing or underspecified information with the aid of human computation. With this approach, we have shown to outperform manual accumulation in terms of efficiency, motivation and completeness. This work demonstrates a successful application of a human computation serious game to facilitate research in the context of robotic learning.

8 Acknowledgments

This work was funded by the German Research Foundation (DFG) as part of Collaborative Research Center (SFB) 1320 EASE - Everyday Activity Science and Engineering, University of Bremen (http://www.easecrc.org/), subprojects H2 and P1. We thank all participants.

References

1. Beetz, M., Bessler, D., Haidu, A., Pomarlan, M., Bozcuoglu, A.K., Bartels, G.: Know Rob 2.0 - A 2nd Generation Knowledge Processing Framework for Cognition-Enabled Robotic Agents. Proceedings - IEEE International Conference on Robotics and Automation pp. 512–519 (2018). https://doi.org/10.1109/ICRA.2018.8460964
2. Beetz, M., Mösenlechner, L., Tenorth, M.: Cram—a cognitive robot abstract machine for everyday manipulation in human environments. In: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 1012–1017. IEEE (2010)
3. Dang, H., Allen, P.K.: Robot learning of everyday object manipulations via human demonstration. In: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 1284–1289. IEEE (2010)
4. Kunze, L., Tenorth, M., Beetz, M.: Putting people's common sense into knowledge bases of household robots. In: Annual Conference on Artificial Intelligence. pp. 151–159. Springer (2010)
5. Mascardi, V., Cordì, V., Rosso, P.: A comparison of upper ontologies (technical report disi-tr-06-21). Dipartimento di Informatica e Scienze dell'Informazione (DISI), Universitŕ degli Studi di Genova, Via Dodecaneso **35**, 16146 (2008)
6. Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A.: Wonderweb deliverable d18, ontology library (final). ICT project **33052**, 31 (2003)

7. McAuley, E., Duncan, T., Tammen, V.V.: Psychometric properties of the Intrinsic Motivation Inventory in a competitive sport setting: a confirmatory factor analysis. Research quarterly for exercise and sport **60**(1), 48–58 (1989)
8. Mösenlechner, L., Beetz, M.: Fast temporal projection using accurate physics-based geometric reasoning. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). pp. 1821–1827. Karlsruhe, Germany (May 6–10 2013)
9. Mossakowski, T.: The distributed ontology, model and specification language–dol. In: International Workshop on Algebraic Development Techniques. pp. 5–10. Springer (2016)
10. Narayanan, S.S.: Karma: Knowledge-based active representations for metaphor and aspect. (1999)
11. Nielsen, R.D., Voyles, R., Bolanos, D., Mahoor, M.H., Pace, W.D., Siek, K.A., Ward, W.H.: A platform for human-robot dialog systems research. In: 2010 AAAI Fall Symposium Series (2010)
12. Nilsson, N.J.: Shakey the robot. Tech. rep., SRI INTERNATIONAL MENLO PARK CA (1984)
13. Parde, N.P., Papakostas, M., Tsiakas, K., Dagioglou, M., Karkaletsis, V., Nielsen, R.D.: I spy: An interactive game-based approach to multimodal robot learning. In: Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence (2015)
14. Pfau, J., Smeddinck, J.D., Malaka, R.: Towards deep player behavior models in mmorpgs. In: Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play. pp. 381–392. CHI PLAY '18, ACM, New York, NY, USA (2018). https://doi.org/10.1145/3242671.3242706
15. Schubert, T., Friedmann, F., Regenbrecht, H.: Embodied presence in virtual environments (1999)
16. Steels, L.: Basics of fluid construction grammar. Constructions and Frames **9**(2), 178–225 (2017). https://doi.org/https://doi.org/10.1075/cf.00002.ste
17. Stulp, F., Fedrizzi, A., Mösenlechner, L., Beetz, M.: Learning and Reasoning with Action-Related Places for Robust Mobile Manipulation. Journal of Artificial Intelligence Research (JAIR) **43**, 1–42 (2012)
18. Walther-Franks, B., Smeddinck, J., Szmidt, P., Haidu, A., Beetz, M., Malaka, R.: Robots, pancakes, and computer games: designing serious games for robot imitation learning. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. pp. 3623–3632. ACM (2015)
19. Winkler, J., Bozcuoğlu, A.K., Pomarlan, M., Beetz, M.: Task parametrization through multi-modal analysis of robot experiences. In: Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems. pp. 1754–1756. AAMAS '17, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2017), http://dl.acm.org/citation.cfm?id=3091125.3091428

We Asked 100 People: How Would You Train Our Robot?

Johannes Pfau
jpfau@tzi.de
University of Bremen
Germany

Rainer Malaka
University of Bremen
Germany
malaka@tzi.de

ABSTRACT

While robotic proficiency excels in constrained environments, the demand for vast amounts of world knowledge to cover unforeseen circumstances, constellations and tasks prevents sufficiently robust real-world application. Human computation has shown to provide successful advances to close this reasoning gap and accumulate knowledge, yet being greatly reliant on the quality of the provided data. In this paper, we introduce the game with a purpose *Tool Feud* that collects popularity rankings of object choices for robotic everyday activity tasks and evaluate an approach for classifying malicious responses automatically.

CCS CONCEPTS

• **Software and its engineering** → **Interactive games**; • **Computer systems organization** → *Robotics*.

KEYWORDS

Human computation; games with a purpose; everyday activities

ACM Reference Format:
Johannes Pfau and Rainer Malaka. 2020. We Asked 100 People: How Would You Train Our Robot?. In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '20 EA), November 2–4, 2020, Virtual Event, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3383668.3419864

1 INTRODUCTION

As long as restricted to well-defined tasks and environments, robotic performances can exceed even human proficiency [1, 13], yet missing or too generic *(underspecified)* information can impede real-world dissemination categorically. To overcome these shortcomings in reasoning, human-level world knowledge can be implemented [8], yet the teaching process of this knowledge remains one of the most challenging tasks for robotic AI research, since autonomous approaches end up with underspecified information and manual accumulation results in incalculable effort [17]. In this work, we introduce and evaluate *Tool Feud*, a mobile *game with a purpose* (GWAP) for the accumulation of human world knowledge in the domain of everyday activities. Using an explorative pilot study, we assessed the usability of the utilized prototype, its potential to aggregate preference distributions for everyday activity tasks and examine quantifiable differences between viable and malicious responses based on the behavior of the original player, effectively approaching the following research questions:

- How can games with a purpose be utilized to produce object preference distributions for everyday activity tasks?
- Can disruptive or malicious responses within this data be automatically classified by contrasting these against the *wisdom of the crowd* without missing creative answers?

Eventually, *Tool Feud* turned out to be a suitable vehicle to accumulate mundane world knowledge on a larger scale, while malicious responses could be identified calculably. This paper contributes to the community of games user research and serious game research, presenting a successful application advantageous to a real-world problem solving field, as well as to the domain of autonomous and cognitive robotics, exemplifying promising methods to overcome underspecified knowledge.

2 RELATED WORK

In order to cope with continuously changing environments, tasks and requirements, the fields of autonomous and cognitive robotics have examined various paradigms of decision making, not limited to action and path planning [13], common sense frameworks [8, 12] or iterative parameter evaluation by simulation [11]. However, even when performing well in test-bed environments, real-world applications require a vast amount of world knowledge and dynamic decision making [2], which results in an interminable effort of data collection. One of the most successful ways to deal with uncertain information on a generalizable scale is the inclusion of human knowledge, either by imitation [1], interaction [14], demonstration [3] or human computation [17]. While the latter bears great potential in aggregating large amounts of diverse world knowledge from participants of many cultural backgrounds, it also suffers from distortions in data quality produced by disruptive or questionable responses [16]. These can stem from users trying to break the system [23, 25] or being not motivated enough to produce viable results [24]. To exclude this malicious data from being processed, some approaches introduce underlying trust or quality functions that again had to be populated manually [7, 17], which can be effective but raise additional effort and workload.

This work extends previous research by introducing a mobile human computation GWAP into the domain of everyday activity robotics, harnessing the *wisdom of the crowd* effect which already showed similar successes in different fields (such as volunteered geographic information [10], automotive design appraisal [4] or galaxy classification [6]) to separate viable responses from those that would harm the quality of the aggregated data.

CHI PLAY '20 EA, November 2–4, 2020, Virtual Event, Canada
© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-7587-0/20/11...$15.00
https://doi.org/10.1145/3383668.3419864

3 APPROACH

To aggregate a large body of responses for everyday activity tasks, we designed and implemented the mobile GWAP *ToolFeud* (cf. Fig. 1). Analogous to the well-established core game mechanic of the popular TV show *Family Feud*[5], players are asked to guess answers that other players have given to common everyday activities in rapid succession (e.g. *"We asked 100 people: Name something to ... open a parcel!"*). In order to submit an answer, labeled images of conventional tools and household objects are presented in moving patterns that have to be selected. Once a solution is chosen, players are rewarded by increasing the session's score and obtaining a time bonus – both proportional to the popularity of the particular answer. In parallel, the remaining time for a single session is decreasing continuously, so players have to ponder between waiting for their preferred solution or picking an opportune alternative. To raise the difficulty progression even further, the most popular choices disappear as time goes by, which simultaneously augments creative answering by restricting players to make spontaneous decisions for inferior but still viable solutions. The better a player can keep up providing viable answers, the higher scores will be accumulated and the longer a session will be, ending only when running out of time. Effectively, this method of dynamic population similarity calculation might not only end up in engaging gameplay, but would also result in a continuously growing dataset of distributions of object preferences for everyday activities that can be employed to augment robotic decision making by the means of gamified human computation.

4 EVALUATION

To conduct an explorative evaluation of the approach, participants were recruited using e-mail distribution lists and public campus announcements. For the creation of a baseline body of solutions, we utilized a pilot version of the game without dynamic calculations for time and score. After the course of one week, (n=92) unique responses could be collected that served for following analyses.

4.1 Measures

For each everyday activity task and every participant, their subjective top three answers were recorded, together with the set of remaining available alternatives, as well as optional custom responses. After the game session, we evaluated the experience and motivation according to the *Player Experience of Need Satisfaction* (PENS) [22] questionnaire through the subscales *Competence*, *Presence* and *Autonomy*. PENS items were presented via 5-point Likert scales, as well as following constructed questions about the understandability of the tasks and the game and the appropriateness and sufficiency of the proposed solutions.

4.2 Procedure

Following informed consent, participants were forwarded to a website that embedded the evaluated game as a Unity WebGL build, accessible from smartphone as well as PC browsers. After submitting three different answers for each of the respective everyday activity questions, they completed the additional questionnaire and were able to state additional remarks. Subsequent to the data collection, individual responses were manually labeled as *ordinary* (when proposed solutions were consistent with the most popular answers), *creative* (when solutions were infrequent but valid) or *disruptive* (when solutions were infrequent and not suitable to solve the task).

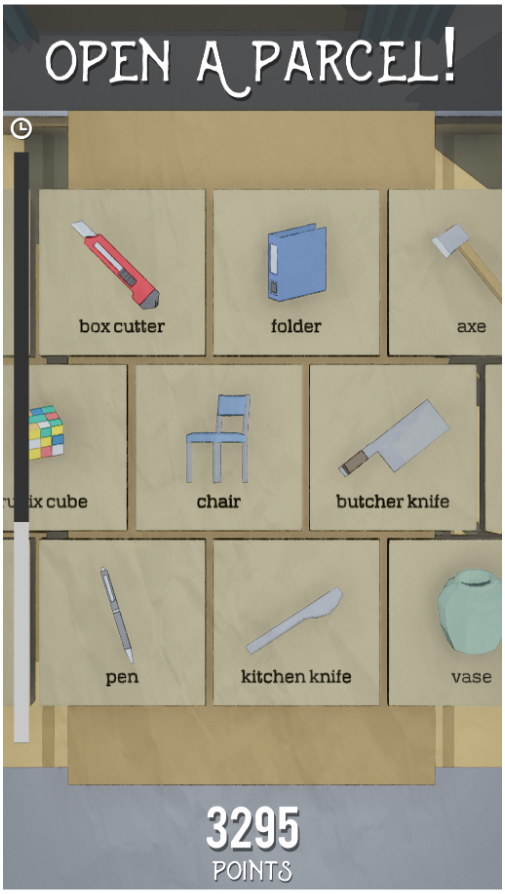


Figure 1: Screenshot of the final version of the mobile game *Tool Feud.* The current task, remaining time, accumulated points as well as moving patterns of object solutions are displayed. Once the players select a solution, they are rewarded with points and time bonus proportional to the popularity of the answer and the next task is presented.

4.3 Participants

In total, (n=92) participants finished the evaluation. Four of these explicitly mentioned that they did not play the game completely on their own, but consulted with a third party on their respective answers. 82 subjects used a laptop or desktop PC for participation, while the remaining 10 drew on their smartphone.

5 RESULTS

Results of the PENS subscales indicate high absolute values of perceived in-game competence (M=4.3, SD=0.97), presence (M=3.97, SD=0.96) and autonomy (M=3.97, SD=1.22). The mechanics of the game as well as the phrasing of the tasks were rated with a high understandability (M=4.62, SD=0.84). 91.3% of the participants stated that viable solutions for a task were provided always or most of the time, which dropped to 83.7% when their preferred objects were eliminated. Only 18.5% submitted custom responses instead of drawing on the presented options.

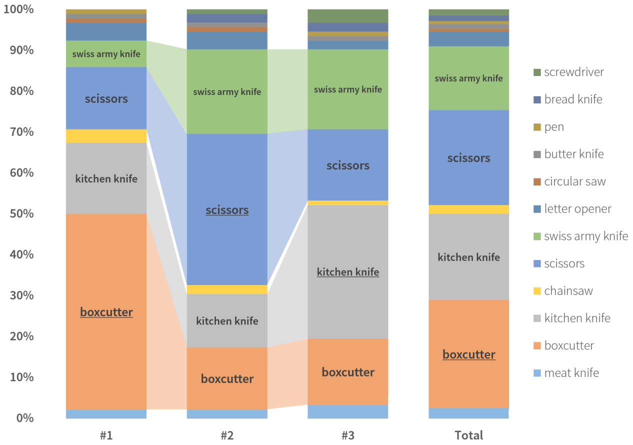


Figure 2: Response objects for the task *"Open a parcel with..."*. #1 displays the initially preferred solution distribution, while #2 and #3 eliminated the previous answers, and *Total* indicates the total overall selection ratio. The most frequent answers of each iteration are underlined.

For each of the tasks, at least five viable solutions could be identified within the most frequent answers (e.g. Fig. 2). In 87.5% of them, the first individual answer (from the unrestricted solution set) was also equal to the most frequent solution in total – otherwise, it appeared in the five most frequent answers (e.g. Fig. 3).

When comparing the similarity of players' answers to the total solution set, we found significant differences between *ordinary*, *creative* and *disruptive* players according to a one-way ANOVA ($F(2, 89)$=42.98, p < .01, partial η^2=0.49, cf. Fig. 4). Subsequently, this was further examined using two-tailed unpaired heteroscedastic t-tests that highlighted significant differences between *ordinary* and *creative* (p < 0.01, Cohen's d=0.96) as well as between *creative* and *disruptive* players (p < 0.01, d=1.35).

6 DISCUSSION

The demonstrated pilot evaluation produced a feasible solution set that can serve as an initial ground truth for the public release of the game and gives insights about object or tool preference distributions of a medium-sized population (n=92). High values of understandability and in-game PENS scores indicate the viable usability of the utilized game prototype, the potential for further related studies and the applicability for accumulating robotic knowledge. Within the game, eliminating obvious or popular answers can reveal a broader scope of viable solutions, compared to one-time or unrestricted tasks, facilitating creative answering. Players that actually made creative use of the whole solution spectrum (e.g. using a "plate" to "cover a pastry bowl") are identifiable through significant similarity differences when compared to the set of players that stick to ordinary or obvious answers. This systematic distinction can be utilized to explicitly reward creative thinking if this is beneficial for the desired target application (i.e. robotic decision making). On the other hand, the set of creative answers can still be separated from disruptive players that do not take the game serious or only try to break it (e.g. using a "chainsaw" to "open a parcel"), which threaten the quality of the accumulated data. Eventually, the initially raised research questions can be answered as following:

- Incentivizing players to provide popular answers to common everyday activity tasks while successively eliminating obvious solutions can lead to the production of meaningful preference distributions among tools or objects. Thus, rankings of affordances and specified information for robotic decision making can directly be extracted from the response set.
- Utilizing the *wisdom of the crowd* effect, malicious responses can be identified by examining the similarity between the player's set of answers and the total solution set, while players that tend to give creative answers are still discriminable from the former.

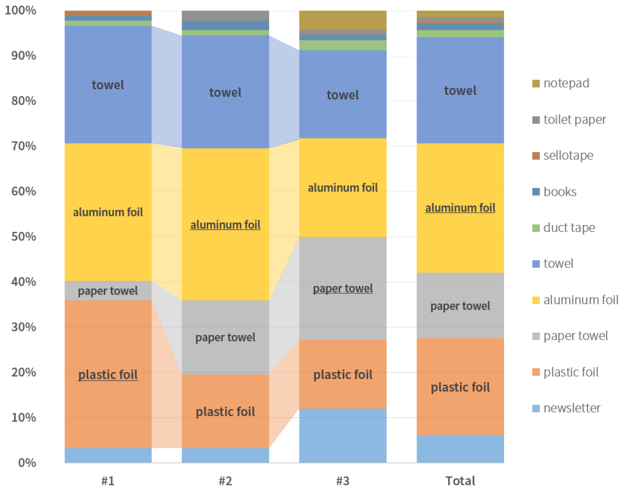


Figure 3: Response objects for the task *"Cover a pastry bowl with..."*. #1 displays the initially preferred solution distribution, while #2 and #3 eliminated the previous answers, and *Total* indicates the total overall selection ratio. The most frequent answers of each iteration are underlined.

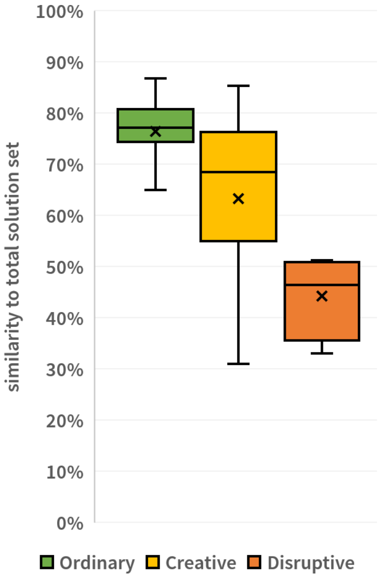


Figure 4: Similarity of the solutions to the total set from either *normal*, *creative* or *disruptive* players. Includes means (x), medians (–), one standard deviation (boxes) and range (whiskers).

7 LIMITATIONS AND FUTURE WORK

Apart from the convincing results that justify the use of a GWAP to augment robotic decision making, several limitations arose within this explorative pilot study. While the eliminative procedure showed to provide a wider set of possible answers, no insights about situations could be gathered where no popular (or viable) solutions at all remained to be chosen. These might evoke truly creative decision making, but on the other hand could also force players into submitting pointless solutions and/or harming their overall experience. Above that, the sample population might be biased by the nature of the acquisition, potentially neglecting answer preference differences among several age groups or cultural backgrounds. Since this pilot study was used to aggregate an initial solution dataset, adaptive and/or emergent game mechanics such as dynamic point/time bonus calculation (based on the similarity to the total set) could not be implemented so far, lacking a ground truth baseline. Thus, all of the aforementioned limitations will be addressed in a fully-fledged field study incorporating a drastically larger and continuously updated set of answers and possible solutions, the now aggregated dataset as a starting point for dynamic similarity calculation and an increased sample size when published on the relevant distribution platforms (i.e. Google Play, App Store). Apart from this, the manual labeling of the player responses into *ordinary*, *creative* and *disruptive* might be biased through subjective influences of the experimenters and is not feasible for automatic classification, but only served for this initial evaluation to demonstrate the possibilities for answer type separation. In future work, we are looking forward to include unsupervised machine learning techniques such as clustering [9] or player modeling [19] to separate the solution quality reliably and autonomously. Beyond the filtering of unwanted responses, we will further investigate the capabilities of these mechanisms with respect to balancing [15, 18] and difficulty adjustment [20, 21] to further strengthen the motivational pull for human computation. Finally, the procedure presented in this work is not limited to the domain of everyday activities or the field of robotics, but further opportunities present themselves in several areas that benefit from large-scale citizen knowledge, such as general opinion surveys, cultural studies or market research.

8 CONCLUSION

Real-world robotic application demands not only dynamic decision making architectures, but heavily relies on vast amounts of world knowledge to deal with unknown environments, constellations or tasks. While autonomous approaches of aggregating this world knowledge often end up with underspecified information, human computation can close this gap in reasoning by implementing human world knowledge into these processes, yet relying on the quality of the submitted data. By introducing *Tool Feud*, we presented an effective way of accumulating quantifiable human preference distributions to everyday activity tasks. Above that, contrasting individual players against the *wisdom of the crowd* has shown to reveal significant differences between players that supply viable responses and those who harm data quality. This work demonstrates a successful application of a human computation game with a purpose to facilitate research in the contexts of robotic learning and games user research.

For transparency, reproducibility and robotic knowledge aggregation, the accumulated data will be made publicly accessible through the Open Science Framework [1] as well as the robotic open collaboration database openEASE [2].

ACKNOWLEDGMENTS

We thank all participants. This work was funded by the German Research Foundation (DFG) as part of Collaborative Research Center (SFB) 1320 EASE - Everyday Activity Science and Engineering, University of Bremen (http://www.easecrc.org/), subproject H2.

REFERENCES

[1] Michael Beetz, Daniel Bessler, Andrei Haidu, Mihai Pomarlan, Asil Kaan Bozcuoglu, and Georg Bartels. 2018. Know Rob 2.0 - A 2nd Generation Knowledge Processing Framework for Cognition-Enabled Robotic Agents. *Proceedings - IEEE International Conference on Robotics and Automation* (2018), 512–519. https://doi.org/10.1109/ICRA.2018.8460964

[2] Michael Beetz, Lorenz Mösenlechner, and Moritz Tenorth. 2010. CRAM—A Cognitive Robot Abstract Machine for everyday manipulation in human environments. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 1012–1017.

[3] Hao Dang and Peter K Allen. 2010. Robot learning of everyday object manipulations via human demonstration. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 1284–1289.

[4] Johann Füller, Kathrin Möslein, Katja Hutter, and Jörg Haller. 2010. Evaluation games–How to make the crowd your jury. *INFORMATIK 2010. Service Science–Neue Perspektiven für die Informatik. Band 1* (2010).

[5] Mark Goodson. 1976. Family Feud. [Television broadcast]. *ABC, CBS, Syndicated.*

[6] Ece Kamar, Ashish Kapoor, and Eric Horvitz. 2013. Lifelong learning for acquiring the wisdom of the crowd. In *Twenty-Third International Joint Conference on Artificial Intelligence*.

[1] https://osf.io/

[2] http://www.open-ease.org/

[7] Markus Krause, Aneta Takhtamysheva, Marion Wittstock, and Rainer Malaka. 2010. Frontiers of a paradigm: exploring human computation with digital games. In *Proceedings of the acm sigkdd workshop on human computation*. ACM, 22–25.

[8] Lars Kunze, Moritz Tenorth, and Michael Beetz. 2010. Putting people's common sense into knowledge bases of household robots. In *Annual Conference on Artificial Intelligence*. Springer, 151–159.

[9] James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, Vol. 1. Oakland, CA, USA, 281–297.

[10] Sebastian Matyas, Peter Kiefer, Christoph Schlieder, and Sara Kleyer. 2011. Wisdom about the crowd: Assuring geospatial data quality collected in location-based games. In *International Conference on Entertainment Computing*. Springer, 331–336.

[11] Lorenz Mösenlechner and Michael Beetz. 2013. Fast Temporal Projection Using Accurate Physics-Based Geometric Reasoning. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. Karlsruhe, Germany, 1821–1827.

[12] Rodney D Nielsen, Richard Voyles, Daniel Bolanos, Mohammad H Mahoor, Wilson D Pace, Katie A Siek, and Wayne H Ward. 2010. A platform for human-robot dialog systems research. In *2010 AAAI Fall Symposium Series*.

[13] Nils J Nilsson. 1984. *Shakey the robot.* Technical Report. SRI INTERNATIONAL MENLO PARK CA.

[14] Natalie Paige Parde, Michalis Papakostas, Konstantinos Tsiakas, Maria Dagioglou, Vangelis Karkaletsis, and Rodney D Nielsen. 2015. I spy: An interactive game-based approach to multimodal robot learning. In *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[15] Johannes Pfau, Antonios Liapis, Georg Volkmar, Georgios Yannakakis, and Rainer Malaka. 2020. Dungeons & Replicants: Automated Game Balancing via Deep Player Behavior Modeling. In *2020 IEEE Conference on Games (CoG)*. IEEE.

[16] Johannes Pfau and Rainer Malaka. 2019. Can You Rely on Human Computation? A Large-scale Analysis of Disruptive Behavior in Games With A Purpose. In *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*. 605–610.

[17] Johannes Pfau, Robert Porzel, Mihai Pomarlan, Vanja Sophie Cangalovic, Supara Grudpan, Sebastian Höffner, John Bateman, and Rainer Malaka. 2019. Give MEANinGS to Robots with Kitchen Clash: A VR Human Computation Serious Game for World Knowledge Accumulation. In *Joint International Conference on Entertainment Computing and Serious Games*. Springer, 85–96.

[18] Johannes Pfau, Jan David Smeddinck, Ioannis Bikas, and Rainer Malaka. 2020. Bot or not? User Perceptions of Player Substitution with Deep Player Behavior Models. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM.

[19] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2018. Towards Deep Player Behavior Models in MMORPGs. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*. ACM, 381–392.

[20] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2019. Deep Player Behavior Models: Evaluating a Novel Take on Dynamic Difficulty Adjustment. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, LBW0171.

[21] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2020. Enemy Within: Long-term Motivation Effects of Deep Player Behavior Models for Dynamic Difficulty Adjustment. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM.

[22] Richard M Ryan, C Scott Rigby, and Andrew Przybylski. 2006. The motivational pull of video games: A self-determination theory approach. *Motivation and emotion* 30, 4 (2006), 344–360.

[23] Jakub Simko, Michal Tvarozek, and Maria Bielikova. 2013. Human computation: Image metadata acquisition based on a single-player annotation game. *International Journal of Human-Computer Studies* 71 (10 2013), 933–945. https://doi.org/10.1016/j.ijhcs.2013.05.002

[24] Kristin Siu and Mark O Riedl. 2016. Reward systems in human computation games. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play*. ACM, 266–275.

[25] Luis Von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. 2008. recaptcha: Human-based character recognition via web security measures. *Science* 321, 5895 (2008), 1465–1468.

Enhancing Game-Based Learning Through Infographics in the Context of Smart Home Security

Mehrdad Bahrini[1], Nima Zargham[1], Johannes Pfau[1], Stella Lemke[1], Karsten Sohr[1], and Rainer Malaka[1]

Digital Media Lab, TZI, University of Bremen, Bremen, Germany
{mbahrini,zargham,jpfau,slemke,sohr,malaka}@uni-bremen.de

Abstract. Constantly evolving advances of smart home devices features require users to persistently keep up with safety concerns. While update reports and news articles are common ways to keep them informed, many users struggle in thoroughly understanding and applying available security recommendations. Educational games have proven to be an intuitive way to increase the incentive for awareness but many of them come short to convey the needed supporting knowledge. In an attempt to raise security awareness on smart home devices, we designed an educational game to demonstrate the latest security challenges and solutions. To ascertain users' attention and motivation, we have developed two versions of the game to contrast the integration of text and infographics as supporting knowledge which are the hints in this case. Our evaluations give evidence that viewing security-related content with a higher deployment of infographics improves users' performance significantly, increases users' interest in the topic, and creates higher levels of confidence solving security problems and complexities.

Keywords: Usable Security · Smart Home · Educational Games · Supporting Knowledge · Infographics.

1 Introduction

Educational games (edu-games) have shown great potential in being a powerful teaching tool as they can increase engagement, creativity and authentic learning [23, 38, 60]. Game-based learning allows users to see themselves in simulated real situations where they can learn through experience and solve the problems through critical thinking [13]. Furthermore, the motivational power of game-based learning towards specific subjects is widely recognised [31]. Harnessing the intrinsically motivating power of games, researches have shown that edu-games can be a great tool to promote user engagement and improve positive usage patterns, such as increasing user activity, social interaction, and the quality and productivity of user actions [22, 37]. Previous work has shown that edu-games can be useful in raising the knowledge and awareness of the users [3, 60], but this alone can not get the best out of the learning experience.

In edu-games, feedback plays a key role in providing the user the necessary information for the learning experience. Using in-game feedback is intended to guide learners to improve their performance, and increase motivation or learning outcomes by providing them with information on the accuracy of their answers in various ways [62]. According to Johnson et al. [35], these feedback messages can be classified into two types. *Outcome-oriented* feedback delivers information to learners about their progress or the accuracy of their answers (e.g. which is the correct answer and why). *Process-oriented* feedback provides learning guidance and supporting knowledge on the processes as well as strategies used to achieve the correct answer or action in the game. Examples of process-oriented feedback are prompts and hints that lead the learners towards the right answer. In many video games, supporting knowledge is used to inform the players about their objectives and guide them throughout the game. This form of process-oriented feedback could be leveraged to improve the effectiveness of educational games [56]. The supporting knowledge can be given to the users in different forms such as text, images, audio, and video, to provide explicit guidance to players as they play the game [35]. In this paper, we study the use of infographics as a way to convey information to the players in an edu-game.

Infographics are a graphical representation of information or knowledge [33]. They are essentially an effective visual representation that explains information simply and quickly using a combination of text and graphical symbols. Some commercial games such as Metrico+ [24], Mini Metro [16], and Lumino City [57] have implemented infographics as their look-and-feel or even game mechanic and have received very positive reviews from the users. Infographics can motivate players and exploit the visual potential to represent and convey knowledge. They aim to increase the amount of information people remember by breaking them into concise, visually attractive chunks of data. This way, the learners can remember more, leading to improvement in their capabilities [8]. Although utilizing infographics have shown to be effective in transferring information, the implementation of infographics in edu-games is still under investigation.

Recent innovations in technology and the rise of inter-connectivity between devices enable the development of innovative solutions in the field of smart homes to take advantage of these opportunities. Along with this rapid development, the security and privacy of users has always been a concern. Making smart home devices more secure may partly address this concern, but users also have a complementary role in protecting their sensitive information. However, users' understanding and ability to adopt and configure the security of smart home devices is not integrated. As users face a plethora of innovations as well as the ever-expanding spread of security news and journals, it has become increasingly difficult for non-tech-savvy users to understand and apply security guidance. Games have long been recognized as an effective and appealing educational strategy in the field of computer security and privacy [61]. This approach has been used to teach various topics related to security [29, 21].

We have designed an edu-game with the aim of aiding owners of smart home devices to get acquainted with security issues and recent risks. Players are asked to find potential smart home devices in different rooms and answer questions about the respective device, helping a virtual smart home owner to protect his home from attacks. For contentual assistance during the game, players have the opportunity to assess security instructions concerning the respective device. Within our evaluation, this information is presented textually (analogous to conventional safety reports or updates) or visualized using infographics, as a structured combination of text, images, charts, and icons. Eventually, infographics aim to enable effective representation of data and explain complex problems in a clear and understandable way [30]. Using a between-subjects design, we investigate the users' motivation and evaluate the impact of infographics on players, to answer the following research question: *To what extent can infographics as supporting knowledge improve the learning experience of users and make learning more effective in an educational game in a smart home security context?*

Our results indicated a significant amount of correct answers, as well as an increase of perceived competence by the introduction of infographics. Harnessing this motivation and illustration potential, this paper augments the area of educational serious games with immediately comprehensible knowledge representation and provides evidence that players are more effective, motivated and spend more time on self-education by the implementation of infographics.

2 Related Work

2.1 Game-Based Learning for Security Topics

Game-based learning uses different techniques to manipulate the behavior of users in the direction of a specific goal within a non-gaming context [27]. For example, it can be utilized as a marketing strategy to promote products or services or for training and simulating complex environments virtually [70]. Games can establish the facilitation of enjoyment and engagement by increasing intrinsic motivation, in contexts that are primarily extrinsically motivated. Game-based learning approaches, especially mobile learning [28], are a relatively new approach to security education. A study comparing the use of text, videos, and games found that mobile learning can raise awareness of security issues and teaches users more effectively in comparison to the traditional text-based and video-based learning materials [1].

Research studies showed that serious games provide promising ways to change cybersecurity behaviour [19]. Bahrini et al. [6] developed a gamified application that helps users to understand the consequences of granting permissions to the applications. Their results showed that playing the gamified application results in a significant increase of player enjoyment and that the game is more informative than the traditional approach of permission administration via the Android system settings.

In an attempt to raise interest and awareness towards the topic of privacy and security settings of mobile devices, Zargham et al. developed a humorous decision-making game that helps users to better understand the consequences of applying security changes on a mobile device [68]. They compared their game to two more models (a serious animated video and a humorous animated video) and found that the game-based approach is more successful in engaging and raising awareness.

Wen et al. designed and developed a role-playing game to engage users to learn more about phishing threats in an active and entertaining manner [67]. Their study showed that the game raises awareness towards the topic and enhances anti-phishing self-efficacy facing phishing emails. Chen et al. presented a desktop game, aiming to change cybersecurity behavior by translating self-efficacy into game design [14]. Their results showed that the game experience could improve users' confidence in tackling security issues.

Many studies have explored the effectiveness of games for increasing cybersecurity awareness, however, most of them have focused primarily on factors of entertainment or engagement of such games, and very little on the learning effect and behavioural change in users [32, 2].

2.2 Supporting Information in Game-Based Learning

Edu-games are seen as one of the most promising forms of computer-based education and multiple studies have shown their highly engaging potentials [54, 34]. Nonetheless, there is less support for their educational effectiveness [20, 66, 43]. Many of the existing work did not evaluate the effectiveness of the components used in an edu-game. One element of the game that is particularly easy to adapt and can have a considerable influence on motivation is feedback. Studies have indicated that in computer-based learning environments, feedback can be a confirmation of a correct answer or an explanation or recommendation in detail. Detailed feedback has a greater impact on learning outcomes and motivation than simple feedback, but this depends on the learners' attention and ability to correct their actions [11, 59].

In an attempt to study the effectiveness of hints, O'Rourke et al. gathered data from 50,000 students and compared four different hint designs based on successful hint systems in intelligent tutoring systems and commercial games [52]. Their results showed that all four hint systems negatively impacted performance compared to a baseline condition with no hints. Authors also suggest that traditional hint systems may not translate well into the educational game environment.

Appropriate presentation of the feedback could have a considerable impact on the effectiveness of the players and can promote deep, meaningful learning [50]. Studies have shown that people learn more deeply when words are presented in spoken form rather than in printed form [55, 25]. However, they did not suggest that feedback should always be presented as spoken words. In this paper we evaluate an approach for comparing text and infographics as *process-oriented* feedback and their impact on the user experience and game outcome.

2.3 Infographics

Information is remembered better when it is supported with pictures [42]. The use of visual information during learning and instructional processes offers many advantages. Studies showed that if a text is followed by illustrations, learners retain information for longer and are more likely to remember it [18, 47, 46, 5, 15, 53]. Infographics are a powerful way to distill and explain complex information as a visual narrative and constitute an effective way of communicating data to decision makers who need high-quality information in a bite-sized and easily accessible form [41]. Visual embellishments, including recognizable comics and images make infographics effective and improve data presentation and memorability [7, 9].

Studies show that infographics brings various modalities together in the hope that they will be understood by a wider audience, regardless of their ability to learn. Infographics use text and illustrations or images to inspire readers to better remember the information presented [45]. Following a study by Kay and Terry [39], they argued that inclusion could be achieved through the use of iconic symbols, short facts and captions as a means of highlighting relevant important information in complex documents. Similarly, Knijnenburg and Cherry [40] suggested using comics as a more inviting, understandable and engaging medium to improve the communication of privacy notices.

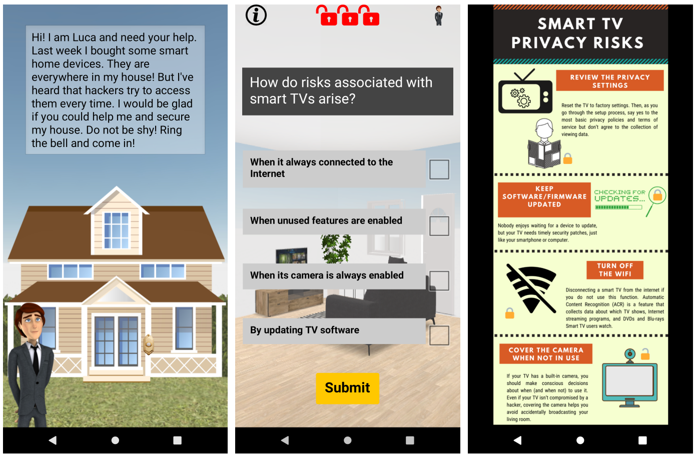
Unlike the efforts made to explore the effects of infographics [41, 45], research on the use of infographics in edu-games has not been studied thoroughly. This paper showcases the potential of using infographics embedded in an educational game. We aim to aid users in becoming more familiar with security concepts of their devices and motivate them to increase their knowledge on the topic. Our approach is focused on providing efficient process-oriented feedback in the context of security to help with the understanding of security issues in the smart home environment.

3 Approach

We designed an educational game that uses infographics as supporting knowledge in an approach to raise players' awareness and increase their interest towards smart home security issues. The learners explore the game levels to interact with smart devices and answer a number of security questions (see Figure 1). The provided supporting information helps the players to answer the questions and gain a deeper understanding of the new security concerns of smart devices.

The game was developed for mobile platforms and has an ordinary person narrative. At the beginning of the game, the player meets the character "Luca" in front of his home, who is worried about the security of his smart home devices. Luca has less understanding of how to configure the smart devices. He asks the player to help him by searching devices and answering related questions. The player enters Luca's home by ringing the doorbell. There are five rooms in the house, each including two smart devices. Each time the player enters a room, a mellow background music is played. The player should tap on each of the devices to display a question. For each question screen, there is also a hint button that helps the player to obtain supporting knowledge to answer the question. After submitting an answer, the game evaluates it and displays a notification. Eventually, the player is awarded based on the number of correct answers at the end of the game.

During the game, users have to answer ten questions where each question is aimed at one smart device. A number of factors were assessed for the selection of devices. It is essential to have a router in the home network. Since most devices are connected to the network via an app, we have considered choosing a smartphone as an intelligent device. We have also selected 6 devices (Smart TV, IP Camera, Smart Speaker, Smart Thermostat, Smart Lamp, Smart Plug) that most smart home owners are familiar with. To arouse the players' curiosity, the last two devices, Smart Home Firewall and Smart Mowing Robot, were chosen.



Hi! I am Luca and need your help. Last week I bought some smart home devices. They are everywhere in my house! But I've heard that hackers try to access them every time. I would be glad if you could help me and secure my house. Do not be shy! Ring the bell and come in!

How do risks associated with smart TVs arise?

When it always connected to the Internet □

When unused features are enabled □

When its camera is always enabled □

By updating TV software □

SMART TV PRIVACY RISKS

REVIEW THE PRIVACY SETTINGS

Reset the TV to factory settings. Then, as you go through the setup process, say yes to the most basic privacy policies and terms of service but don't agree to the collection of viewing data.

KEEP SOFTWARE/FIRMWARE UPDATED

Nobody enjoys waiting for a device to update, but your TV needs timely security patches, just like your smartphone or computer.

TURN OFF THE WIFI

Disconnecting a smart TV from the internet if you do not use this function. Automatic Content Recognition (ACR) is a feature that collects data about which TV shows, Internet streaming programs, and DVDs and Blu-rays Smart TV users watch.

COVER THE CAMERA WHEN NOT IN USE

If your TV has a built-in camera, you should make conscious decisions about when (and when not) to use it. Even if your TV isn't compromised by a hacker, covering the camera helps you avoid accidentally broadcasting your living room.

Fig. 1. The game helps players to get acquainted with security issues of smart home devices. Narrative (left), question (middle), and supporting knowledge screens (right).

3.1 Question Scenarios

The selected question for each device is based on the security and privacy concerns that have been addressed as threat models in research and articles in recent years [69, 58]. Consequently, 10 recommendations have been selected that are closer to the daily life of the users. Certainly, there is no doubt that the number of available recommendations is very large. However, all these items must be taken into account in the device settings. The following is an overview of the selected questions:

- *Router:* Setting up routers might be a tedious task for non-tech-savvy users. Although companies provide manuals, there is not enough information about the security issues caused by incorrect settings. Users have difficulties with understanding the configurations such as setting a secure admin password, choosing an appropriate protocol to encrypt the connection and utilizing technologies such as Wi-Fi Protected Setup (WPS) [36]. Consequently, the router question concerns which setup could help to have a secure router.
- *Smartphone:* Nowadays, smartphones are very popular and a convenient means of accessing and controlling smart home devices. Applications are being developed and are available for download from App Stores. The use of a fake, unofficial or outdated applications could lead to security problems for users' data and also for smart home devices [63]. Hence, we ask the players how an application could cause a security breach for smart devices.
- *Smart TV:* New generation of TVs integrate an operating system running multiple applications and an internet connection, allowing them to offer more services to users, however this might raise security concerns [4]. Webcam hacking, tracking problems and outdated software pose threats to user privacy [1]. In this scenario, users are encouraged to examine their understanding of these security and privacy issues.
- *IP Camera:* The IP cameras allow users to monitor their properties. It is easy to set up and does not require complex configuration. Users can also use an application to access the camera at any time and from anywhere. These functions are interesting for hackers. Various types of security attacks on the internet have become a serious threat to the video stream from IP cameras [17]. Therefore, users are advised to configure a variety of security recommendations, such as camera passwords, use of up-to-date applications and video encryption to protect against these threats [2]. This question investigates whether users understand the basic settings of a secured IP camera.
- *Smart Speaker:* It is easy to neglect that intelligent assistants are designed to be at the heart of smart home systems. While they allow users to surf the Internet, they can communicate and control other internet-enabled technologies at home. Recently, it was discovered that one type of attack allows hackers to secretly communicate with your device via white noise or YouTube videos - so they can send text messages or open malicious websites without the owners knowing [12]. Providing users with information about such harmful attacks helps them to protect their voice assistants from being attacked unwanted.
- *Smart Thermostat:* Controlling the smart thermostat via apps on smartphones allow the users to raise or lower the temperature remotely. The smart thermostats could create a gap in privacy and security of smart home networks, precisely because they learn about your habits and behaviour. Hackers could attack the vulnerable thermostat and get information about when users are not home, so they know when to break in without worrying about

[1] https://us.norton.com/internetsecurity-iot-smart-tvs-and-risk.html

[2] https://www.consumer.ftc.gov/articles/0382-using-ip-cameras-safely

users returning [26]. Such complex scenarios should be deeply understandable to users in order to protect their information and properties from attackers. The aim of this question is to inform users about the risks if someone gaining access to a smart thermostat.

- *Smart Lamp:* By connecting a smart lamp to the home network, users can control the brightness and sometimes change the color of the light from their smartphone. This provides more advanced features such as connecting the lamp to an alarm clock or flickering the desk lamp when new messages are received. These facilities are sometimes associated with security problems that could cause health and financial damages [51]. The purpose of this question is to provide users with recommendations to improve their knowledge to better decide how to purchase a suitable and secure intelligent lamp.
- *Smart Plug:* Smart plugs with cloud connection enable users to monitor and control electronic household appliances from anywhere. To manage them over the Internet, users should have a cloud account on the manufacturer's website or application and register the smart plug devices in the cloud service. However, they may suffer from insecure communication protocols and lack of device authentication [44]. With this question, we investigate the player's knowledge about user profile creation and understanding why the authentication and authorization of smart plug on the cloud server is important.
- *Smart Home Firewall:* By connecting smart devices to each other and to the Internet, smart home applications automate complex household tasks. Keeping track of the actions performed and controlling data communication could be confusing for inexperienced users. Rules for firewalls help protect the home network from malicious attacks as well as controlling the security vulnerabilities [65]. In this scenario, we encourage players to consider getting familiar with the firewall and the role of using them in smart home networks.
- *Smart Mowing Robot:* Mowing robots are becoming increasingly intelligent. They use GPS information to calculate the desired location and have an internet connection that enables them to communicate with cloud services and their applications. This scenario examines the advantages of using VPN when the user is away from home and wants to access the home network via a public Wi-Fi hotspot to take control of the smart mowing robot [49].

3.2 Game Procedure

The game consists primarily of the following building blocks:

- Finding devices: Players should find two devices in each room and answer the following questions regarding these.
- Request help: During the game, players may lack background knowledge to answer the questions. This event gives users insights about the context of the smart device and related security issues.
- Feedback of answers: After the player submits an answer, the game displays the result. If the answer was wrong, the player will receive the correct answer.

After starting a game session, the avatar will be displayed, expressing his goal via a textual speech bubble. By tapping on the doorbell, the player goes to the next state of the game (see Figure 1).

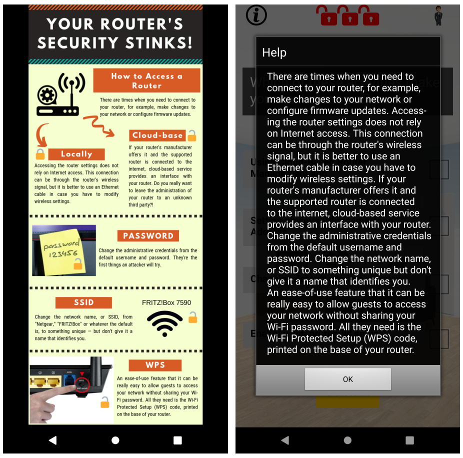
Question: Once the player enters a room, there are two available smart devices. By clicking on one of them, the question screen will be shown. All questions are multiple choice and the game informs the players while choosing the first device. On the top of the question screen, the player finds two buttons: The hint button on the left displays the supporting knowledge about the device's security, while the avatar icon on the right explains general game controls (see Figure 1). The player is directed to proceed to the next room after answering two questions.

Progression: Each play-through consists of 10 questions. Luca's home will become more secure, proportional to the number of correct answers. In order to transfer this concept to the player, 3 open red locks are displayed at the start of the game. Each of these locks turns into green closed locks after three correct answers given by the player. With 9 correct answers the player could get 3 green closed locks.

Supporting knowledge: By clicking on the information icon, the player is directed to the supporting knowledge screen. For the comparison of using text and infographics regarding their effect on player's motivation and performance, either text or infographics are displayed (see Figure 2). The content provided for the supporting knowledge is exactly the same for both versions. Every question includes a different supporting knowledge, separated from other questions. As for the used infographics, Various symbols have been added to transfer the concepts to the players and to increase their attention. A caption was selected for each infographic based on the associated device. For every device, we also designed symbols that convey basic concepts about device configuration or physical forms. To express the concept of being secure and insecure, there is a closed or open lock icon next to the titles or symbols. These concepts were applied to all infographics. The backyard is considered as the last room. By answering the two related questions, the player is directed to the reward interface where the number of correct answers and the corresponding reward are displayed on the screen.

4 Evaluation

To evaluate our research question, we conducted a between-subjects design user study with 60 participants. Within the first group (Text-Group), we evaluated with ($n = 30$) participants, using descriptive textual background information in the supporting knowledge screen. The second group (Infographics-Group) contained also ($n = 30$) participants, mutually excluded from the first, and introduced infographics instead of text in the supporting knowledge screen. We conducted laboratory study sessions on the university campus, with one participant per session and a duration of 30 to 45 minutes. As a mobile device, we provided a Google Pixel 2 XL with Android 9.0.



YOUR ROUTER'S SECURITY STINKS!

How to Access a Router

There are times when you need to connect to your router, for example, make changes to your network or configure firmware updates.

Locally

Accessing the router settings does not rely on Internet access. This connection can be through the router's wireless signal, but it is better to use an Ethernet cable in case you have to modify wireless settings.

Cloud-base

If your router's manufacturer offers it and the supported router is connected to the internet, cloud-based service provides an interface with your router. Do you really want to leave the administration of your router to an unknown third party?!

PASSWORD

Change the administrative credentials from the default username and password. They're the first things an attacker will try.

SSID

Change the network name, or SSID, from "Netgear," "FRITZ!Box" or whatever the default is, to something unique – but don't give it a name that identifies you.

WPS

An ease-of-use feature that it can be really easy to allow guests to access your network without sharing your Wi-Fi password. All they need is the Wi-Fi Protected Setup (WPS) code, printed on the base of your router.

Help

There are times when you need to connect to your router, for example, make changes to your network or configure firmware updates. Accessing the router settings does not rely on Internet access. This connection can be through the router's wireless signal, but it is better to use an Ethernet cable in case you have to modify wireless settings. If your router's manufacturer offers it and the supported router is connected to the internet, cloud-based service provides an interface with your router. Change the administrative credentials from the default username and password. Change the network name, or SSID to something unique but don't give it a name that identifies you. An ease-of-use feature that it can be really easy to allow guests to access your network without sharing your Wi-Fi password. All they need is the Wi-Fi Protected Setup (WPS) code, printed on the base of your router.

Fig. 2. Supporting knowledge screen: Infographics (left) and Text (right).

1. The interviewer provided an introduction about the game and security problems about smart home devices to the player.
2. The player ran the game, entered the rooms and answered related questions. Play time was measured.
3. After the game was over, the player answered a number of questionnaires.
   (a) The first questionnaire contained general questions regarding demographic information (e.g. age and gender).
   (b) In order to measure the usability of the game, the second questionnaire consisted of the System Usability Scale (SUS) [10].
   (c) Motivation of the player was measured by utilizing the Intrinsic Motivation Inventory (IMI) [48] on a 7 point Likert-scale.
   (d) Beside standard questionnaires, we had a number of self-designed context questions. The purpose of these questions was to understand the backgrounds of the players and their familiarity with smart devices.

4.1 Participants

A quota sampling approach was used to recruit participants for this study in which the selection was based on mailing lists, social networks, word-of-mouth and looking for users of smart home devices. Participation was voluntary and uncompensated. The first group consisted of 30 participants, 9 participants had a college degree, while 21 completed high school. Among the subjects, 15 people identified themselves as male and 15 as female. In terms of age, participants ranged between 18 to 54 years with an average age of 28.9 ($SD = 10.25$). The second group consisted of 30 participants, 14 participants had a college degree, while 16 completed high school. Among the subjects, 15 people identified themselves as male and 15 as female. In terms of age, participants ranged between 21 to 44 years with an average age of 30.6 ($SD = 6.38$).

5 Results

Statistical analysis was applied to identify possible differences between the two groups. To determine the impact of infographics on the players, the data from both groups were compared to each other.

After playing the game, participants were also asked to select all the smart home devices they own to see which devices are most commonly used amongst them. It turned out that all participants in the Text-Group owned at least one smart device in their homes and all of them had a smartphone. Table 1 shows an overview of the smart devices owned by the participants in the Text-Group.

Table 1. The number of smart devices owned by the participants in both groups

	Number of Devices	
	Text-Group	Infographics-Group
Smart TV	25	29
Smart Lamp	12	10
Smart Speaker	9	10
Smart Plug	3	2
IP Camera	2	3
Smart Thermostat	2	1
Smart Mowing Robot	0	0
Smart Firewall	0	0

The calculated mean value of SUS score for the Text-Group was 89.9 ($N = 30$, $SD = 14.70$). The IMI score of *Interest-Enjoyment* was rated 6.2 ($SD = 0.78$), *Perceived Competence* score was rated 3.4 ($SD = 0.1$) and *Effort-Importance* score was rated 5.6 ($SD = 0.97$). The average of correct answers was 2.4 ($SD = 0.17$) and the average play time was 9.27 minutes ($SD = 1.36$).

In the Infographics-Group, participants were also asked to select all the smart home devices they own. The results showed that all participants in this group also owned at least one smart device in their homes and had smartphones (see Table 1).

The calculated mean value of SUS score for this group was 84.0 ($N = 30$, $SD = 7.32$). The IMI score of *Interest-Enjoyment* was rated 6.0 ($SD = 0.65$), *Perceived Competence* score was rated 5.8 ($SD = 0.39$) and *Effort-Importance* score was rated 5.6 ($SD = 0.84$). The average of correct answers was 7.3 ($SD = 0.15$) and the average play time was 14.77 minutes ($SD = 2.89$).

The independent student's t-Tests [64] revealed that the participants in the Infographics-Group ($M = 7.3, SD = 1.15$) who received supporting knowledge in the form of infographics demonstrated significantly better average of correct answers ($t(58) = 11.734, p < .001, Cohen'sd = 3.030$) compared to the Text-Group participants ($M = 2.4, SD = 1.70$) (see Figure 3).

For average of playing time between two groups, the independent t-tests indicated that Infographics-Group participants ($M = 14.77, SD = 2.89$) showed a significantly higher average playing time ($t(58) = 9.441, p < .001, Cohen'sd = 2.438$) compared to the Text-Group participants ($M = 9.27, SD = 1.36$) (see Figure 3).

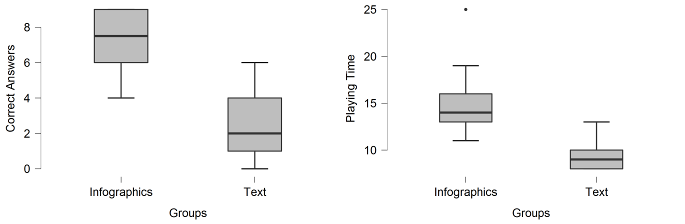


Fig. 3. The number of correct answers (left) playing time (right).

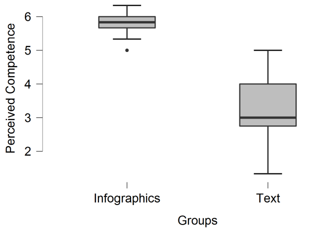


Fig. 4. The score of IMI test (Perceived Competence).

For IMI's *Perceived Competence* scores, independent t-Tests showed that the Infographics-Group ($M = 5.8, SD = 0.39$) significantly outperformed ($t(58) = 12.456, p < .001, Cohen'sd = 3.216$) the Text-Group ($M = 3.4, SD = 0.1$) (see Figure 4). We did not witness any significant differences in *Interest-Enjoyment* ($t(58) = 1.317, p = .193$), and *Effort-Importance* ($t(58) = 0.237, p = .814$) of IMI between the two groups.

Also, no significant differences in the SUS scores ($t(58) = 1.364, p = .178$) between the two groups could be found.

6 Discussion & Limitations

The purpose of this study was to investigate how a particular style of feedback, in this case infographics, affects the performance of edu-game players in the context of smart home security. Ultimately, the aim of this experiment was to provide answers to the comprehensive question: *To what extent can infographics as supporting knowledge improve the learning experience of users and make learning more effective in an educational game in a smart home security context?*

Results from the user study indicate that the game has a distinct usability and players enjoyed playing it, regardless of the difference in the form of supporting knowledge. Furthermore, our results showed high engagement towards the topic for the people who played the game. Participants were eager to spend time playing the game in both groups.

Players in the Infographics-Group answered significantly more questions correctly compared to the Text-Group. We evaluated that users performed better when they got infographics as supporting knowledge. Due to high complexity of the topic, the questions could be considered as difficult for the average user. However, participants in the Infographics-Group performed reasonably well. This could indicate that using infographics as supporting knowledge could improve the performance of players in an edu-game even when the topic is rather difficult for the average user.

The resulting IMI *Perceived Competence* scores indicate that reading and viewing infographics considerably raise the players' confidence. The IMI (*Effort-Importance*) scores also show that the players made an effort to answer the questions in both groups. However, they were significantly less successful in terms of performance in the Text-Group. Even though participants were eager to answer the questions in both games, the infographics scored better. This could be evidence that not only a difference in motivation leads to the increase in correct answers, but the technical understanding was actually improved.

Although there was a significant difference in terms of (*Perceived Competence*), We did not witness any significant difference in terms of (*Interest-Enjoyment*) and (*Effort-Importance*) in the IMI results.

Nonetheless both groups rated very high absolute scores for these subgroups. This indicates that both versions managed to foster intrinsic motivation and raise players' interest and effort towards the topic regardless of the form of supporting knowledge.

Many of the game questions were selected from the security content which are available on web pages and users may read them throughout their daily life. It should be stressed that understanding the wording and sentences of questions could also affect the results. Based on performances of the players and their comments after the experiment, we found out that the difficulty of the questions were perceived differently between participants. Therefore, for the future we suggest to designing questions and creating levels based on complexity and difficulty of the topic. Users' playing time on average was observed significantly higher in the Infographics-Group than the Text-Group. One could argue that the difference in play time has an effect on the learning experience of the players. Although this might be true, nonetheless, it could indicate that the users would spend more time on the information if it's visualized with infographics rather than text which further will lead to a better learning experience. For future research, we suggest implementing a fixed time period for all conditions in which the player can access the supporting knowledge in order to focus more on the evaluation of the provided supporting knowledge and minimize other possible effects on the learning experience.

The game was characterized as a simple quiz-genre type, thus other game genres could be evaluated to extend the findings within different game genres. Our approach was aimed to help users gain more knowledge on how to make specific security decisions and raise their awareness towards smart home security issues. This knowledge can later help players to make more informed decisions while configuring and setting up their smart home environment. One should keep in mind that it is crucial for educational games in the context of privacy and security to be updated regularly based on recent changes and updates to provide the latest information on the topic.

While these results present some significant steps forward in the investigation of using infographics as supporting knowledge in the context of smart home security, there are still some limitations that should be addressed. This experiment investigated how well a person performed in answering a question in an edu-game environment when they received two different feedback interventions. Although significant differences in performance between the conditions were found, there was no direct measurement of long-term learning after training. Furthermore, individual difference factors such as playing experience or learning type as well as the background knowledge can also lead to differences in players' performance. Although the question criteria used in this experiment were carefully calibrated from many research materials, they were limited to 10 items. It is possible that these criteria were still not specific enough. To understand the full impact of different approaches in game-based learning, future research needs to examine its potential effects in terms of alternative types of instructional support, as well as possible differential effects of timing (e.g., near real-time, delayed).

7 Conclusion & Future Work

This paper presents a novel approach to facilitate awareness and motivation as well as enhancing learning experience in an educational game by using infographics as supporting knowledge. We present a game that increases the intrinsic motivation of users and gives them more self-confidence in terms of the smart home security concerns. Our study shows that the adoption of infographics as supporting knowledge helps users to gain a better understanding of the complex context during the game and allows the players to produce a more engaging output. Our game has shown great potential in terms of usability and, according to most players, can be used to educate people about smart home security concerns. The extent to which users can remember the solutions and security recommendations remains a question for future work. Based on the results of this evaluation, we will attempt to assess the learnability of the topic through the game and the knowledge progress of the users by means of pre- and post-questions and additional smart home devices, questions and problems. The impact of the graphical elements used in the infographics for the purpose of privacy and security learning is also a topic for the future work.

8 Acknowledgement

This work was supported by the German Federal Ministry of Education and Research (BMBF) under the grant 16SV8503 (UsableSec@Home project).

References

1. Abawajy, J.: User preference of cyber security awareness delivery methods. Behav. Inf. Technol. **33**(3), 237–248 (Mar 2014). https://doi.org/10.1080/0144929X.2012.708787
2. Alotaibi, F., Furnell, S., Stengel, I., Papadaki, M.: A review of using gaming technology for cyber-security awareness. Int. J. Inf. Secur. Res.(IJISR) **6**(2), 660–666 (2016)
3. Arachchilage, N.A.G., Love, S., Beznosov, K.: Phishing threat avoidance behaviour: An empirical investigation. Computers in Human Behavior **60**, 185–197 (2016)
4. Bachy, Y., Nicomette, V., Kaâniche, M., Alata, E.: Smart-tv security: risk analysis and experiments on smart-tv communication channels. Journal of Computer Virology and Hacking Techniques **15**(1), 61–76 (2019)
5. Baddeley, A.D.: Human memory: Theory and practice. Psychology Press (1997)
6. Bahrini, M., Volkmar, G., Schmutte, J., Wenig, N., Sohr, K., Malaka, R.: Make my phone secure!: Using gamification for mobile security settings. In: Proceedings of Mensch Und Computer 2019. pp. 299–308. MuC'19, ACM, New York, NY, USA (2019). https://doi.org/10.1145/3340764.3340775
7. Bateman, S., Mandryk, R.L., Gutwin, C., Genest, A., McDine, D., Brooks, C.: Useful junk?: The effects of visual embellishment on comprehension and memorability of charts. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 2573–2582. CHI '10, ACM, New York, NY, USA (2010). https://doi.org/10.1145/1753326.1753716
8. Bellato, N.: Infographics: A visual link to learning. ELearn **2013**(12) (Dec 2013). https://doi.org/10.1145/2556598.2556269
9. Borkin, M.A., Vo, A.A., Bylinskii, Z., Isola, P., Sunkavalli, S., Oliva, A., Pfister, H.: What makes a visualization memorable? IEEE Transactions on Visualization and Computer Graphics **19**(12), 2306–2315 (Dec 2013). https://doi.org/10.1109/TVCG.2013.234
10. Brooke, J.: Sus: a retrospective. Journal of usability studies **8**(2), 29–40 (2013)
11. Burgers, C., Eden, A., [van Engelenburg], M.D., Buningh, S.: How feedback boosts motivation and play in a brain-training game. Computers in Human Behavior **48**, 94 – 103 (2015). https://doi.org/https://doi.org/10.1016/j.chb.2015.01.038
12. Carlini, N., Mishra, P., Vaidya, T., Zhang, Y., Sherr, M., Shields, C., Wagner, D., Zhou, W.: Hidden voice commands. In: 25th USENIX Security Symposium (USENIX Security 16). pp. 513–530. USENIX Association, Austin, TX (Aug 2016), https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/carlini
13. Chang, C.Y., Hwang, G.J.: Trends in digital game-based learning in the mobile era: a systematic review of journal publications from 2007 to 2016. International Journal of Mobile Learning and Organisation **13**(1), 68–90 (2019)
14. Chen, T., Hammer, J., Dabbish, L.: Self-efficacy-based game design to encourage security behavior online. In: Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems. CHI EA '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3290607.3312935
15. Clark, J.M., Paivio, A.: Dual coding theory and education. Educational psychology review **3**(3), 149–210 (1991)
16. Club, D.P.: *Mini Metro*. Game [Windows] (November 2015), dinosaur Polo Club, Aotearoa, New Zeland.

17. Costin, A.: Security of cctv and video surveillance systems: Threats, vulnerabilities, attacks, and mitigations. In: Proceedings of the 6th International Workshop on Trustworthy Embedded Devices. p. 45–54. TrustED '16, Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2995289.2995290
18. Cuevas, H.M., Fiore, S.M., Oser, R.L.: Scaffolding cognitive and metacognitive processes in low verbal ability learners: Use of diagrams in computer-based training environments. Instructional Science **30**(6), 433–464 (Nov 2002). https://doi.org/10.1023/A:1020516301541
19. Culyba, S.: The Transformational Framework: A Process Tool for the Development of Transformational Games (9 2018). https://doi.org/10.1184/R1/7130594.v1
20. De Castell, S., Jenson, J.: Digital games for education: When meanings play. Intermédialités: Histoire et théorie des arts, des lettres et des techniques/Intermediality: History and Theory of the Arts, Literature and Technologies (9), 113–132 (2007)
21. Denning, T., Lerner, A., Shostack, A., Kohno, T.: Control-alt-hack: The design and evaluation of a card game for computer security awareness and education. In: Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security. p. 915–928. CCS '13, Association for Computing Machinery, New York, NY, USA (2013). https://doi.org/10.1145/2508859.2516753
22. Deterding, S., Dixon, D., Khaled, R., Nacke, L.: From game design elements to gamefulness: Defining "gamification". In: Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments. pp. 9–15. MindTrek '11, ACM, New York, NY, USA (2011). https://doi.org/10.1145/2181037.2181040
23. Dixon, M., Gamagedara Arachchilage, N.A., Nicholson, J.: Engaging users with educational games: The case of phishing. In: Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems. CHI EA '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3290607.3313026
24. Dreams, D.: *Metrico+*. Game [Windows] (August 2016), digital Dreams, Utrecht ,Netherlands
25. Fiorella, L., Vogel-Walcutt, J., Schatz, S.: Applying the modality principle to real-time feedback and the acquisition of higher-order cognitive skills. Educational Technology Research and Development **60**, 223–238 (04 2012). https://doi.org/10.1007/s11423-011-9218-1
26. Fu, K., Kohno, T., Lopresti, D., Mynatt, E., Nahrstedt, K., Patel, S., Richardson, D., Zorn, B.: Safety, security, and privacy threats posed by accelerating trends in the internet of things. Computing Community Consortium (CCC) Technical Report **29**(3) (2017)
27. Fuchs, M., Fizek, S., Ruffino, P., Schrape, N.: Rethinking gamification. meson press (2015)
28. Georgiev, T., Georgieva, E., Smrikarov, A.: M-learning: A new stage of e-learning. In: Proceedings of the 5th International Conference on Computer Systems and Technologies. pp. 1–5. CompSysTech '04, ACM, New York, NY, USA (2004). https://doi.org/10.1145/1050330.1050437
29. Giannakas, F., Kambourakis, G., Gritzalis, S.: Cyberaware: A mobile game-based app for cybersecurity education and awareness. In: 2015 International Conference on Interactive Mobile Communication Technologies and Learning (IMCL). pp. 54–58 (2015)

30. de Haan, Y., Kruikemeier, S., Lecheler, S., Smit, G., van der Nat, R.: When does an infographic say more than a thousand words? Journalism Studies **19**(9), 1293–1312 (2018). https://doi.org/10.1080/1461670X.2016.1267592
31. Heintz, S., Law, E.L.C.: Digital educational games: Methodologies for evaluating the impact of game type. ACM Trans. Comput.-Hum. Interact. **25**(2) (Apr 2018). https://doi.org/10.1145/3177881
32. Hendrix, M., Al-Sherbaz, A., Bloom, V.: Game based cyber security training: are serious games suitable for cyber security training? International Journal of Serious Games **3**(1) (2016)
33. Huang, W., Tan, C.L.: A system for understanding imaged infographics and its applications. In: Proceedings of the 2007 ACM Symposium on Document Engineering. p. 9–18. DocEng '07, Association for Computing Machinery, New York, NY, USA (2007). https://doi.org/10.1145/1284420.1284427
34. Hwang, G.J., Wu, P.H.: Advancements and trends in digital game-based learning research: a review of publications in selected journals from 2001 to 2010. British Journal of Educational Technology **43**(1), E6–E10 (2012). https://doi.org/10.1111/j.1467-8535.2011.01242.x
35. Johnson, C., Bailey, S., Buskirk, W.: Designing Effective Feedback Messages in Serious Games and Simulations: A Research Review, pp. 119–140 (11 2017). https://doi.org/10.1007/978-3-319-39298-1_7
36. Kaaz, K.J., Hoffer, A., Saeidi, M., Sarma, A., Bobba, R.B.: Understanding user perceptions of privacy, and configuration challenges in home automation. In: 2017 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC). pp. 297–301 (2017)
37. Kappen, D.L., Mirza-Babaei, P., Nacke, L.E.: Gamification through the application of motivational affordances for physical activity technology. In: Proceedings of the Annual Symposium on Computer-Human Interaction in Play. pp. 5–18. CHI PLAY '17, ACM, New York, NY, USA (2017). https://doi.org/10.1145/3116595.3116604
38. Karoui, A., Marfisi-Schottman, I., George, S.: A nested design approach for mobile learning games. In: Proceedings of the 16th World Conference on Mobile and Contextual Learning. mLearn 2017, Association for Computing Machinery, New York, NY, USA (2017). https://doi.org/10.1145/3136907.3136923
39. Kay, M., Terry, M.: Textured agreements: Re-envisioning electronic consent. In: Proceedings of the Sixth Symposium on Usable Privacy and Security. SOUPS '10, Association for Computing Machinery, New York, NY, USA (2010). https://doi.org/10.1145/1837110.1837127
40. Knijnenburg, B., Cherry, D.: Comics as a medium for privacy notices. In: Twelfth Symposium on Usable Privacy and Security (SOUPS 2016). USENIX Association, Denver, CO (Jun 2016), https://www.usenix.org/conference/soups2016/workshop-program/wfpn/presentation/knijnenburg
41. Lankow, J., Ritchie, J., Crooks, R.: Infographics: The power of visual storytelling. John Wiley & Sons (2012)
42. Levie, W.H., Lentz, R.: Effects of text illustrations: A review of research. Ectj **30**(4), 195–232 (1982)
43. Linehan, C., Kirman, B., Lawson, S., Chan, G.: Practical, appropriate, empirically-validated guidelines for designing educational games. In: Proceedings of the SIGCHI conference on human factors in computing systems. pp. 1979–1988 (2011)
44. Ling, Z., Luo, J., Xu, Y., Gao, C., Wu, K., Fu, X.: Security vulnerabilities of internet of things: A case study of the smart plug system. IEEE Internet of Things Journal **4**(6), 1899–1909 (2017)

45. Lyra, K.T., Isotani, S., Reis, R.C.D., Marques, L.B., Pedro, L.Z., Jaques, P.A., Bitencourt, I.I.: Infographics or graphics+text: Which material is best for robust learning? 2016 IEEE 16th International Conference on Advanced Learning Technologies (ICALT) (Jul 2016). https://doi.org/10.1109/icalt.2016.83
46. Mayer, R., Bove, W., Bryman, A., Mars, R., Tapangco, L.: When less is more: Meaningful learning from visual and verbal summaries of science textbook lessons. Journal of Educational Psychology **88**, 64–73 (03 1996). https://doi.org/10.1037/0022-0663.88.1.64
47. Mayer, R.E.: Multimedia Learning. Cambridge University Press, 2 edn. (2009). https://doi.org/10.1017/CBO9780511811678
48. McAuley, E., Duncan, T., Tammen, V.V.: Psychometric properties of the intrinsic motivation inventory in a competitive sport setting: A confirmatory factor analysis. Research quarterly for exercise and sport **60**(1), 48–58 (1989)
49. Molina, M.D., Gambino, A., Sundar, S.S.: Online privacy in public places: How do location, terms and conditions and vpn influence disclosure? In: Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems. CHI EA '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3290607.3312932
50. Moreno, R., Mayer, R.E.: Role of guidance, reflection, and interactivity in an agent-based multimedia game. Journal of educational psychology **97**(1), 117 (2005)
51. Morgner, P., Mattejat, S., Benenson, Z.: All your bulbs are belong to us: Investigating the current state of security in connected lighting systems. ArXiv **abs/1608.03732** (2016)
52. O'Rourke, E., Ballweber, C., Popovií, Z.: Hint systems may negatively impact performance in educational games. In: Proceedings of the First ACM Conference on Learning @ Scale Conference. p. 51–60. L@S '14, Association for Computing Machinery, New York, NY, USA (2014). https://doi.org/10.1145/2556325.2566248
53. Paivio, A.: Mental representations: A dual coding approach, vol. 9. Oxford University Press (1990)
54. Papastergiou, M.: Digital game-based learning in high school computer science education: Impact on educational effectiveness and student motivation. Computers & Education **52**(1), 1 – 12 (2009). https://doi.org/https://doi.org/10.1016/j.compedu.2008.06.004
55. Park, B., Flowerday, T., Brünken, R.: Cognitive and affective effects of seductive details in multimedia learning. Computers in Human Behavior **44**, 267 – 278 (2015). https://doi.org/https://doi.org/10.1016/j.chb.2014.10.061
56. Plass, J.L.: Handbook of Game-Based Learning. Mit Press (2020)
57. of Play Games, S.: *Lumino City*. Game [Windows] (December 2014), state of Play Games, London, United Kingdom
58. Schiefer, M.: Smart home definition and security threats. In: 2015 Ninth International Conference on IT Security Incident Management IT Forensics. pp. 114–118 (2015)
59. Serge, S.R., Priest, H.A., Durlach, P.J., Johnson, C.I.: The effects of static and adaptive performance feedback in game-based training. Computers in Human Behavior **29**(3), 1150 – 1158 (2013). https://doi.org/https://doi.org/10.1016/j.chb.2012.10.007
60. Sheng, S., Magnien, B., Kumaraguru, P., Acquisti, A., Cranor, L.F., Hong, J., Nunge, E.: Anti-phishing phil: the design and evaluation of a game that teaches people not to fall for phish. In: Proceedings of the 3rd symposium on Usable privacy and security. pp. 88–99 (2007)

61. Sheng, S., Magnien, B., Kumaraguru, P., Acquisti, A., Cranor, L.F., Hong, J., Nunge, E.: Anti-phishing phil: The design and evaluation of a game that teaches people not to fall for phish. In: Proceedings of the 3rd Symposium on Usable Privacy and Security. p. 88–99. SOUPS '07, Association for Computing Machinery, New York, NY, USA (2007). https://doi.org/10.1145/1280680.1280692
62. Shute, V.: Focus on formative feedback. Review of Educational Research **78**, 153–189 (03 2008). https://doi.org/10.3102/0034654307313795
63. Sivaraman, V., Chan, D., Earl, D., Boreli, R.: Smart-phones attacking smart-homes. In: Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks. p. 195–200. WiSec '16, Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2939918.2939925
64. Student: The probable error of a mean. Biometrika pp. 1–25 (1908)
65. ur Rehman, S., Gruhn, V.: An approach to secure smart homes in cyber-physical systems/internet-of-things. In: 2018 Fifth International Conference on Software Defined Systems (SDS). pp. 126–129 (2018)
66. Van Eck, R.: Building artificially intelligent learning games. In: Games and simulations in online learning: Research and development frameworks, pp. 271–307. IGI Global (2007)
67. Wen, Z.A., Lin, Z., Chen, R., Andersen, E.: What.hack: Engaging anti-phishing training through a role-playing phishing simulation game. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. CHI '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3290605.3300338
68. Zargham, N., Bahrini, M., Volkmar, G., Wenig, D., Sohr, K., Malaka, R.: What could go wrong? raising mobile privacy and security awareness through a decision-making game. In: Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts. p. 805–812. CHI PLAY '19 Extended Abstracts, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3341215.3356273
69. Zeng, E., Mare, S., Roesner, F.: End user security and privacy concerns with smart homes. In: Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017). pp. 65–80. USENIX Association, Santa Clara, CA (Jul 2017), https://www.usenix.org/conference/soups2017/technical-sessions/presentation/zeng
70. Zichermann, G., Cunningham, C.: Gamification by design: Implementing game mechanics in web and mobile apps. " O'Reilly Media, Inc." (2011)

Good vs. Evil: Investigating the Effect of Game Premise in a Smart Home Security Educational Game

Mehrdad Bahrini
Digital Media Lab
University of Bremen
Bremen, Germany
mbahrini@uni-bremen.de

Nima Zargham
Digital Media Lab
University of Bremen
Bremen, Germany
zargham@uni-bremen.de

Johannes Pfau
Digital Media Lab
University of Bremen
Bremen, Germany
jpfau@uni-bremen.de

Stella Lemke
Digital Media Lab
University of Bremen
Bremen, Germany
slemke@uni-bremen.de

Karsten Sohr
Digital Media Lab
University of Bremen
Bremen, Germany
sohr@tzi.de

Rainer Malaka
Digital Media Lab
University of Bremen
Bremen, Germany
malaka@tzi.de

ABSTRACT

While smart home devices are spreading rapidly, the privacy and security of users are key concerns. Many users struggle in acquiring and applying security recommendations to protect against malicious behavior in smart home systems, which can cause users to lose interest in this topic. Game-based learning is a powerful practice to increase the motivation of users in an entertaining and intuitive way. In this paper, we explore the effect of game premise on user's motivation and performance in an educational game. We designed a game with the aim to enlighten users about smart home security challenges. We developed two versions of the game with opposing game premises, a good and an evil, and compared them in a between-group experiment. The results show high motivation ratings in both versions of the game towards solving smart home security problems. However, there are no significant differences between the opposing game premises.

CCS CONCEPTS

• **Security and privacy → Usability in security and privacy**; • **Human-centered computing → Information visualization**.

KEYWORDS

Premise, Usable Security, Smart Home, Educational Games

ACM Reference Format:
Mehrdad Bahrini, Nima Zargham, Johannes Pfau, Stella Lemke, Karsten Sohr, and Rainer Malaka. 2020. Good vs. Evil: Investigating the Effect of Game Premise in a Smart Home Security Educational Game. In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '20 EA), November 2–4, 2020, Virtual Event, Canada.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3383668.3419887

CHI PLAY '20 EA, November 2–4, 2020, Virtual Event, Canada
© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-7587-0/20/11…$15.00
https://doi.org/10.1145/3383668.3419887

1 INTRODUCTION

With the advancements in technology and popularity of the internet, the number of internet-connected devices at home is growing rapidly [43]. These digital devices can communicate with each other and create a smart home environment by providing innovative and smart services to the users [2]. Researches have been looking at the concept of smart home since the late 1970s, but since then, the concept of home automation and the expectations of the user's from smart homes has changed drastically [12, 30]. Along with this rapid development, the security and privacy of users has always been a major concern. This issue can be partially solved by the improvements in the design of the security settings of smart home devices, nevertheless, that still leaves users with a great amount of responsibility to protect their sensitive data. Many users struggle to adopt and configure the security settings of their smart home devices due to the complexity of the topic. As users are overwhelmed with the daily updates and changes regarding privacy and security as well as the ever-expanding spread of security news and journals, it has become increasingly difficult for non-tech-savvy users to understand and apply security guidance.

Game-based learning has been a common method used to sustain motivation in learning for many decades [18, 25, 26, 31]. Educational games (edu-games) have the ability to help users learn about complex topics by using the entertaining nature of games which can serve as a powerful educational tool to motivate players. The learning process in edu-games is situated, where users can learn through experience and solve the problems in different stages through critical thinking. Games have long been recognized as an effective and appealing educational strategy to teach about various topics in the field of computer security and privacy [16, 23, 41]. In every digital game, there are multiple elements which provide context to the game. The premise of the game is one of these elements. In common parlance, premise is understood as "a statement or an idea that forms the basis for a reasonable line of argument" [37]. In the context of games, premise refers to the core of the story [20]. It is the main theme of the game which stays the same throughout the game. To date, several studies have investigated the effectiveness of digital game-based learning [3, 5, 8, 17, 42]. Nonetheless, research on the role of the game premise on player's motivation and learning are rather limited. The aim of this paper is to provide insights into the influence of good and evil game premises on users' motivation and learning procedure in an edu-game about smart home security. Using a between-subjects design, we attempt to answer the following research questions: *Can we measure a difference in motivation or learning progress between opposing game premises?*

Our preliminary results shows no significant differences between the good and the evil game premise.

2 RELATED WORK

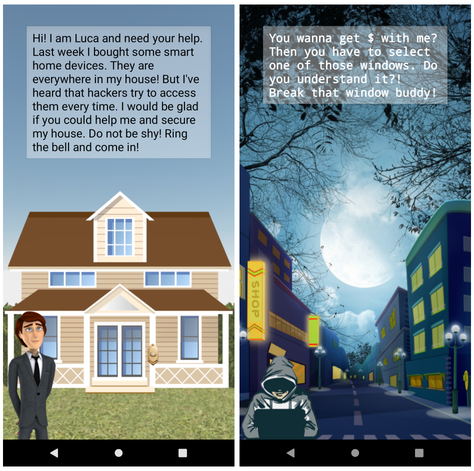
Previous research has investigated the privacy and security challenges of smart homes [10, 35, 46] and suggested possible recommendations to protect users [28, 38]. Such recommendations are most effective when considering the knowledge of the end-user about this topic. There are several studies on users' awareness regarding the risks of smart home devices [7, 32] which pointed out various misconceptions on the user side. These studies show that users are unaware of, and do not understand, many potential privacy threats and consequences. Previous research on smart home users also focused on usability issues such as installation, motivations and use cases, as well as control and automation interfaces [15, 44]. In our work, we selected smart home security content that is shared by most users. For example, incorrect security settings which can cause significant harm to user privacy because they do not have sufficient knowledge.

Game-based learning benefits from using interesting narratives and competitive exercises to motivate players to engage with specific learning targets [1]. Various studies have assessed the use of educational games to raise users' awareness and teach them about privacy and security [19, 33]. In our previous work [45], we designed a humorous decision-making game to help users better understand the consequences of applying security changes on a mobile device and found that our game-based approach was successful in engaging and raising user awareness. Sheng et al. [40] designed an online game to educate users to avoid phishing attacks. Their results showed that the game was effective in educating user's about phishing and other security attacks. Researchers also reviewed approaches to cybersecurity education [27, 29]. Compte et al. [34] analysed a number of serious games for information security and gave observations and suggestions for designing serious games in the context of cybersecurity education. Chen et al. [11] presented a desktop game to change users' cybersecurity behavior by translating self-efficacy into game design and found that users' confidence in tackling security issues was improved after playing the game.

Many researchers have explored the effectiveness of games for increasing cybersecurity awareness, however, they mostly focus primarily on the entertainment and engagement aspects of games [4, 27]. In a game-centered design, the formal elements of a game (objectives, procedures and mechanics) limit the actions of the players. Furthermore, games are also emotional experiences that challenge players to achieve their goals. Dramatic actions such as character, premise and story create a fascinating game experience [21]. Identifying the player with a character or an avatar can facilitate engagement and subsequent learning through games [13]. Moreover, Fullerton [20] defines that premise is a way to create engagement and gives context to the formal game elements. Many games would be too abstract without a dramatic premise for the players to be emotionally integrated into their outcome [20]. Our approach contains two different versions of premise, namely good and evil. The players could either be a good person and save the smart home systems or an evil person trying to learn how to hack them. Grudpan et al. [24] explored the effect of game premise on player motivation and engagement with the game. Their results showed that the game premise significantly influenced the players' intrinsic motivation and their engagement with the game. In our work we conduct a comparative study to find out the influence of two opposing game premises on the motivation and performance of players.

3 GAME DESIGN

The game was developed for mobile platforms and includes two separated, opposing narratives called *Save My Home* and *Hacker War* (see Figure 1). By playing *Save My Home*, players meet the character Luca in front of his home, who is worried about the security of his smart home devices. He asks the player to help him by searching devices and answering related questions. This version serves as the good game premise. Contrary to that, the player encounters an anonymous hacker on the street in *Hacker War*. He asks the player to help him intrude into one of the neighbors' houses and earns money. This game version serves as the evil premise. We have considered two characters with opposing attitudes. In order to enrich the environmental effect, we adjusted the background music and sound effects respective to the chosen premise. In both versions, same game procedure and mechanics were used.



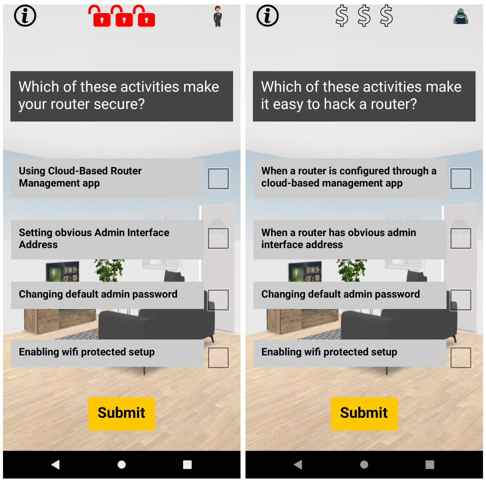
Hi! I am Luca and need your help. Last week I bought some smart home devices. They are everywhere in my house! But I've heard that hackers try to access them every time. I would be glad if you could help me and secure my house. Do not be shy! Ring the bell and come in!

You wanna get $ with me? Then you have to select one of those windows. Do you understand it?! Break that window buddy!

Figure 1: The game consists of two opposing premises, the good *Save My Home* (left) and the evil *Hacker War* (right).

3.1 Procedure

The game aims to introduce an innovative way of tackling recent security issues of smart home devices, by allowing users to see themselves in a simulated smart home environment. It consists primarily of the following building blocks:



Which of these activities make your router secure?

Using Cloud-Based Router Management app □

Setting obvious Admin Interface Address □

Changing default admin password □

Enabling wifi protected setup □

Submit

Which of these activities make it easy to hack a router?

When a router is configured through a cloud-based management app □

When a router has obvious admin interface address □

Changing default admin password □

Enabling wifi protected setup □

Submit

Figure 2: Question screen: *Save My Home* (left) and *Hacker War* (right).

- Finding devices: The game contains five rooms, each consisting of two smart devices. Players have to tap on these devices and answer their corresponding question.
- Request help: Users can use the supporting knowledge provided in the game in order to get more insights about the context of the smart device and related security issues.
- Feedback of answers: After the player submits an answer, the game displays a meaningful message. If the answer was wrong, the player will be notified about the correct answer. Eventually, the player is awarded based on the number of correct answers at the end of the game.

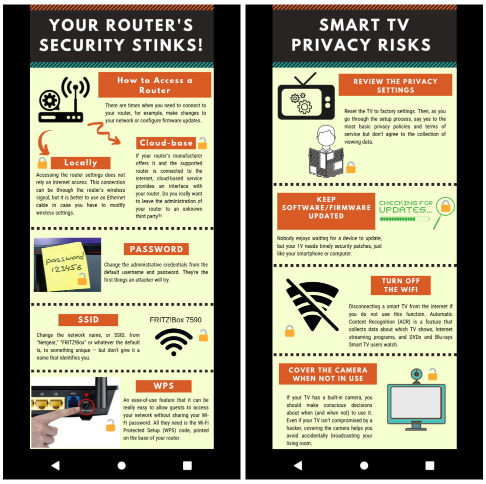
3.1.1 Introduction. After starting a game session, the respective avatar (home owner or anonymous hacker) will be displayed, expressing his goal via a textual speech bubble (see Figure 1). The avatar further explains the game mechanism to the player. By tapping on the door bell in *Save My Home* or touching a window in *Hacker War*, the player starts the game.

3.1.2 Progression. Independent from the narrative, each play-through consists of ten questions. Within *Save My Home*, Luca's home will become more secure, proportional to the number of correct answers. In order to transfer this concept to the player, three open red locks are displayed, which are changed to green closed locks after three correct answers respectively. Regarding *Hacker War*, the same set of questions is asked, only differing in the phrasing, since they stem from the hacker. After each three correct answers, the player gets a golden dollar sign.

3.1.3 Question. By tapping on each smart device, the question screen will be shown. All questions are multiple choice. On top of the question screen, the player can find two buttons: The help button on the left displays background information about the device's security, while the avatar icon on the right explains general game controls (see Figure 2). Within each room, the player is directed to proceed to the next room after answering two questions.

3.1.4 Request help. By tapping on the information icon, the player is directed to the related help screen which includes infographics. Various symbols have been used to transfer the concepts to the players and to increase their attention. E.g., the unlock icon symbolizes insecure action or meaning. This concept was applied to all infographics (see Figure 3).

3.1.5 Completion. At the end of the game, the player is directed to the scoring interface. The number of correct answers and the corresponding reward are displayed to the player.



YOUR ROUTER'S SECURITY STINKS!

How to Access a Router

There are times when you need to connect to your router, for example, make changes to your network or configure firmware updates.

Locally

Accessing the router settings does not rely on Internet access. This connection can be through the router's wireless signal, but it is better to use an Ethernet cable in case you have to modify wireless settings.

Cloud-base

If your router's manufacturer offers it and the supported router is connected to the internet, cloud-based service provides an interface with your router. Do you really want to leave the administration of your router to an unknown third party?!

PASSWORD

Change the administrative credentials from the default username and password. They're the first things an attacker will try.

SSID

Change the network name, or SSID, from "Netgear," "FRITZ!Box" or whatever the default is, to something unique – but don't give it a name that identifies you.

FRITZ!Box 7590

WPS

An ease-of-use feature that it can be really easy to allow guests to access your network without sharing your Wi-Fi password. All they need is the Wi-Fi Protected Setup (WPS) code, printed on the base of your router.

SMART TV PRIVACY RISKS

REVIEW THE PRIVACY SETTINGS

Reset the TV to factory settings. Then, as you go through the setup process, say yes to the most basic privacy policies and terms of service but don't agree to the collection of viewing data.

KEEP SOFTWARE/FIRMWARE UPDATED

CHECKING FOR UPDATES...

Nobody enjoys waiting for a device to update, but your TV needs timely security patches, just like your smartphone or computer.

TURN OFF THE WIFI

Disconnecting a smart TV from the internet if you do not use this function. Automatic Content Recognition (ACR) is a feature that collects data about which TV shows, Internet streaming programs, and DVDs and Blu-rays Smart TV users watch.

COVER THE CAMERA WHEN NOT IN USE

If your TV has a built-in camera, you should make conscious decisions about when (and when not) to use it. Even if your TV isn't compromised by a hacker, covering the camera helps you avoid accidentally broadcasting your living room.

Figure 3: Help Screen: Router (left) and Smart TV (right).

4 EVALUATION

We evaluated the effect of the opposite game premises with ($n = 30$) participants, equally distributed into two groups, in a between-subjects lab study. The study sessions were held on the university campus, with one participant per session and a duration of 30 to 45 minutes. As a mobile device, we provided a Google Pixel 2 XL.

Before starting the game session, the interviewer provided an introduction about the game and security problems about smart home devices. The participants then played the game and answered all the questions. Play time was measured by the interviewer. After the game, the player answered a number of questionnaires. The first questionnaire contained general questions regarding demographic information and experience with smart home devices. To measure the usability of the game, the second questionnaire consisted of the System Usability Scale (SUS) [9]. Motivation of the player was measured by utilizing the Intrinsic Motivation Inventory (IMI) [36]. The IMI has been used in research focused on intrinsic motivation and self-regulation in diverse fields such as computer activities and training and focuses on different dimensions like *Interest-Enjoyment*, *Perceived Competence* and *Effort-Importance*.

4.1 Participants

The study consisted of 30 participants, where 14 players had a college degree, and 16 completed high school. Among the subjects, 15 people identified themselves as male and 15 as female. In terms of age, participants ranged between 21 and 44 years with an average age of 30.6 (*SD* = 6.38).

4.2 Results

When asked about their smart home device usages, we found that all of our participants had at least one smart home device and all owned a smart phone. From the standardized questionnaires, we learn that both game versions, *Save My Home* and *Hacker War* can result in comparably good usability and motivation ratings. Regarding usability, *Save My Home* has a mean SUS score of 84.2 (*SD* = 8.99) and *Hacker War* of 87.5 (*SD* = 4.90). This indicates an above average usability score for both versions with no significant difference (see Table 1).

In both groups, the IMI Questionnaire shows high ratings for all the sub-scales. For *Save My Home*, IMI score of *Interest-Enjoyment* was rated 5.84 (*SD* = 0.76), *Perceived Competence* score was rated 5.67 (*SD* = 0.4) and *Effort-Importance* score was rated 5.3 (*SD* = 0.70). For *Hacker War*, the sub-scale *Interest-Enjoyment* was rated 6.08 (*SD* = 0.51), *Perceived Competence* score was rated 5.91 (*SD* = 0.34) and *Effort-Importance* score was rated 5.8 (*SD* = 0.92). The data show no significant differences between the two game versions on any of the sub-scales or aggregated scores ($p > .05$) (see Table 1).

We measured the players' correct answers for each session to evaluate performance of the players. Players in *Save My Home* had an average of 6.9 correct answers per session (*SD* = 0.13). Similarly, in *Hacker War*, players' average of correct answers was 7.1 (*SD* = 0.17). This data also did not show significant differences between the two versions ($p > .05$). The play time for all the players was also calculated. The average play time for *Save My Home* was 14.6 minutes (*SD* = 3.38), and for *Hacker War* 14.93 minutes (*SD* = 2.42) which show no significant differences in play time.

Table 1: Independent Samples T-Test

	t	df	p	Cohen's d
SUS	1.260	28.000	0.218	0.460
Correct Answers	-0.597	28.000	0.555	-0.218
Play Time	0.311	28.000	0.758	0.114
IMI-Enjoyment	1.029	28.000	0.312	0.376
IMI-Competence	1.798	28.000	0.083	0.657
IMI-Importance	1.641	28.000	0.112	0.599

5 DISCUSSION AND LIMITATIONS

In this study, we set out to understand the impact of opposing game premises on user's motivation and performance by comparing a good game premise with an evil one. We used two standard questionnaires (SUS and IMI) to evaluate the game's usability and players' motivation to compare our two conditions. We found no evidence for a difference between the opposing premises, regarding their similar SUS and IMI scores. This finding suggests that a good or an evil game premise do not differ from one another in terms of influencing players' performance and motivation. Results from the user study indicate that both versions of the game have distinct usability and participants enjoyed playing them. Regardless of the premise of the game, the participants' interest in playing the game was high and both versions of the game were successful in motivating users to learn about the security of smart homes. In line with the previous literature, participants found our game a useful and motivating tool to learn about security recommendations [22, 39]. The results of the SUS showed high usability ratings [6] for the two versions. Nonetheless, no significant differences were found. This might be attributed to the game mechanics and procedure which were the same in both versions. The resulting IMI (*Effort-Importance*) scores show that the players were eager to play the game and made an effort to answer the questions in both game versions. Users' rating for all the three sub-scales of the IMI Questionnaire were positive and supporting. Still, no significant differences was identified between the two versions in any of these sub-scales. The good and evil game premises did not differ in influencing user's performance. In both versions, participants correctly answered approximately on average 70% of the time. Regarding the play time, participants in both groups on average spent a similar amount of time to run through the game. The time experience is not only tied to the game time relation and to the challenges provided by the game, but also to the relation between game difficulty and player ability. By looking at the calculated playing times and the number of correct answers, it became apparent that the game challenges matched the skills of the players. According to the flow framework [14], if the challenges match the player's abilities, the player will better enjoy the game.

Although the good and the evil premises had a clear distinction, in both versions, the player has the role of a helpful person. This feeling of helpfulness might convey a "good" premise despite the game version. Moreover, the game premise was distributed at random among the participants. While the results suggest that both versions are equally playable, different premise versions are likely to have varying effects on different player types. Using the insights gathered from this initial pilot study, we are looking forward to conduct a fully-fledged field study that investigates the relation of player type and adaptive game premise within the context of usable security education. Further, we think it would be helpful to use an iterative design process in order to make sure that players connect with the narrative [11].

6 CONCLUSION

In this paper, we investigated the effect of game premise on user's motivation and performance in an educational game. We presented two versions of an edu-game with opposing game premises, a good and an evil, to inform users about smart home security and motivate them to learn about this topic. Our results showed high usability and motivation ratings for both game versions. However, we found no significant differences between the opposing game premises. Participants enjoyed playing the game and found both versions useful and motivating. Our findings could provide useful insights for researchers, game designers and security experts to consider the influence of game premise and its motivational aspects.

ACKNOWLEDGMENTS

This work was supported by the German Federal Ministry of Education and Research (BMBF) under the grant 16SV8503 (UsableSec@ Home project).

REFERENCES

[1] Wilfried Admiraal, Jantina Huizenga, Sanne Akkerman, and Geert Dam. 2011. The concept of flow in collaborative game-based learning. *Computers in Human Behavior* 27 (05 2011), 1185–1194. https://doi.org/10.1016/j.chb.2010.12.013

[2] Waqar Ali, Ghulam Dustgeer, Muhammad Awais, and Munam Ali Shah. 2017. IoT based smart home: Security challenges, security requirements and solutions. In *2017 23rd International Conference on Automation and Computing (ICAC)*. IEEE, 1–6.

[3] Anissa All, Elena Patricia Nuñez Castellar, and Jan Van Looy. 2016. Assessing the effectiveness of digital game-based learning: Best practices. *Computers & Education* 92 (2016), 90–103.

[4] Faisal Alotaibi, Steven Furnell, Ingo Stengel, and Maria Papadaki. 2016. A Review of Using Gaming Technology for Cyber-Security Awareness. *International Journal for Information Security Research* 6 (06 2016). https://doi.org/10.20533/ijisr.2042.4639.2016.0076

[5] Nalin Asanka Gamagedara Arachchilage, Steve Love, and Konstantin Beznosov. 2016. Phishing threat avoidance behaviour: An empirical investigation. *Computers in Human Behavior* 60 (2016), 185–197.

[6] Aaron Bangor, Philip T. Kortum, and James T. Miller. 2008. An Empirical Evaluation of the System Usability Scale. *International Journal of Human–Computer Interaction* 24, 6 (2008), 574–594. https://doi.org/10.1080/10447310802205776 arXiv:https://doi.org/10.1080/10447310802205776

[7] Xavier Bellekens, Preetila Seeam, Quentin Franssen, Andrew Hamilton, Kamila Nieradzinska, and Amar Seeam. 2016. Pervasive eHealth services a security and privacy risk awareness survey. In *International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA), London,*. IEEE. https://doi.org/10.1109/CyberSA.2016.7503293

[8] Francesco Bellotti, Bill Kapralos, Kiju Lee, and Pablo Moreno-Ger. 2013. User assessment in serious games and technology-enhanced learning.

[9] John Brooke. 2013. SUS: a retrospective. *Journal of usability studies* 8, 2 (2013), 29–40.

[10] Joseph Bugeja, Andreas Jacobsson, and Paul Davidsson. 2016. On Privacy and Security Challenges in Smart Connected Homes. 172–175. https://doi.org/10.1109/EISIC.2016.044

[11] Tianying Chen, Jessica Hammer, and Laura Dabbish. 2019. Self-Efficacy-Based Game Design to Encourage Security Behavior Online. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI EA '19)*. Association for Computing Machinery, New York, NY, USA, Article LBW1610, 6 pages. https://doi.org/10.1145/3290607.3312935

[12] Sudhir Chitnis, Neha Deshpande, Arvind Shaligram, et al. 2016. An investigative study for smart home security: Issues, challenges and countermeasures. *Wireless Sensor Network* 8, 04 (2016), 61.

[13] Jonathan Cohen. 2001. Defining Identification: A Theoretical Look at the Identification of Audiences With Media Characters. *Mass Communication and Society* 4, 3 (2001), 245–264. https://doi.org/10.1207/S15327825MCS0403_01 arXiv:https://doi.org/10.1207/S15327825MCS0403_01

[14] Mihaly Csikszentmihalyi and Mihaly Csikzentmihaly. 1990. *Flow: The psychology of optimal experience*. Vol. 1990. Harper & Row New York.

[15] A. Demeure, S. Caffiau, E. Elias, and C. Roux. 2015. Building and Using Home Automation Systems: A Field Study. In *End-User Development*, Paloma Díaz, Volkmar Pipek, Carmelo Ardito, Carlos Jensen, Ignacio Aedo, and Alexander Boden (Eds.). Springer International Publishing, Cham, 125–140.

[16] Tamara Denning, Adam Lerner, Adam Shostack, and Tadayoshi Kohno. 2013. Control-Alt-Hack: The Design and Evaluation of a Card Game for Computer Security Awareness and Education. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security* (Berlin, Germany) *(CCS '13)*. Association for Computing Machinery, New York, NY, USA, 915–928. https://doi.org/10.1145/2508859.2516753

[17] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. 2011. From Game Design Elements to Gamefulness: Defining "Gamification". In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments* (Tampere, Finland) *(MindTrek '11)*. ACM, New York, NY, USA, 9–15. https://doi.org/10.1145/2181037.2181040

[18] Matt Dixon, Nalin Asanka Gamagedara Arachchilage, and James Nicholson. 2019. Engaging Users with Educational Games: The Case of Phishing. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI EA '19)*. Association for Computing Machinery, New York, NY, USA, Article LBW0265, 6 pages. https://doi.org/10.1145/3290607.3313026

[19] Mathias Fuchs, Sonia Fizek, Paolo Ruffino, and Niklas Schrape. 2015. *Rethinking gamification.* meson press.

[20] T. Fullerton. 2019. *Game Design Workshop: A Playcentric Approach to Creating Innovative Games.* CRC Press, Taylor & Francis Group.

[21] Tracy Fullerton, Jenova Chen, Kellee Santiago, Erik Nelson, Vincent Diamante, Aaron Meyers, Glenn Song, and John Deweese. 2006. That cloud game: dreaming (and doing) innovative game design. (07 2006), 51–59. https://doi.org/10.1145/1183316.1183324

[22] James Paul Gee. 2003. What Video Games Have to Teach Us about Learning and Literacy. *Comput. Entertain.* 1, 1 (Oct. 2003), 20. https://doi.org/10.1145/950566.950595

[23] F. Giannakas, G. Kambourakis, and S. Gritzalis. 2015. CyberAware: A mobile game-based app for cybersecurity education and awareness. In *2015 International Conference on Interactive Mobile Communication Technologies and Learning (IMCL)*. 54–58.

[24] Supara Grudpan, Dmitry Alexandrovky, Jannicke Baalsrud Hauge, and Rainer Malaka. 2019. Exploring the Effect of Game Premise in Cooperative Digital Board Games. In *Joint International Conference on Entertainment Computing and Serious Games*. Springer, 214–227.

[25] Juho Hamari, David J Shernoff, Elizabeth Rowe, Brianno Coller, Jodi Asbell-Clarke, and Teon Edwards. 2016. Challenging games help students learn: An empirical study on engagement, flow and immersion in game-based learning. *Computers in human behavior* 54 (2016), 170–179.

[26] Stephanie Heintz and Effie L.-C. Law. 2018. Digital Educational Games: Methodologies for Evaluating the Impact of Game Type. *ACM Trans. Comput.-Hum. Interact.* 25, 2, Article 8 (April 2018), 47 pages. https://doi.org/10.1145/3177881

[27] Maurice Hendrix, Ali Al-Sherbaz, and Victoria Bloom. 2016. Game Based Cyber Security Training: are Serious Games suitable for cyber security training? *International Journal of Serious Games* 3, 1 (Mar. 2016). https://doi.org/10.17083/ijsg.v3i1.107

[28] Andreas Jacobsson and Paul Davidsson. 2015. Towards a model of privacy and security for smart homes. 727–732. https://doi.org/10.1109/WF-IoT.2015.7389144

[29] Ge Jin, Manghui Tu, Tae-Hoon Kim, Justin Heffron, and Jonathan White. 2018. Game Based Cybersecurity Training for High School Students. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education* (Baltimore, Maryland, USA) *(SIGCSE '18)*. Association for Computing Machinery, New York, NY, USA, 68–73. https://doi.org/10.1145/3159450.3159591

[30] Arun Cyril Jose and Reza Malekian. 2017. Improving smart home security: Integrating logical sensing into smart home. *IEEE Sensors Journal* 17, 13 (2017), 4269–4286.

[31] Aous Karoui, Iza Marfisi-Schottman, and Sébastien George. 2017. A Nested Design Approach for Mobile Learning Games. In *Proceedings of the 16th World Conference on Mobile and Contextual Learning* (Larnaca, Cyprus) *(mLearn 2017)*. Association for Computing Machinery, New York, NY, USA, Article 4, 4 pages. https://doi.org/10.1145/3136907.3136923

[32] Predrag Klasnja, Sunny Consolvo, Jaeyeon Jung, Benjamin M. Greenstein, Louis LeGrand, Pauline Powledge, and David Wetherall. 2009. "When I Am on Wi-Fi, I Am Fearless": Privacy Concerns & Practices in Eeryday Wi-Fi Use. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. ACM, New York, NY, USA, 1993–2002. https://doi.org/10.1145/1518701.1519004

[33] Aron Laszka, Mark Felegyhazi, and Levente Buttyan. 2014. A Survey of Interdependent Information Security Games. *ACM Comput. Surv.* 47, 2, Article 23 (Aug. 2014), 38 pages. https://doi.org/10.1145/2635673

[34] Alexis Le Compte, David Elizondo, and Tim Watson. 2015. A renewed approach to serious games for cyber security. 203–216. https://doi.org/10.1109/CYCON.2015.7158478

[35] Huichen Lin and Neil Bergmann. 2016. IoT privacy and security challenges for smart home environments. *Information* 7, 3 (2016), 44. https://doi.org/10.3390/info7030044

[36] Edward McAuley, Terry Duncan, and Vance V Tammen. 1989. Psychometric properties of the Intrinsic Motivation Inventory in a competitive sport setting: A confirmatory factor analysis. *Research quarterly for exercise and sport* 60, 1 (1989), 48–58.

[37] Oxford. 2020. Definition of premise. https://www.oxfordlearnersdictionaries.com/definition/english/premise?q=premise.

[38] Charith Perera, Ciaran McCormick, Arosha K. Bandara, Blaine A. Price, and Bashar Nuseibeh. 2016. Privacy-by-Design Framework for Assessing Internet of Things Applications and Platforms. In *Proceedings of the 6th International Conference on the Internet of Things* (Stuttgart, Germany) *(IoT'16)*. ACM, New York, NY, USA, 83–92. https://doi.org/10.1145/2991561.2991566

[39] Gerhard Schwabe and Christoph Göth. 2005. Mobile learning with a mobile game: design and motivational effects. *Journal of Computer Assisted Learning* 21, 3 (2005), 204–216. https://doi.org/10.1111/j.1365-2729.2005.00128.x arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1365-2729.2005.00128.x

[40] Steve Sheng, Bryant Magnien, Ponnurangam Kumaraguru, Alessandro Acquisti, Lorrie Cranor, Jason Hong, and Elizabeth Nunge. 2007. Anti-Phishing Phil: The design and evaluation of a game that teaches people not to fall for phish. *ACM International Conference Proceeding Series* 229, 88–99. https://doi.org/10.1145/1280680.1280692

[41] Steve Sheng, Bryant Magnien, Ponnurangam Kumaraguru, Alessandro Acquisti, Lorrie Faith Cranor, Jason Hong, and Elizabeth Nunge. 2007. Anti-Phishing Phil: The Design and Evaluation of a Game That Teaches People Not to Fall for Phish. In *Proceedings of the 3rd Symposium on Usable Privacy and Security* (Pittsburgh, Pennsylvania, USA) *(SOUPS '07)*. Association for Computing Machinery, New York, NY, USA, 88–99. https://doi.org/10.1145/1280680.1280692

[42] Steve Sheng, Bryant Magnien, Ponnurangam Kumaraguru, Alessandro Acquisti, Lorrie Faith Cranor, Jason Hong, and Elizabeth Nunge. 2007. Anti-phishing phil: the design and evaluation of a game that teaches people not to fall for phish. In *Proceedings of the 3rd symposium on Usable privacy and security.* 88–99.

[43] Statista. 2020. Number of Smart Homes forecast worldwide from 2017 to 2024. https://www.statista.com/forecasts/887613/number-of-smart-homes-in-the-smart-home-market-worldwide.

[44] Jong-bum Woo and Youn-kyung Lim. 2015. User Experience in Do-it-yourself-style Smart Homes. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Osaka, Japan) *(UbiComp '15)*. ACM, New York, NY, USA, 779–790. https://doi.org/10.1145/2750858.2806063

[45] Nima Zargham, Mehrdad Bahrini, Georg Volkmar, Dirk Wenig, Karsten Sohr, and Rainer Malaka. 2019. What Could Go Wrong? Raising Mobile Privacy and Security Awareness Through a Decision-Making Game. In *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts* (Barcelona, Spain) *(CHI PLAY '19 Extended Abstracts)*. Association for Computing Machinery, New York, NY, USA, 805–812. https://doi.org/10.1145/3341215.3356273

[46] K. Zhang, J. Ni, K. Yang, X. Liang, J. Ren, and X. S. Shen. 2017. Security and Privacy in Smart City Applications: Challenges and Solutions. *IEEE Communications Magazine* 55, 1 (January 2017), 122–129. https://doi.org/10.1109/MCOM.2017.1600267CM

Handle With Care: Exploring Recognition Error Handling Methodologies for Speech-Based Systems

NIMA ZARGHAM, Digital Media Lab, University of Bremen, Germany
JOHANNES PFAU, Digital Media Lab, University of Bremen, Germany
TOBIAS SCHNACKENBERG, University of Bremen, Germany
RAINER MALAKA, Digital Media Lab, University of Bremen, Germany

Advances in speech recognition, language processing and natural interaction have led to an increased industrial and academic interest. While the robustness, vocabulary, adaptability and usability of such systems are steadily increasing, speech-based systems will remain susceptible to recognition errors. This is commonly due to the vastly noisy input format, consistently exposed to varying hardware quality, background noise and the large spectrum of voice characteristics – making intelligent error handling of utmost importance for the success of those systems. In this work, we evaluate ($N = 34$) the user experience of optimal error handling (given that optimal decisions can be found) versus traditional, repetition-based error handling, situated in a voice-controlled video game. Our results indicate that implementing error handling can improve the usability of a system, if it follows the intention of the user. Otherwise, it impairs the user experience, even when deciding for technically optimal decisions.

CCS Concepts: • **Human-centered computing** → **Natural language interfaces**; • **Applied computing** → **Computer games**.

Additional Key Words and Phrases: Voice User Interfaces; Game Design; Error Handling; Speech-Based Systems; Voice-Controlled Game; Voice Interaction

ACM Reference Format:
Nima Zargham, Johannes Pfau, Tobias Schnackenberg, and Rainer Malaka. 2018. Handle With Care: Exploring Recognition Error Handling Methodologies for Speech-Based Systems. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Voice user interfaces (VUIs) are gaining more and more attention in recent years due to the intuitive nature of their interaction. Speaking is a natural way of communication amongst humans and people find it easier to interact with technology that resembles some of their own characteristics [10]. Voice input is now a feature in many devices such as mobile phones, cars, and home assistants. In their early days, VUIs were designed for handling few specialized tasks [47], but due to the advancements in the technology, they now can have a broad range of capabilities in performing various functions in different settings. Current VUIs are used for various purposes such as smart home control, scheduling, navigation, education, and entertainment. The technical aspects of the VUIs, as well as their usability and user experience (UX), have been the subject of extensive research in the recent years [15, 17, 19, 35, 36].

Manuscript submitted to ACM

In order to integrate speech recognition, developers need to have a large repository of collected voice data so that the system has enough information to process different inflections and variations in different voices. If the product is aimed at the global market, different languages, accents, and dialects need to be considered to assure a better recognition system. On top of that, different forms of phrasing for a single command should be incorporated to allow for a more natural experience, underlining the issue that designing a satisfying experience with speech-based systems is a complex and difficult process.

Although this technology is steadily improving in various aspects, speech-based systems are still prone to recognition failures. Several elements such as hardware limitations, background noises and language barriers make designing voice interfaces a very complex and time-consuming task. Researchers believe that problems with speech recognition and limited functionality are the main reasons for disliking or not using voice systems [26]. Users have frequently reported that they find voice interaction disappointing or embarrassing, which lets such systems appear as unintelligent and immature [4, 12, 26, 33, 40]. This makes error handling a critical part of designing VUIs, not limited to situations in which the system does not understand the user's command, the given command is out of context, or the command is misunderstood [30]. Several different guidelines for designing fallback strategies have been proposed, such as asking the user to repeat the command, redirecting the user to the tasks that the system can support, or presenting user options to correct their commands [8, 30, 43]. In some cases, the voice assistant (VA) falls back on humor in response to complex conversational input and commands that cannot be handled otherwise which might be seen as sarcastic or entertaining [48].

Recently, this technology has gained considerable attention in the entertainment industry and video game companies have been adopting voice-activated services to their games. As speech recognition technology is improving rapidly and the number of available microphones in consumer gaming devices is growing everyday, it leaves a great potential for using VUIs in games [1]. This allows voice-control to be used as an appealing and intuitive feature in video games to enhance the experience of the players. Speaking is a natural and enjoyable way of interacting which can increase social presence within the game and make them more immersive. With the release of Microsoft Kinect in 2010, Xbox games in various genres such as *Mass Effect 3* [23], *FIFA 14* [22], *Forza Motorsport 5* [54], and *Ryse: Son of Rome* [20] took advantage of the voice interaction that was provided by Kinect. However, in most cases, voice input is an optional feature and not a core element of the game design. Voice-activated games attempted to provide natural language input, but this experience has been frequently described as "uncomfortable" and "awkward" by players [21].

Speech-controlled video games and especially error handling have been under-investigated in academic research [4, 14]. On this basis, we designed a voice-controlled video game with the aim of investigating user experience with two different error handling methodologies. In this game, players control the game protagonist using voice commands. A between-subjects user study was conducted to compare optimized error handling with traditional repetition-based error handling within the game. In the control group, the game would notify the player of the recognition failure so that the player could repeat the command once again. With optimized error handling, if a command was not recognized, the game would perform a locally optimized action in respect to goal completion and obstacle avoidance without notifying the player about the recognition failure.

In this study, we pursue the following two research questions:

RQ1 Does implementing optimized error handling lead to a measurably improved usability in speech-based video games?

RQ2 What are the effects on user experience if error handling mechanisms decide for unintended actions?

Based on our design space and the existing literature, we developed the following hypotheses:

- *H1: Participants will observe a lower number of recognition errors in case of optimized error handling.*
- *H2: The optimized error handling is preferred by participants.*

Our results showed significantly higher usability ratings for the optimized error handling, as well as a significantly lower number of perceived errors for this condition. Furthermore, this study contributes useful insights and implications on the user experience with recognition error handling in speech-based systems, most importantly the users' aversion to error handling that opposed their intention – even in cases of goal-directed and optimized solutions.

2 RELATED WORK

Since the early success of voice and gesture in an interface with the "Put that There" system [9], voice user interfaces have been largely investigated by researchers in the field of HCI. In this section, we provide a summary of the previous literature on speech-based systems, voice interaction in video games, and complication with VUIs.

2.1 Research on Speech-Based Systems

Developing speech-based systems requires techniques, methodologies, and development tools that are capable of flexible and adaptive interaction, bearing in mind the need of different user groups and different environments [55]. In recent years, natural language processing (NLP) has become much more sophisticated and reliable [13]. Apart from technical development, interaction research tackled multitudes of novel voice interfaces, investigating how people use these devices and how they respond to different kinds of speech from computers [4, 7, 17, 32].

Speech-based systems have been evaluated for various purposes and professional fields. In the medical domain, Austerjost et al. presented a VUI for controlling laboratory instruments [5], while Miehle et al. presented a concept for voice assistants (VAs) as a support in surgical operating rooms [38]. Zargham and Bonfert et al. [60] investigated voice interaction in a single-player VR game where they compared a version of the game in which the players could talk to multiple characters using natural language to a version where they verbally interact with a single character. The study showed that the participants preferred conversing with a group of interlocutors, found it more entertaining and felt like being part of a team. Another prominent application area resides in teaching. Jung et al. [27] developed a voice-controlled educational game to teach children computer programming, concluding that their game led children to be more immersed in the game and understand the elements of programming with ease and confidence. Winkler et al. [58] compared groups who either used a human or a VA tutor when solving a problem. Their results indicated that groups interacting with VA showed significantly higher task outcomes and higher degrees of collaboration quality compared to groups interacting with human tutors.

Although the functionality and ease of use of VUIs are frequently researched and enhanced, research suggests that the reliability of these systems is not more important than their attractiveness [59]. In a study by Lopatovska et al. [31], the authors explored user interactions with the popular VA Amazon Alexa. They report that people were still satisfied with the system even when Alexa did not produce desirable outcomes. Authors suggest that the UX might be more important to the users than the quality of the output.

One particular challenge with VUIs is that it can lead to unrealistic expectations from the system's intelligence, what it can do, and how well it can keep a natural and fluid conversation [35]. Users tend to test the capabilities of VUIs by asking different questions and in many cases, their expectations tend to exceed the agent's capabilities [32, 34]. In a study by Lovato et al. on children's experience with Siri, authors found that children predominantly ask Siri personal questions, to get to know the agent and test its potential [32]. When users initial expectations from such systems are not met, it can lead to disappointment and a generally negative experience [47].

Overall, a great deal of the design research is focused on narrow application areas and specific interface components. This in turn leads to the lack of more generalizable design guidelines [16]. In our work, we seek to advance the state of the art by exploring methodologies of recognition error handling.

2.2 Voice Interaction in Video Games

The intuitive nature of voice user interfaces allowed them to become an increasing trend, not only in an assisting function within smart homes, phones or cars, but also for the advancement of mechanics within the entertainment industry. Although the rate of VUI studies has increased in recent years, research on voice interaction in games – those where voice control has a fundamental role in the game – is rather limited [14]. Using alternative means of interaction for games such as voice can not only expand the possibility space for novel in-game mechanics, but can also be especially important for users with disabilities, where traditional controls are not feasible [57]. Other human modalities can be combined together with speech to optimize players' performance and overcome the drawbacks of using only speech [42]. Nonetheless, there are still essential aspects and questions regarding voice interaction in games that have gone largely unexplored [4].

Voice interaction in video games is rather distinct from the other contexts. Research shows that in-game voice commands are associated with a sense of taking on a character in the game's world [3]. Allison et al. suggest that voice interactions which creates a conflict with the social world can impede the player's engagement with the in-game world [4]. Early research on voice interaction in digital games roots back to the 1970s, where *VoiceChess*, a game which could support standardized chess instructions using a speech recognition system, was developed [1, 51]. Since then, numerous video game titles have embraced the use of voice as input. In a successive study by Allison et al., the authors surveyed 449 video games and 22 audio games in which players use their voice to affect the game state [2]. They observed that academic research has focused on a narrow subset of design patterns, especially pronunciation, and suggest game designers to consider non-verbal forms, which have proven to provide enjoyable game experiences with fast and discrete input possibilities [24, 44, 53, 56]. A number of studies have investigated non-verbal voice input to control the game using volume and pitch [44, 53, 56]. In essence, this modality can offer faster and discrete input compared to a speech-based interaction [24]. Some popular video games that have used non-verbal forms of voice interaction are mobile games such as *Scream Go Hero* [29], *Rock Band* [25], and *Chicken Scream* [46], which use volume and vocal pitch as an input to control the game's character.

Although there are plenty of examples of video games that use speech-based voice interaction, those which use non-verbal forms of voice input have been more successful. The reason behind the success of such games is that they avoid recognition errors entirely [2, 3]. However, due to the limited controls, these games are usually restricted to relatively simple mechanics. In this work, we simulate an environment that enables fast and reliable calculation of technically optimized actions so that gaps in recognition can be handled and the resulting experience investigated.

2.3 Complications with Voice Interaction

A large portion of research about voice interaction is concerned with speech recognition and its accuracy rates [2]. These systems are commonly trained with a large sample of voice data, connected with ontologies and knowledge graphs, in order to identify and understand users' commands and respond with a reasonable and satisfying answer [30]. Nevertheless, the given commands by the users can be fuzzy, personal, and complicated, resulting in the system not being able to understand them, which is not likely to be overcome by soft- or hardware advancements in recognition alone. To conquer the difficulties inherent in processing the commands, users usually need to put more effort on formulating the command so that it is recognized by the system. When interacting with a VUI, users often speak differently than they would speak to a human. Many expect natural language not to be understood by such systems and adapt special communication strategies therefore. Reducing the talking pace, re-formulating command sentences and physically relocating themselves and/or the system are popular observable patterns when users are confronted with recognition errors [26]. Jentsch et al. observed that users took a considerable amount of time to formulate their prompts before commanding them to a VUI [26]. In their study, authors also witnessed that even when the users are not instructed to use keywords, they are still likely to restrict themselves to a set of words or commands when addressing a speech assistant. This has led users to refrain from speech-based systems to perform difficult tasks. In a study by Lugar et al. [34], authors interviewed frequent users of conversational agents and found that the study participants did not trust the system to do complex tasks – like writing emails or making phone calls – down to an apprehension that the system would not get the task done correctly. Authors also note that the interaction with the agent was generally considered as a secondary task.

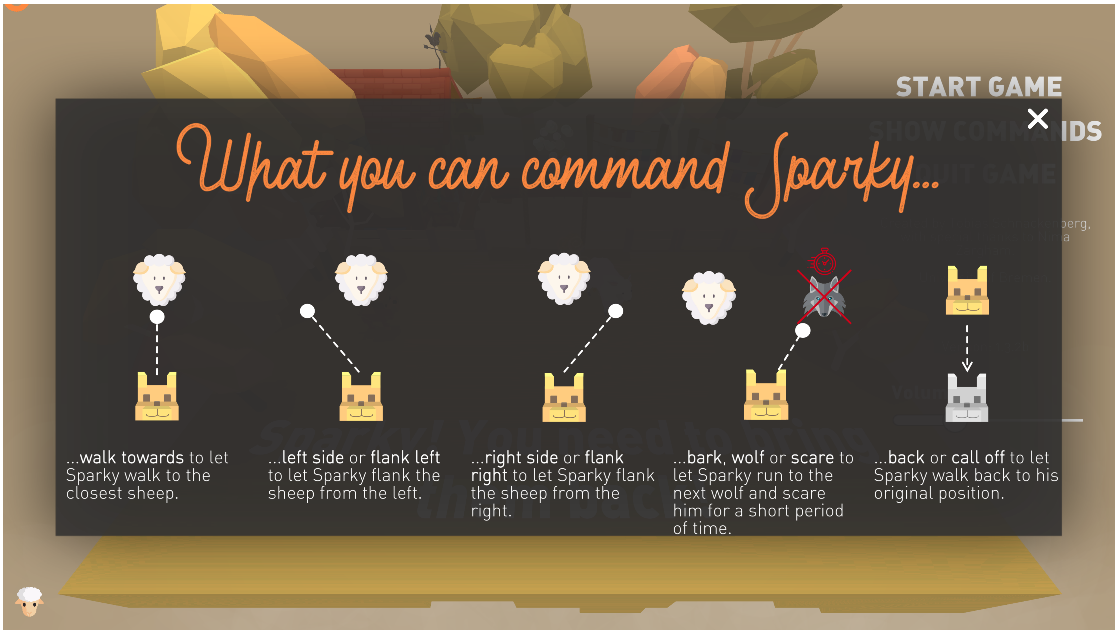
On the other hand, when errors occur, the system should give an appropriate response. In her book about designing VUIs, Cathy Pearl suggests that, if the error handling is done well, it will not derail users and you can get them back on the track and have them successfully complete a task [45]. If it's done poorly, not only the user will fail to complete a task, but they actually might refuse to use the system again. Although the technical aspects of VUIs have been largely investigated, researchers agree on the stance that the user side of speech interaction is relatively less explored [6, 16, 39, 41]. Above that, language barriers pose a further common problem with VUIs. A study by Pyae et al. showed that VUIs are easier to use, friendlier and potentially more useful for native English speakers than non-native speakers [50]. The complex and expensive process of implementing a reliable speech-based system, impels researchers in this field to often use a Wizard of Oz approach [28, 37].

Eventually, technical limits, unnatural assumptions, and lack of faith in the system's technological capabilities still make up the major reasons for users' reservations against using VUIs. In our approach, we focus on overcoming innate technical limits of speech recognition with optimized error handling and examine the impact of this intervention on the perceived intelligence, appraisal and usability of the system.

3 PROTOTYPE DESIGN

To evaluate our hypotheses, we designed and implemented "Listen, Sparky!", a speech-controlled arcade game. In this game, players are in control of the sheepdog "Sparky" who has to guide a sheep throughout restricted courses and keep away hazardous encounters. Using speech-controlled commands, players impersonate a shepherd that gives directions to his sheepdog. The game consists of eight levels. In every level, players have to safely navigate and return the sheep that escaped from a meadow, up to a designated goal location (gate).

The first four levels of the prototype served as a tutorial. In these, players were taught about the game controls and the commands to use. Every level would introduce one new command to the players with the exception of the fourth level that would introduce two commands. The participants were able to access an overview of the available commands at any time in the game menu (see Figure 1). After going through the first two levels, a hostile wolf character was introduced that threatened the survival of the escorted sheep. If the sheep would get too close to the wolf, the level failed and had to be restarted. With increasing progression of the levels, the challenge of the game would similarly increase. For instance, in the early levels, the wolf is standing still and does not move and the player has to simply



What you can command Sparky...

...walk towards to let Sparky walk to the closest sheep.

...left side or flank left to let Sparky flank the sheep from the left.

...right side or flank right to let Sparky flank the sheep from the right.

...bark, wolf or scare to let Sparky run to the next wolf and scare him for a short period of time.

...back or call off to let Sparky walk back to his original position.

Fig. 1. Voice commands making up the core game controls, assessable anytime during gameplay.

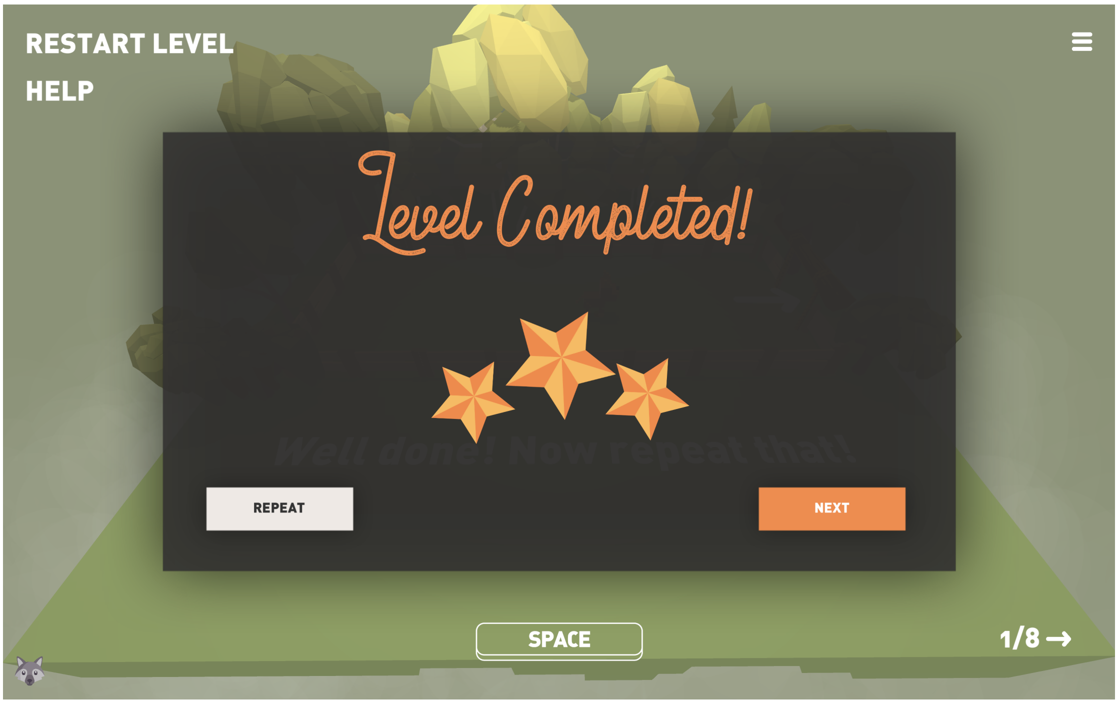
avoid those areas of the game. In higher levels, the wolf would start moving or even chase the sheep to make the game more demanding for the player and enforce quick acting. At the end of each level, the game would display a screen indicating that the level was successfully completed while presenting performance feedback throughout a classic star rating system (see Figure 2).

In order to start the speech recognition and have Sparky listen to the commands, players had to press and hold the spacebar. As long as the space bar was pressed, the default computer microphone was used to record the players' voice. If the space bar was released too fast, the system would not process that command.

While holding the space bar, the player's voice input was recorded, processed and (if possible) interpreted as one of the following actions:

- "Walk towards": Sparky walks straight towards the sheep, navigating the sheep to the same direction.
- "Flank Left": Sparky flanks the sheep from the left side, navigating the sheep to the right side (relative to the fixed view angle of the participant).
- "Flank Right": Sparky flanks the sheep from the right side, navigating the sheep to the left side.
- "Back": Sparky goes back to the position where it began the level.
- "Bark at wolf": Sparky moves towards the wolf and barks. This results in paralyzing the wolf for some seconds and making it harmless to the sheep.

For every action, we trained the system to accept multiple phrases to perform that action. For instance, if players wanted to command Sparky to "flank right", they could also use phrases such as "go right!", "right side" or "move right". If a command was recognized by the voice recognition system, Sparky would execute the corresponding command. If no



Level Completed!

Well done! Now repeat that!

Fig. 2. Feedback screen: After completing each level, the players would receive a star rating based on their performance.

matching command was found, the system would consider that as a failed attempt. In such cases, the game would refer to the error handling system based on the respective experimental group. In order to evaluate different error handling methods, we needed to ensure that at least a few instances of recognition failure would occur. To achieve this, both game versions were programmed to have a minimum overall error occurrence of 15% after the first 10 commands. This means, if a player managed to get lower than the target error rate, the next request was intentionally misrecognized by the system.

The environment of the game and the game logic have been built with Unity 3D[1]. For speech recognition, the Google Cloud Speech-To-Text service[2] was used. The requests were directly sent to the Google services. We chose this service as it does not require any native library to run and makes the prototype compatible to any available platform. We created builds for Windows, Mac OS and Linux.

4 EVALUATION

4.1 Study Design

We conducted a between-subjects design user study with (N = 34) participants to compare and evaluate our two conditions. In the control group, 17 participants played a version that employed traditional error handling, i.e. in the case of non-recognition, the character would not react but only indicate that the command was not recognized by

[1] https://unity3d.com/unity
[2] https://cloud.google.com/speech-to-text

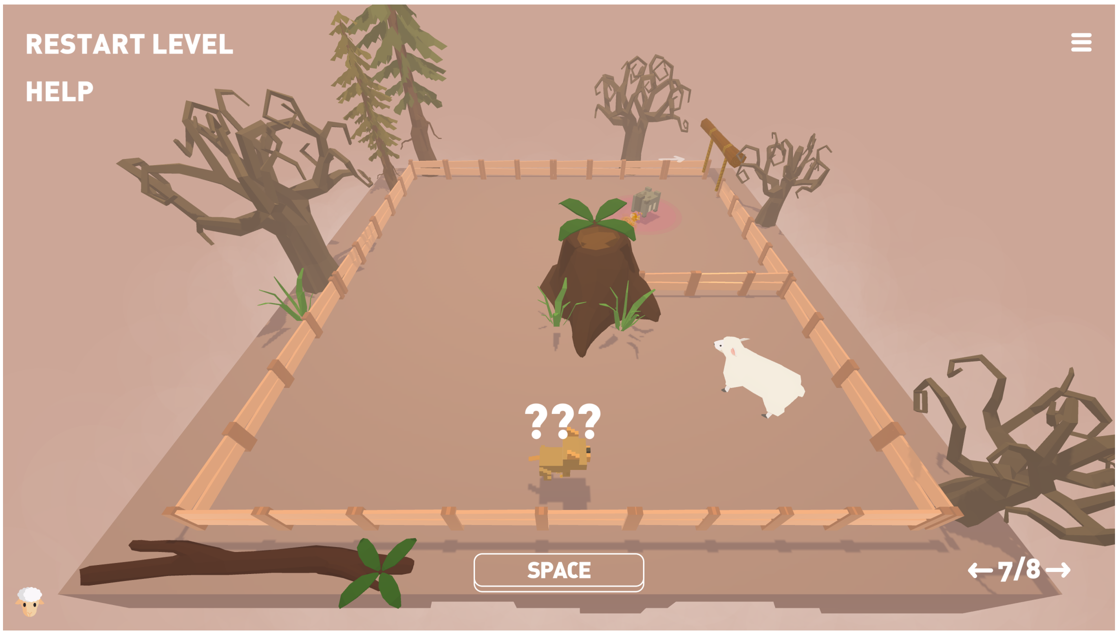


Fig. 3. In the control group, when a command is not recognized, the game displays question marks over Sparky's head.

displaying some question marks above its head (see Figure 3). In the intervention group, 17 players which were mutually excluded from the first group, played a version that implemented optimized error handling, based on the underlying game state. In effect, if a command was not recognized, the game would perform a locally optimized action regarding goal completion and obstacle avoidance without letting him/her know that the recognition failed.

Among both conditions, levels, game environment, and mechanics remained equal, leaving the error handling method as the single manipulated variable. Group assignment was pseudo-randomized between two equally distributed groups.

Participants were asked to play all eight levels of "Listen, Sparky" – yet, if they became stuck on a specific level after multiple tries, they were allowed to skip it. The execution took place on the subjects' own PC or laptop device. Before every session, the experimenter made sure that every player had a functional microphone to use for the game.

4.2 Procedure

Every experimental session was held remotely via video calls, where the experimenter recorded verbal statements and in-game observations while providing assistance in cases of complicacy. Before starting the session, participants were briefly informed about the experiment procedure. Although the game contained an explanatory tutorial, the interview conductor would shortly explain the game and the controls. After the participants gave informed consent, they would share their screen with the experiment conductor. Participants would then play through the game in either one of the two conditions. After finishing the game, participants completed the post-exposure questionnaires. At the end of the session, we held a short semi-structured interview with each participant. Each session took approximately 40 – 50 minutes with an average of 18.4 minutes game-play time (SD = 5.16).

4.3 Measures

In order to evaluate our hypotheses and to understand how players experience the error handling in both conditions, we used standardized questionnaires to assess the player experience and the perceived usability of the system. Our post-exposure questionnaires included demographic questions, the System Usability Scale (SUS) [11], as well as the Player Experience of Need Satisfaction (PENS) [52] throughout the subscales of *Competency*, *Autonomy*, *Relatedness*, *Presence/immersion*, and *Intuitive controls*. Additionally, we recorded a series of customized questions regarding their experience with the game. These were executed via 5-point Likert scales and concerned the extent with which Sparky behaved as the participant expected him to do so, Sparky's perceived intelligence and the overall experience with the game. Above that, players were asked to estimate the approximate number of commands that were not recognized, and to explicate what Sparky did when the commands were not recognized by the system. We concluded the session with a brief, semi-structured interview to further evaluate qualitative aspects of player experience, usability, and individual preferences for both conditions. The experiment and interview were recorded acoustically and transcribed for later analysis.

4.4 Participants

A quota sampling approach was used to recruit participants for this study in which the selection was based on mailing lists, social networks, word-of-mouth and gaming forums. Participation was voluntary and uncompensated. ($N = 34$) people participated in the experiment (10 self-identified as female, 24 as male), between 21 and 43 years of age ($M = 28.68$, $SD = 5.24$). 85% of our participants had previous experience with voice assistants (18 rarely, 11 often). Only 17% of the participants have previously played a voice-controlled video game. We conducted the experiment in English with international participants, thus, the sample also consisted of non-native speakers.

5 RESULTS

In order to identify possible differences between both conditions, we applied statistical significance tests as well as qualitative content analysis towards our issued research questions.

For the comparison of each sub-scale in PENS, we ran an unpaired Student's t-test, with an alpha level of .05, where missing values were imputed. In our study, we focused on the four sub-scales of *Competency*, *Autonomy*, *Presence/Immersion*, and *Intuitive Controls* (cf. Figure 4). Consequential, we found a significant effect for *Intuitive Controls* in favor for the intervention group ($M = 5.96$, $SD = 1.29$), compared to the control group ($M = 4.7$, $SD = 1.98$), $T(32) = 2.184$, $p = .036$, 95% CI, displaying a medium effect ($d_{Cohen} = 0.75$) [18]. In contrast, *Competency*, *Autonomy*, and *Presence/Immersion* did not show significant differences between the two conditions ($p > .05$).

Regarding usability, SUS scores reached an average of 63.23 ($SD = 20.47$) within the control group, whereas the intervention group resulted in 80.88 ($SD = 8.96$). The subsequent Student's t-test indicates that optimized error handling outperformed the control group significantly in terms of usability ($T(32) = 3.254$, $p = .0027$, 95% CI, cf. Figure 4), revealing a large effect between conditions ($d_{Cohen} = 1.572$).

For the overall game experience, players of the control group rated it as 3.411 ($SD = 1.18$) on average, not significantly different from the intervention group (M =3.889, $SD = 0.93$; $T(32) = 1.295$, $p = .2047$, 95% CI). Assessing to what extent Sparky followed the users expectations, no significant differences between the control ($M = 3.12$, $SD = 0.99$) and intervention group ($M = 3.24$, $SD = 0.90$) could be found ($T(32) = 0.361$, $p = .720$, 95% CI). Similarly, no significant

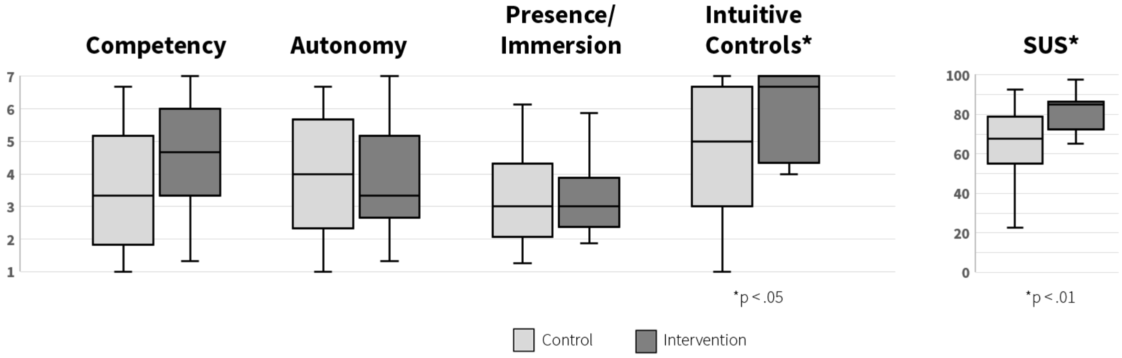


Fig. 4. Boxplot indicating significant results from PENS-subscales and SUS between control and intervention group. Includes median (–), standard deviation (box) and range (whiskers).

effect on Sparky's perceived intelligence emerged ($T(32) = 0.323$, $p = .748$, 95% CI), with an average of 3.06 ($SD = 0.97$) under the control condition, and 2.94 ($SD = 1.14$) within the intervention group.

Overall, the participants in the intervention group had a mean error rate of 42.94% ($SD = 18.41$), while the control group resulted in 33.53% ($SD = 11.31$) errors on average. However, when participants were asked to write down the approximate number of commands that were not recognized by the system, the mean number of perceived errors in the control group resulted in 34.863 ($SD = 36.882$) which is significantly higher ($T(32) = 3.0491$, $p = .0048$, 95% CI) than that of the intervention group ($M = 6.438$, $SD = 5.501$), revealing a large effect ($d_{Cohen} = 1.078$).

We also asked the participants to explain Sparky's behavior in cases where commands were not correctly recognized by the system. In the intervention group, 59% believed he did something wrong, 23% said he did something random, 12% said it always understood the commands, and 6% thought, he helped to perform the right action. Among the participants under the control condition, 76% said Sparky did not react when the command was not recognized, 12% said he did something wrong, one participant (6%) said he did something random and another stated that the commands were always recognized.

5.1 Qualitative Results

Interpreting the post-exposure interview sessions, qualitative insights could be extracted with respect to the different error handling methodologies and the overall game experience itself. The recordings were systematically examined using qualitative content analysis, following a coding scheme construed by a first selection of interviews samples. Subsequently, all recordings were analyzed, coded along this categorization, and summarized. Additionally, we highlighted insightful and unique statements.

Participants generally enjoyed playing the game and attributed it as entertaining. In both groups, players liked the idea of playing a speech-based video game in general. Several players mentioned that they especially liked the game's aesthetics. In both groups, participants mentioned that they got better at controlling Sparky after some playing time. One participant stated: "I felt that I learned how to speak for the game to understand me". However, many believed that with their improvements, the game's challenges also got more complex. Several players stated that they enjoyed the progressive enhancement of the game's difficulty. Many players perceived the game's controls as intuitive. The recognition system was trained to accept different styles of commands in the same context that were likely to be given by participants, and thus not limited to the particular commands from the tutorial. One of the participants stated: "The commands were intuitive. I did not use exactly the game's commands and it still worked. I liked that". On the other hand, some players wished for less restrictions regarding the commands for the game controls. One player said "I'd expect all the normal replacement phrases to work as well". Some participants demanded more controls, e.g. one participant stated that "it would be nice to have a command that repeats the previous one". One participant said that single-word commands would be better for such games. Others believed that using phrases felt more natural and interesting. Two participants (both none-native English speakers) mentioned that it would have been nice if the system could learn their voice and accent.

More than half of the participants in the intervention group mentioned that they sometimes found the behaviour of Sparky unexpected. Only one player in the control condition mentioned something similar. Both groups equally reported the disliking of the occurrence of voice recognition malfunctioning, as well as the delay between the command and execution. During the interviews, we revealed both conditions and their difference in error handling to the participants. Four of them mentioned that they would prefer to have optimized error handling as an optional feature that they could activate in the game's settings. One participant stated "When the recognition is not working, that means there is a problem. If I don't see the errors, I don't see the problem. So I think the errors should be seen to acknowledge the problem and improve the recognition". Multiple participants of the intervention group shared the opinion that they like that the game's flow is not being disturbed by recognition errors. One of them stated: "I really like the idea of this game since it does not disturb the flow when there is a problem with the recognition technology". One participant said "I would prefer that the game performs an action randomly. That way, it makes the game more exciting and challenging". Another participant mentioned that speech-based games such as this game could be an interesting medium to teach foreign languages to children.

6 DISCUSSION

This evaluation aimed at exploring the impact of recognition error handling techniques on the user experience by contrasting traditional to optimized handling within a speech-controlled video game. Overall, users' feedback about "Listen, Sparky!" were rather positive and supporting. Players in both conditions generally enjoyed playing our voice-controlled game. Many wanted to continue playing even after the experiment. During the experiment, some participants asked for repeating the levels even after successfully finishing that level. Additionally, we observed that players often perceived time pressure, leading to more complications with command recognition. This was mainly due to the change in the talking pace and fast decisions, which in times led to unclear and incorrect inquiries. We also recorded a higher error rate for non-native speakers. This led to more frustration for these players during the game, aligning with the results of the study by Pyae et al [50].

Participants improved in understanding how the recognition system works after spending some time in the game. They learned how to formulate their commands and to speak clearly in order to be recognized by the system. Additionally, they also developed their ability to play the game by adopting the game mechanics over the various levels.

Eventually, we interpreted the results of this experiment to provide answers to the following comprehensive questions:

RQ1: Does implementing optimized error handling lead to a measurably improved usability in speech-based video games?
RQ2: What are the effects on user experience if error handling mechanisms decide for unintended actions?

Regarding **RQ1**, results indicate a significantly higher usability, as well as higher ratings of intuitive control for the version employing optimized error handling. Yet, qualitative statements underline that this increase of usability is mainly due to the cases where the error handling actually followed the user's intention, which was not always the case, even when deciding for the technically optimized solution. In cases of mismatch, participants perceived it as a different kind of error, even if the performed action was the technically optimized choice. As soon as doubts to the system were raised, this even impacted the learning curve of the users. Thus, we argue that error handling can improve the user experience of speech-based games, though the major objective of the handling technique should not approximate technically optimized decisions, but individually tailored predictions. Supplementary to the usability analysis, quantitative findings of the recorded error observations confirm the former results: Although participants of the intervention group committed more errors on average, they in fact reported a significantly lower amount of perceived errors, compared to the control group. In effect, we approve our first hypothesis:
H1: Participants will observe a lower number of recognition errors in case of optimized error handling.

Concerning **RQ2**, we observed differences between both groups and interpreted users' reactions and responses to error handling that conflicted their original intention. Players of the intervention group were repeatedly confused by Sparky acting against their original intention, resulting in a misleading learning experience that impaired in-game progress and proficiency attainment. Since the control group was not affected by automatically handled actions, this issue did only occur in the former condition. Even if quantitative insights suggest a higher usability through the optimized error handling intervention, qualitative statements reflect the dissatisfaction in situations where the handling deviates from the user's intention. Above that, since correctly handled errors were not perceived as errors in the first place, participants rated the intervention version as not more intelligent than the without handling. Consequently, we reject *H2*, as no concordant results for a preferred version could be found.

Based on the interpretation of the results regarding both research questions, we conclude with the following implications: Error handling can significantly improve the usability of a speech-controlled video game and aid in bridging the technological gap of speech recognition. Yet, ideal error handling should model (and predict) the individual user's intention, be equipped with an internal likelihood estimation whether the handled decision is appropriate or follow similar methods to ensure user satisfaction. Otherwise, false handling can impair both the experience as well as the learning progress and raise doubts about error handling in general.

6.1 Limitations and Future Work

While the findings of this study present significant steps forward in exploring recognition error handling methodologies in speech-based games, there are still some limitations that should be addressed. In this work, we investigated optimized error handling in a speech-based video game. Although the broader insights of this evaluation can apply to the use and error handling of VUIs in general, in future work, these methods could be transferred and evaluated in other domains such as navigation, medicine, education, and smart homes, to explore conversationally more complex settings. During the experiment, we noticed that some participants had difficulties learning the game controls and game mechanics. For future studies, we recommend longer tutorials as well as gaming sessions to counter influences of individual learning rate. Apart from this, differences in player types and players' current emotional and social states could lead to different experiences, which should be incorporated and reflected in further studies. The implemented voice recognition system for the game has not been trained with data from non-native English speakers, yet the majority of participants fell under this condition. The recognition with those who spoke a strong accent was therefore not optimal and could have been improved by training the system differently. Although our game controls were limited to a predefined set of commands, this helped us to have a structured procedure with high comparability [49]. In order to yield scalable insights for broader application fields and cover large command vocabularies, future studies will expand the scope of the potential actions. Furthermore, user responses in our study indicated the desire to have different error handling methods as an optional feature, which will be addressed in upcoming work.

7 CONCLUSION

In this paper, we investigated an optimized error handling for a speech recognition system and explored its potentials and challenges. We designed a voice-controlled video game called "Listen, Sparky!" to evaluate our concept. In a between-subjects design study, we compared our optimized error handling model to a traditional repetition-based version. Our results showed that implementing error handling can improve the usability of a system, if it follows the intention of the user. Otherwise, it can impair the user experience, even when deciding for technically optimized decisions. Ideal error handling should therefore model the individual user's intention, be equipped with an internal likelihood estimation whether the handled decision is appropriate or follow similar methods to ensure user satisfaction. Our findings contribute useful insights for researchers and developers on how to address, display and handle recognition errors in speech-based video games and the greater application field of voice user interfaces.

ACKNOWLEDGMENTS

[Blank for Review]

REFERENCES

[1] Fraser Allison, Marcus Carter, and Martin Gibbs. 2017. Word Play: A History of Voice Interaction in Digital Games. *Games and Culture* 15, 2 (2017), 91 – 113. https://doi.org/10.1177/1555412017746305

[2] Fraser Allison, Marcus Carter, Martin Gibbs, and Wally Smith. 2018. Design Patterns for Voice Interaction in Games. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play* (Melbourne, VIC, Australia) *(CHI PLAY '18)*. Association for Computing Machinery, New York, NY, USA, 5–17. https://doi.org/10.1145/3242671.3242712

[3] Fraser Allison, Joshua Newn, Wally Smith, Marcus Carter, and Martin Gibbs. 2019. Frame Analysis of Voice Interaction Gameplay. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3290605.3300623

[4] Fraser John Allison. 2020. *Voice interaction game design and gameplay.* Ph.D. Dissertation.

[5] Jonas Austerjost, Marc Porr, Noah Riedel, Dominik Geier, Thomas Becker, Thomas Scheper, Daniel Marquard, Patrick Lindner, and Sascha Beutel. 2018. Introducing a virtual assistant to the lab: A voice user interface for the intuitive control of laboratory instruments. *SLAS TECHNOLOGY: Translating Life Sciences Innovation* 23, 5 (2018), 476–482.

[6] Matthew P Aylett, Per Ola Kristensson, Steve Whittaker, and Yolanda Vazquez-Alvarez. 2014. None of a CHInd: relationship counselling for HCI and speech technology. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems*. 749–760.

[7] Frank Bentley, Chris Luvogt, Max Silverman, Rushani Wirasinghe, Brooke White, and Danielle Lottridge. 2018. Understanding the Long-Term Use of Smart Speaker Assistants. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 3, Article 91 (Sept. 2018), 24 pages. https://doi.org/10.1145/3264901

[8] Dan Bohus and Alexander I Rudnicky. 2005. Constructing accurate beliefs in spoken dialog systems. In *IEEE Workshop on Automatic Speech Recognition and Understanding, 2005.* IEEE, 272–277.

[9] Richard A Bolt. 1980. "Put-that-there" Voice and gesture at the graphics interface. In *Proceedings of the 7th annual conference on Computer graphics and interactive techniques*. 262–270.

[10] Cynthia Breazeal. 2003. Toward sociable robots. *Robotics and autonomous systems* 42, 3-4 (2003), 167–175.

[11] John Brooke et al. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996), 4–7.

[12] Raluca Budiu and Page Laubheimer. 2018. Intelligent assistants have poor usability: A user study of Alexa, Google assistant, and Siri. *Nielsen Norman Group. Available online at https://www. nngroup. com/articles/intelligentassistant-usability/(last accessed 4/12/2019)* (2018).

[13] Erik Cambria and Bebo White. 2014. Jumping NLP curves: A review of natural language processing research. *IEEE Computational intelligence magazine* 9, 2 (2014), 48–57.

[14] Marcus Carter, Fraser Allison, John Downs, and Martin Gibbs. 2015. Player Identity Dissonance and Voice Interaction in Games. In *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play* (London, United Kingdom) *(CHI PLAY '15)*. Association for Computing Machinery, New York, NY, USA, 265–269. https://doi.org/10.1145/2793107.2793144

[15] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 4960–4964.

[16] Leigh Clark, Philip Doyle, Diego Garaialde, Emer Gilmartin, Stephan Schlögl, Jens Edlund, Matthew Aylett, João Cabral, Cosmin Munteanu, Justin Edwards, et al. 2019. The State of Speech in HCI: Trends, Themes and Challenges. *Interacting with Computers* 31, 4 (2019), 349–371.

[17] Leigh Clark, Nadia Pantidi, Orla Cooney, Philip Doyle, Diego Garaialde, Justin Edwards, Brendan Spillane, Emer Gilmartin, Christine Murad, Cosmin Munteanu, and et al. 2019. What Makes a Good Conversation? Challenges in Designing Truly Conversational Agents. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, Article 475, 12 pages. https://doi.org/10.1145/3290605.3300705

[18] Jacob Cohen. 1988. *Statistical Power Analysis for the Behavioral Sciences* (2nd ed.). Erlbaum, Hillsdale, NJ.

[19] Erica Cooper, Alison Chang, Yocheved Levitan, and Julia Hirschberg. 2016. Data Selection and Adaptation for Naturalness in HMM-Based Speech Synthesis.. In *INTERSPEECH*. 357–361.

[20] Crytek. 2013. *Ryse: Son of Rome*. Game [XBox One]. Microsoft Studios, Redmond, Washington, U.S.

[21] Steven Dow, Manish Mehta, Ellie Harmon, Blair MacIntyre, and Michael Mateas. 2007. Presence and engagement in an interactive drama. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 1475–1484.

[22] EA Sports. 2013. *Fifa 14*. Game [XBox One]. Microsoft Studios, Redwood City, California, U.S.

[23] Electronic Arts. 2012. *Mass Effect 3*. Game [XBox 360]. Electronic Arts, Redwood City, California, U.S.

[24] Susumu Harada, Jacob O Wobbrock, and James A Landay. 2011. Voice games: investigation into the use of non-speech voice input for making computer games more accessible. In *IFIP Conference on Human-Computer Interaction*. Springer, 11–29.

[25] Harmonix and Pi Studios. 2007. *Rock Band*. Game [XBox 360].

[26] Martin Jentsch, Maresa Biermann, and Evelyn Schweiger. 2019. Talking to Stupid?!? Improving Voice User Interfaces. *Mensch und Computer 2019-Usability Professionals* (2019).

[27] Hyunhoon Jung, Hee Jae Kim, Seongeun So, Jinjoong Kim, and Changhoon Oh. 2019. TurtleTalk: an educational programming game for children with voice user interface. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–6.

[28] John F Kelley. 1983. An empirical methodology for writing user-friendly natural language computer applications. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. 193–196.

[29] Ketchapp. [n.d.]. *Scream Go Hero*. Game [Android and IOS].

[30] Toby Jia-Jun Li, Igor Labutov, Brad A Myers, Amos Azaria, Alexander I Rudnicky, and Tom M Mitchell. 2018. An end user development approach for failure handling in goal-oriented conversational agents. *Studies in Conversational UX Design* (2018).

[31] Irene Lopatovska, Katrina Rink, Ian Knight, Kieran Raines, Kevin Cosenza, Harriet Williams, Perachya Sorsche, David Hirsch, Qi Li, and Adrianna Martinez. 2019. Talk to me: Exploring user interactions with the Amazon Alexa. *Journal of Librarianship and Information Science* 51, 4 (2019), 984–997. https://doi.org/10.1177/0961000618759414 arXiv:https://doi.org/10.1177/0961000618759414

[32] Silvia Lovato and Anne Marie Piper. 2015. "Siri, is This You?": Understanding Young Children's Interactions with Voice Input Systems. In *Proceedings of the 14th International Conference on Interaction Design and Children* (Boston, Massachusetts) *(IDC '15)*. ACM, New York, NY, USA, 335–338. https://doi.org/10.1145/2771839.2771910

[33] Ewa Luger and Abigail Sellen. 2016. " Like Having a Really Bad PA" The Gulf between User Expectation and Experience of Conversational Agents. In *Proceedings of the 2016 CHI conference on human factors in computing systems*. 5286–5297.

[34] Ewa Luger and Abigail Sellen. 2016. "Like Having a Really Bad PA": The Gulf Between User Expectation and Experience of Conversational Agents. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 5286–5297. https://doi.org/10.1145/2858036.2858288

[35] Michal Luria, Samantha Reig, Xiang Zhi Tan, Aaron Steinfeld, Jodi Forlizzi, and John Zimmerman. 2019. Re-Embodiment and Co-Embodiment: Exploration of Social Presence for Robots and Conversational Agents. In *Proceedings of the 2019 on Designing Interactive Systems Conference* (San Diego, CA, USA) *(DIS '19)*. Association for Computing Machinery, New York, NY, USA, 633–644. https://doi.org/10.1145/3322276.3322340

[36] Michal Luria, Rebecca Zheng, Bennett Huffman, Shuangni Huang, John Zimmerman, and Jodi Forlizzi. 2020. Social Boundaries for Personal Agents in the Interpersonal Space of the Home. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376311

[37] David Maulsby, Saul Greenberg, and Richard Mander. 1993. Prototyping an intelligent agent through Wizard of Oz. In *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*. 277–284.

[38] Juliana Miehle, Daniel Ostler, Nadine Gerstenlauer, and Wolfgang Minker. 2017. The next step: intelligent digital assistance for clinical operating rooms. *Innovative surgical sciences* 2, 3 (2017), 159–161.

[39] Cosmin Munteanu, Matt Jones, Sharon Oviatt, Stephen Brewster, Gerald Penn, Steve Whittaker, Nitendra Rajput, and Amit Nanavati. 2013. We need to talk: HCI and the delicate topic of spoken language interaction. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*. 2459–2464.

[40] Christine Murad and Cosmin Munteanu. 2019. " I don't know what you're talking about, HALexa" the case for voice user interface guidelines. In *Proceedings of the 1st International Conference on Conversational User Interfaces*. 1–3.

[41] Christine Murad, Cosmin Munteanu, Benjamin R Cowan, and Leigh Clark. 2019. Revolution or Evolution? Speech Interaction and HCI Design Guidelines. *IEEE Pervasive Computing* 18, 2 (2019), 33–45.

[42] Moyen Mohammad Mustaquim. 2013. Automatic speech recognition-an approach for designing inclusive games. *Multimedia tools and applications* 66, 1 (2013), 131–146.

[43] Aasish Pappu and Alexander Rudnicky. 2014. Knowledge acquisition strategies for goal-oriented dialog systems. In *Proceedings of the 15th annual meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*. 194–198.

[44] Jim R Parker and John Heerema. 2008. Audio interaction in computer mediated games. *International Journal of Computer Games Technology* 2008 (2008).

[45] Cathy Pearl. 2016. *Designing voice user interfaces: principles of conversational experiences.* " O'Reilly Media, Inc.".

[46] Perfect Tap Games. [n.d.]. *Chicken Scream.* Game [Android and IOS]. Unit 3269, DMCC Business Centre Level No 1, Jewellery and Gemplex 3 , Dubai, United Arab Emirates.

[47] Martin Porcheron, Joel E. Fischer, Stuart Reeves, and Sarah Sharples. 2018. Voice Interfaces in Everyday Life. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. ACM, New York, NY, USA, Article 640, 12 pages. https://doi.org/10.1145/3173574.3174214

[48] Martin Porcheron, Joel E. Fischer, and Sarah Sharples. 2017. "Do Animals Have Accents?": Talking with Agents in Multi-Party Conversation. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing* (Portland, Oregon, USA) *(CSCW '17)*. ACM, New York, NY, USA, 207–219. https://doi.org/10.1145/2998181.2998298

[49] Robert Porzel and Manja Baudis. 2004. The Tao of CHI: Towards Effective Human-Computer Interaction. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*. Association for Computational Linguistics, Boston, Massachusetts, USA, 209–216. https://www.aclweb.org/anthology/N04-1027

[50] Aung Pyae and Paul Scifleet. 2018. Investigating differences between native English and non-native English speakers in interacting with a voice user interface: A case of Google Home. In *Proceedings of the 30th Australian Conference on Computer-Human Interaction*. 548–553.

[51] D Reddy, Lee Erman, and R Neely. 1973. A model and a system for machine recognition of speech. *IEEE Transactions on Audio and Electroacoustics* 21, 3 (1973), 229–238.

[52] Richard M Ryan, C Scott Rigby, and Andrew Przybylski. 2006. The motivational pull of video games: A self-determination theory approach. *Motivation and emotion* 30, 4 (2006), 344–360.

[53] Adam J Sporka, Sri H Kurniawan, Murni Mahmud, and Pavel Slavík. 2006. Non-speech input and speech recognition for real-time control of computer games. In *Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility*. 213–220.

[54] Turn 10 Studios. 2013. *Forza Motorsport 5.* Game [XBox One]. Microsoft Studios, Redmond, Washington, U.S.

[55] Markku Turunen, Jaakko Hakulinen, K-J Raiha, E-P Salonen, Anssi Kainulainen, and Perttu Prusi. 2005. An architecture and applications for speech-based accessibility systems. *IBM Systems Journal* 44, 3 (2005), 485–504.

[56] Marco Filipe Ganança Vieira, Hao Fu, Chong Hu, Nayoung Kim, and Sudhanshu Aggarwal. 2014. PowerFall: a voice-controlled collaborative game. In *Proceedings of the first ACM SIGCHI annual symposium on Computer-human interaction in play*. 395–398.

[57] Tom Wilcox, Mike Evans, Chris Pearce, Nick Pollard, and Veronica Sundstedt. 2008. Gaze and voice based game interaction: the revenge of the killer penguins. *SIGGRAPH Posters* 81, 10.1145 (2008), 1400885–1400972.

[58] Rainer Winkler, Matthias Söllner, Maya Lisa Neuweiler, Flavia Conti Rossini, and Jan Marco Leimeister. 2019. Alexa, Can You Help Us Solve This Problem?: How Conversations With Smart Personal Assistant Tutors Increase Task Group Outcomes. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI EA '19)*. ACM, New York, NY, USA, Article LBW2311, 6 pages. https://doi.org/10.1145/3290607.3313090

[59] Beste F. Yuksel, Penny Collisson, and Mary Czerwinski. 2017. Brains or Beauty: How to Engender Trust in User-Agent Interactions. *ACM Trans. Internet Technol.* 17, 1, Article 2 (Jan. 2017), 20 pages. https://doi.org/10.1145/2998572

[60] Nima Zargham, Michael Bonfert, Georg Volkmar, Robert Porzel, and Rainer Malaka. 2020. Smells Like Team Spirit: Investigating the Player Experience with Multiple Interlocutors in a VR Game. In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY EA '20)*. ACM.

Understanding Instructions all the Way: A Simulation-based Approach

First Author
Affiliation — Line 1
Affiliation — Line 2
Affiliation — Line 3
contact@email

Second Author
Affiliation — Line 1
Affiliation — Line 2
Affiliation — Line 3
contact@email

Abstract

In this paper we introduce a new architecture for systems that understand instructions. In our case textual instructions are given to autonomous robotic agents to be executed correspondingly. The proposed architecture and our implementation of a system conforming to that architecture go beyond the traditional bounds of natural language understanding to include abstract schematic/counterfactual reasoning, physics-based simulation as well as a human computation loop for improving specific parameterizations of simulations and learning new aspects of meaning; we consider these functionalities essential for understanding instructions 'all the way'. We evaluate the system in the domain of everyday activities, such as cooking or setting a table, where household robots are tasked to carry out diverse activities based on given linguistic input.

1 Introduction

If the proof of the pudding is in the eating then the ultimate test for understanding an instruction is its proper execution. This view greatly expands the scope of natural language understanding (NLU) beyond the usual syntactic and semantic analysis. To illustrate this extended scope let us consider a seemingly simple example such as *put the cup on the table*. Most NLU systems would not consider the actual grounding of the respective objects, where exactly on the table the cup is to be placed, or where the handle should face at the end. However, we know that mental simulations are crucial in human processing for understanding both visual as well as linguistic input (Bergen et al., 2007; Feldman, 2008) and as soon as we seek to run a computational simulation of putting a cup on a table, we have to provide a grounding and specific answers to questions of the kind raised above.

Robotic proficiency is possible for well-defined tasks and in constrained environments (Nilsson, 1984; Kunze et al., 2010; Beetz et al., 2018), but ends when confronted with vague and underspecified instructions in unpredictably varying contexts. The central aim of our broader research effort is to progress from robots executing specific tasks to ones mastering the corresponding activity, responding flexibly and appropriately as contexts change. Our domain encompasses everyday activities as they occur in households, e.g. preparing meals, cleaning or setting tables. A specific problem lies in enabling autonomous robots to execute the vaguely specified instructions typical of recipes or uttered as verbal commands in such scenarios. To get from an underspecified instruction to an executable plan may be considered definitional for 'understanding' in this domain and is a challenging enterprise that requires:

- various types of knowledge, i.e, linguistic as well as ontological world knowledge,
- different types of reasoning, i.e. for parameterization and comporting,
- flexible language analysis of imperative constructions, i.e. semantic parsing.

In this work we introduce a natural language understanding system that outputs fully parameterized structures that satisfy expected logical constraints and support task execution. After discussing pertinent related work, we present the overall architecture and individual components in greater detail and showcase typical examples of the system at work.

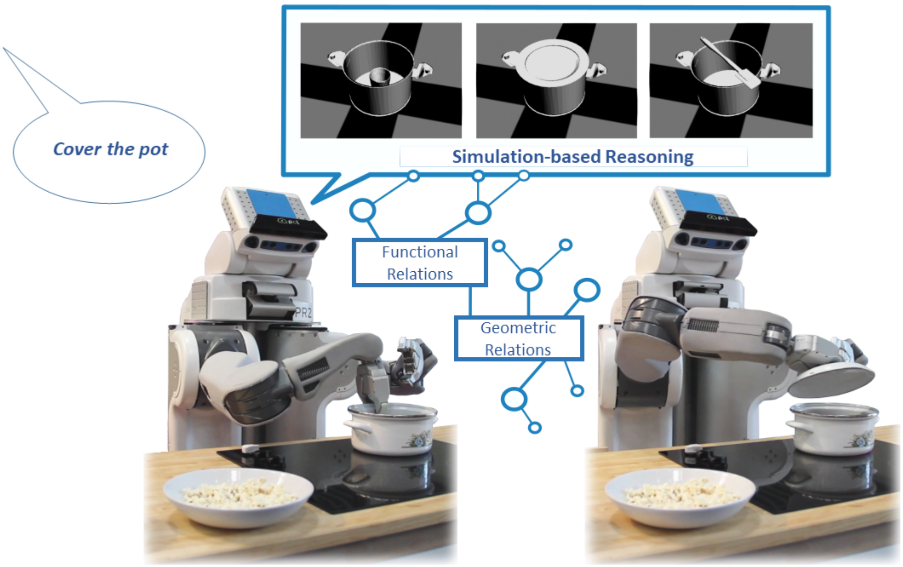


Figure 1: A PR2 Robot executing an underspecified task

2 Related Work

One of the earliest research programs to study autonomous robots was the Shakey project (Nilsson, 1984). Shakey was a mobile robot that used planning to reason about its actions and performed tasks that required planning of paths and actions as well as re-arranging of simple objects. This work was seminal for the fields of classical planning and computer vision. Nevertheless, even in Shakey's simple environment, the limitations of the approach became clear, as the computational complexity of planning problems proved, in general, to be intractable. Many researchers (Kunze et al., 2010; Nilsson, 1984; Nielsen et al., 2010; Parde et al., 2015; Walther-Franks et al., 2015) have worked on providing robotic systems with human-like common sense knowledge so that the robots could, hopefully, avoid costly planning from scratch or trial and error. Knowledge representation and reasoning in autonomous robot control is an extensive ield of research with developments in service and industrial robotics. Olivares et al. provide a comprehensive comparison of different approaches (Olivares-Alarcos et al., 2019).

One example in the industrial robotics domain is the ROSETTA project (Patel et al., 2012; Malec et al., 2013; Stenmark et al., 2015). Its initial scope was reconfiguration and adaptation of robot-based manufacturing cells, however, the authors have, since then, further developed their activity modeling for coping with a wider range of industrial tasks. Other authors have focused on modeling industrial task structure, part geometry features, or task teaching from examples (Balakirsky et al., 2013; Balakirsky, 2015; Polydoros et al., 2016; Kootbally et al., 2015; Perzylo et al., 2016). Compared to the everyday activity domain, industrial tasks considered in above works are more structured, and less demanding in terms of flexibility.

An approach to activity modeling in the service robotics domain is presented by Tenorth and Beetz (Tenorth and Beetz, 2015). The scope of their work is similar to ours as the authors also consider how activity knowledge can be used to fill knowledge gaps in abstract instructions given to a robotic agent performing everyday activities. However, the scope of the work presented here is wider, as we also consider how activity knowledge can be used for the interpretation of natural language instructions. Our activity modeling and the ensuing reasoning processes are more detailed in terms of activity structure as we also consider the processes and states that occur during an activity. Another difference is that, in their modeling, there is no distinction between physical and social context, and therefore less expressivity compared to our model.

A more general approach to activity modeling for robotic agents is presented by the IEEE-RAS working group ORA (Schlenoff et al., 2012). The group has the goal of defining a standard ontology for various sub-domains of robotics, including a model for object manipulation tasks. It has defined a core ORA ontology (Prestes et al., 2013), as well as additional modules for industrial tasks such as kitting (Fiorini et al., 2015). In terms of methodology, we differ in foundational assumptions we assert, which has important consequences on the structure of our ontology, modeling workflow, and inferential power. In the case of ORA the SUMO upper-level ontology is used as foundational layer. Compared to SUMO, we use a richer axiomatization of entities on the foundational layer, and put particular emphasis on the distinction between physical and social activity context as will be discussed in Section 3.2.

In our minds, the incorporation of principles of semantics and pragmatics is essential for building systems that can reasonably be said to understand natural language. In the past decades research in cognitive linguistics has yielded valuable insights in these areas, but often lacked a form that is rigorous enough for implementation. Early logical and statistical approaches to natural language, especially unification-based approaches (Shieber, 1986), constitute a a starting point for capturing these insights. However, additional representational tools and techniques are needed for formalizing the conceptual primitives of cognitive linguistics that provide a basis for scalable language understanding systems.

Cognitively motivated approaches to linguistics have sought to demonstrate how diverse phenomena affecting language use are grounded in the rest of cognition. The meanings of linguistic units are subject to category effects (Lakoff, 1987) and largely based on abstractions over sensorimotor patterns, called *image schemas* (Johnson, 1987) and *force-dynamic schemas* (Talmy, 2000). They are often defined against a constellation of related concepts captured in a *frame* (Fillmore, 1988). Apparently exotic phenomena, including metaphorical inference (Lakoff and Johnson, 1980) and mental space phenomena (Fauconnier, 1985), are taken as reflecting basic facts of cognitive organization.

In general, linguistic knowledge is seen as a collection of conventionalized pairings between form and meaning (Langacker, 1987) so called *constructions* (Goldberg, 1995; Fillmore, 1988). Corresponding computational approaches have brought forth the embodied construction grammar analyser (Bryant, 2003) and the Babel parser based on fluid construction grammar (FCG) (Steels and De Beule, 2006). For the work described here, we implemented a semantic parser that is technically akin to the Babel system and representationally close to embodied construction grammar (Chang et al., 2002); we describe this in Section 3.3 below.

In contrast to the earlier systems, we employ a set of ontologies, based on the SOMA framework (hidden for review), for modelling the semantics of instructions as well as the world knowledge required to understand them. Various approaches to model linguistic knowledge, i.e. the entities and features that make up human language, in formal ontologies have been proposed. These approaches differ in some respects such as alignment to upper layers, their modeling intent and their scope, and their specific alignment to well-specified foundational ontologies. While, for example, the GOLD ontology (Farrar and Langendoen, 2004) is aligned to the SUMO upper ontology (Niles and Pease, 2001), the OntoWordNet model (Gangemi et al., 2003), is aligned to the DOLCE foundational ontology (Masolo et al., 2003a). The LingInfo model (Cimiano et al., 2006) can be used with any foundational framework as it relies on meta-classes to model information about the lexical entities. In contrast, the OntoWordNet aims at merging the linguistic information contained in WordNet with the respective classes employed in specific domain models, while both LingInfo and GOLD seek to incorporate more linguistic information, such as morphological and grammatical features of language. They all allow for a direct connection of the respective linguistic information to corresponding classes and properties in a domain ontology.

These efforts are, in a sense, orthogonal to ours and each model could be integrated as an additional module to allow reasoning about linguistic information or as a link between lexical and ontological resources. For those purposes we actually employ a *Lower Semantic Model* that connects lexical and ontological information and is interchangeable with several of the models described above. More closely tuned to the representation of the propositional content of instructions is the so-called *General Upper Model* (GUM) (Bateman et al., 2010). GUM provides a detailed semantics for linguistic spatial expressions based on a principled ontological engineering approach. It covers language concerned with space, actions in space and functional spatial relationships in particular, for which an ontological organization is proposed relating such expressions to general classes of fixed semantic import. However, as we seek to align our model with a specific foundational model and construct it as a module within the SOMA ontological framework, we do not employ the upper model GUM as is, but re-use relevant details concerning schematic theories about functional relations (see below) where applicable.

Overall, the representations needed for knowledge about actions have also been a topic of research because the symbolic, highly abstract, 'actions as black boxes' representations of the Shakey era have not resulted in robust behaviors in realistic environments. In general, action knowledge tends to be subsymbolic, and often takes the form of success/failure probability distributions over an action's parameter space (Stulp et al., 2012; Winkler et al., 2017). Note that the experience from which a robot may learn does not have to be from the real world: simulated episodes, produced either by a human player of a game or a robot simulating itself, can also be used for this purpose. Simulation will of course not provide a complete description of a realistic action, but even very coarse simulation can already be useful for a robot that needs to validate its plans and/or pick a better set of parameters (Mösenlechner and Beetz, 2013) as we discuss in Section 3.4.

3 An Understanding Architecture

In the following, we set out the components and features of our architecture for turning underspecified linguistic instructions into executable robotic plans. In principle this architecture can be regarded as an update and extension of the systems proposed and tested as part of the Neural Theory of Language project (Narayanan, 1999; Eppe et al., 2016). While physics-based simulation engines were unavailable at the beginning of this seminal project, the idea of using simulation as an integral part of a language understanding system has been around for half a century by now. What has changed is the technology available for reasoning with simulations as well as for implementing construction-based semantic analyzers.

3.1 An Overview: Components

Figure 2 demonstrates the flow of information, as well as the impact of and interaction between the particular modules of our system. Offering an interface to any set of natural language based instructions, the information in those instruction is processed in order to represent subtasks as tuples of manipulation actions and objects acted on or with as described in Section 3.3. In natural language, it is usual that the ontological scope of any objects required is severely underspecified because humans are used to working with generalized information and specifying these in terms of individual choice, influenced by world knowledge, availability and preference. Yet, this underspecification does not render all possible objects contained in a general term as viable (or usual). Thus, we add human knowledge in a specification layer through *Kitchen Clash*, a serious game presenting a decision making paradigm within the relevant set of objects (hidden for review). Parallel to this, complementary world knowledge is derived from physical properties of the objects and their surroundings via simulation. In both approaches, object choices can be quantified and thus ranked by efficiency and effectiveness. Within the simulation, this assessment can be realized in a fully autonomous manner, while the serious game offers further qualitative insights, since peer-rated quality measurements are included in the rating process as well as preference and conventionality measures. World knowledge is successively aggregated with trajectorial and contextual information and provided as *narrative-enabled episodic memories* (*NEEMs*) according to the KnowRob paradigm (Beetz et al., 2018). All the representational structures exchanged by the individual components are based on a set of ontologies that will be described in the following section.

3.2 Interface Definitions via the SOMA Ontology

We decided to base our model on the DOLCE+DnS Ultralite (DUL) foundational framework (Masolo et al., 2003b). This decision is strongly motivated by their underlying ontological commitments. Firstly, DUL is not a revisionary model, but seeks to express standpoints that shape human cognition. Furthermore, it assumes a reductionist approach: rather than capturing, for example, the flexibility of the usage of objects via multiple inheritance in a multiplicative manner, we commit to a reduced *ground* classification and use a *descriptive* approach for handling this flexibility. For this a primary branch of the ontology

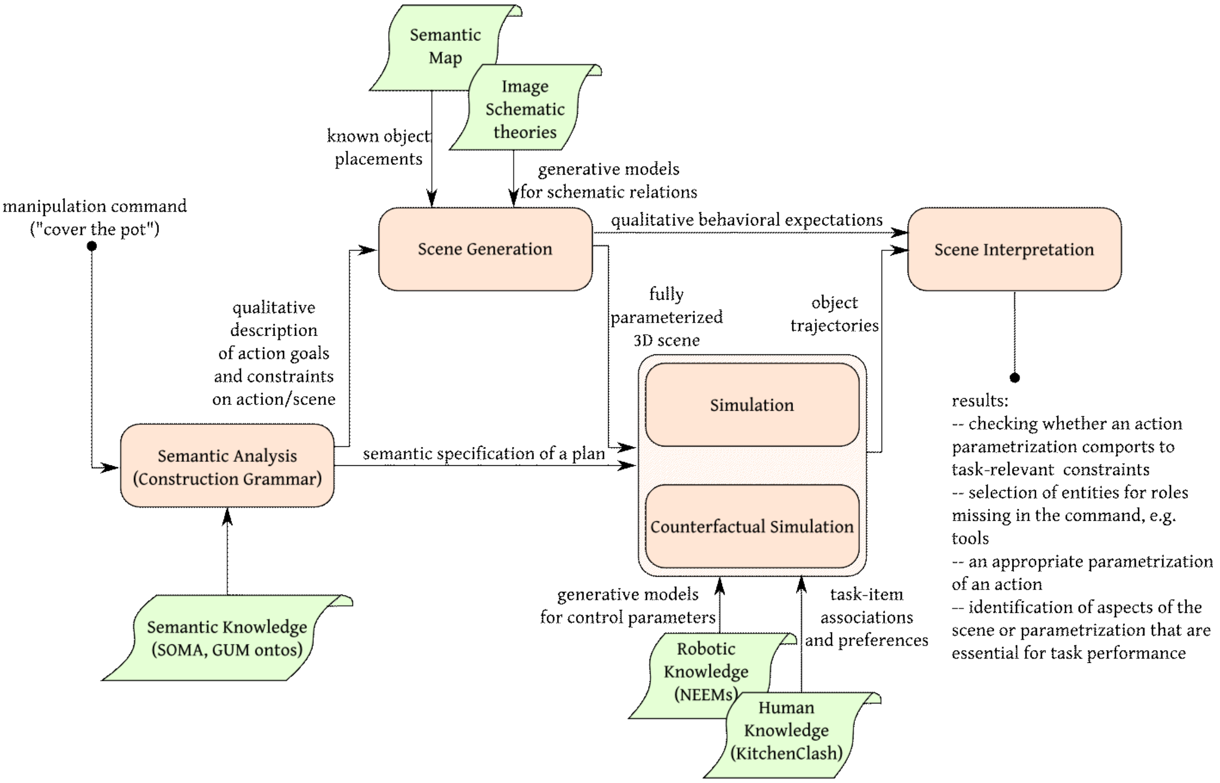

Semantic Map
Image Schematic theories
known object placements
generative models for schematic relations
manipulation command ("cover the pot")
Scene Generation
qualitative behavioral expectations
Scene Interpretation
qualitative description of action goals and constraints on action/scene
fully parameterized 3D scene
object trajectories
Simulation
Semantic Analysis (Construction Grammar)
semantic specification of a plan
Counterfactual Simulation
results:
-- checking whether an action parametrization comports to task-relevant constraints
-- selection of entities for roles missing in the command, e.g. tools
-- an appropriate parametrization of an action
-- identification of aspects of the scene or parametrization that are essential for task performance
generative models for control parameters
task-item associations and preferences
Semantic Knowledge (SOMA, GUM ontos)
Robotic Knowledge (NEEMs)
Human Knowledge (KitchenClash)

Figure 2: Components of the NLU pipeline together with knowledge sources. The input may come from a user request or from a linguistic description of how to perform a task, e.g. a recipe or wikiHow instructional text.

represents the ground physical model, e.g. objects and actions, while a secondary branch represents the social model, e.g. roles and tasks. All entities in the social branch are ontologically dependent on humans and constitute social objects representing concepts about, or descriptions of, ground elements.

Every axiomatization in the physical branch can, therefore, be regarded as expressing some physical context, whereas axiomatizations in the descriptive social branch are used to express social contexts. A set of dedicated relations is provided that connect both branches. For example, the relation *classifies* connects ground objects, e.g. a hammer, with the roles they can play, i.e. potential classifications. Thus, we can state that a hammer can in some context be conceptualized as a murder weapon, a paper weight or a door stopper. Nevertheless, neither will its ground ontological classification as a tool change nor will hammers be subsumed as kinds of door stoppers, paper weights or weapons via multiple inheritance.

3.3 Language Analysis

The genre of written instructions, manuals, and recipes, which is commonly employed in the household domain, poses many linguistic challenges. One such is the phenomena of homonyms, which, mainly due to conversion, tend to arise across domain boundaries. In particular, short instructions, such as ”season with salt” or ”cover”, regularly cause part of speech tagging issues for syntactic parsers trained on ‘standard’ declarative corpora. On the syntactic level, instructions ordinarily employ imperatives, subjunctive mood is used to describe desired world states, and implicit as well as explicit conditions are prevalent. Also, instructions make heavy use of ellipses, omitting both determiners and direct objects (Ruppenhofer and Michaelis, 2010), thus sentences such as ”Stir until smooth” are common. In consequence, an understanding system capable of analysing instructions requires mechanisms for anaphora resolution, as well as tools to handle a broad variety of kinds of ambiguities.

In this work, a deep semantic parser based on the Construction Grammar formalism (Fillmore, 1988; Goldberg, 1995; Fillmore and Kay, 1999) is employed in order to flexibly analyse abstract and underspecified instructions. Both the constructions’ meaning poles and the analysis itself make use of ontological knowledge to guide extensive search processes, disambiguate otherwise unclear instructions, and to evoke unspecified parameters which need to be inferred by later processing steps. In this way, natural language commands are transformed into interpretations of ontological models that consist of a series of scenes and state transitions specifying the evoked schemas; this is described in Section 3.4.4.

The underlying mechanism of the employed parser is based on the unification and merging algorithms implemented in FCG (Steels and De Beule, 2006), while its ontology integration and schema handling are inspired by Embodied Construction Grammar (Chang et al., 2002). The parsing process integrates tightly into our knowledge base-focused understanding pipeline, representing both the internal data structures as well as the semantic output as semantic triples. In general, integrating with the infrastructure of a robotic system is no trivial task, with considerable complexity stemming from integrating many diverse data sources, systems, and viewpoints. This growing complexity makes it harder to retain formalisability and implementability, makes integration with other components and data sources harder, and impacts the ability to experiment with algorithms and their implementation. In other words, complexity begets complexity. To counteract this, we simplify the feature structure-based representations and algorithms found in FCG with the expectation that a simpler grammar formalism will prove easier to implement and be more supportive of integration with its surrounding systems. We approach this twofold. First, we deconstruct the feature structure into triples. These provide the minimal unit of information as [unit feature value] pairs, which, instead of operating on monolithic composite structures, also enable fine-grained reasoning with individual facts. This further removes all special cases and rules related to the position of symbols within units. Performing this deconstruction thus simplifies the formal models, as well as allowing ideas and optimisations to be imported from the extensive body of research on database and knowledge representation theory and implementation. Furthermore, it aligns the internal structures of the parser and grammar with that of the surrounding ontologies and middleware used for communication with subsequent systems. This not only reduces the complexity that arises when binding multiple systems together, but also allows reasoning about them in a holistic way.

The grammar employed in the system was engineered for the domain of household instructions, covering the ensuing domain-specific linguistic challenges presented above. The benefit of integrating ontological knowledge into the grammar itself is clearly evidenced by instructions such as ”Whisk the eggs into the pan”. In particular, one construction querying for the semantic category of the final referring expression is critical in disambiguating the sentence’s meaning from that of syntactically analogous instructions, such as ”Roll the dough into a rectangular shape”. The sentence ”Put the dough into the bowl and cover it.” is another such example. This instruction sequence expresses two state transitions and respective pre- and post-scenes. Due to both the type of verb and the preposition used, the CAUSED-MOTIONSCHEMA is satisfied by the first transition, while the post-scenes specify CONTAINMENT and COVERAGE configurations respectively. Mentioned entities are employed as corresponding role fillers, and the final pronoun is resolved to a preceding ‘coverable’ entity, in this case the ”bowl”. The resulting qualitative description, encompassing scenes, actions, and pre- and post-conditions, is then handed to our subsequent scene generation and simulation systems.

3.4 Schematic and counterfactual simulation

As mental simulations are part of the human understanding process, we include simulations as an integral part of the understanding pipeline to inform our understanding based on realistic and physical considerations. In the following we sketch this inclusion in more detail.

3.4.1 Scene generation

A simulation in a physics engine requires very concretely specified objects: shapes, positions, velocities etc. must be known. Typically, such information is absent from linguistic expressions of command and from typical descriptions of object arrangements. In the case of understanding a command, a significant part of the bridge between these different levels of abstraction can be established by using shared knowledge of the environment. It is presumed that both human user and robot know where unmovable walls or door frames, or rarely-moved furniture, are. In the case of the robot, this knowledge is formalized as a semantic map (Beetz et al., 2010) which includes all the information needed to build a 3D scene, together with additional semantic annotation to describe facts about the objects, such as type and purpose.

However, semantic maps tend to cover only objects that are fixed. Some movable items may be perceived by the robot and as such be included in its belief state – a representation of the world that is more up to date than the semantic map; but some movable items may be located where the robot does not see them. This makes it useful to include a *scene generation* procedure in addition: this takes as input some linguistic description of an arrangement of objects, optionally together with some incomplete 3D scene, and produces a fully specified 3D scene satisfying the linguistic description. To define this procedure, we again distinguish several levels of description. These are, in decreasing order of abstraction:

- *Functional relations*: imply locations and constrain expected behavior of objects. Examples include Containment ("the pot contains the popcorn"), Support ("the table supports the pot"), Coverage ("the plate covers the pot").

- *Locations*: qualitatively describe the position or movement of a locatum or trajector object relative to a relatum object. The description is in terms of geometric primitive relations. Examples include OnTopOf ("the pot is on the table"), Inside ("the popcorn is in the pot").

- *Geometric primitive relations*: describe the relative poses or movements of geometric primitives of objects. Examples include AxisAlignment, PointContainment, SurfaceContainment.

- *Geometric primitives*: identifiable features of object shapes, which can be specified either in an object-centric or world-centric reference frame. Examples include Centroid, ObjectRelativeForward, WorldRelativeTopSurface.

- *Fully-specified objects*: these contain all the information needed to run a physics simulator. This includes shape, coordinates, velocities, physical parameters.

Typically, linguistic descriptions operate at the level of functional and/or spatial relations, and it is necessary to proceed down through the abstraction layers as listed.

The first step moves from functional and spatial relations to geometric primitive relations. This is achieved by instantiating propositional Horn logic theories based on the participants in a functional relation or location. The theory for one level of abstraction may only use entities from a lower level of abstraction. An example fragment of the theory for OnTopOf is:

$$Location(X, Y,' OnTopOf') \rightarrow$$
$$SurfaceContainment(ObjectRelativeBottomSurface(X), WorldRelativeTopSurface(Y))$$

Once at the level of geometric primitive relations level, we use generative models to guide constrained sampling. The approach is that some parameter of the object, such as its orientation or translation, may take a value from a grid in some appropriate space. For translation, this is a suitably large, axis aligned box in $\mathbb{R}^3$; for orientation this is a point on a Fibonacci sphere sampling of S^3, the space of quaternions. That is, a parameter may take values from a discrete set of points $\mathscr{M}$. Each geometric primitive relation g that is relevant for that parameter defines a probability distribution over the set $\mathscr{M}$, which we denote by $P(p|g)$, for $p \in \mathscr{M}$. We make a simplifying assumption that the constraints are independent, so the combined distribution resulting from applying several constraints at once is approximated by:

$$P(p|g_1, g_2, .., g_n) \approx \frac{P(p|g_1) * P(p|g_2) * ..P(p|g_n)}{\sum_{p \in \mathscr{M}} P(p|g_1) * P(p|g_2) * ..P(p|g_n)} \tag{1}$$

The resulting probability distribution controls how likely it is for a particular point to be sampled as a candidate value for the parameter. Some post-sampling validation steps are sometimes needed. In the case of translation and orientation, the validation is a collision check against objects already present in the partially constructed scene.

3.4.2 Physics simulation

The semantic specification of a robot action plan, together with a fully parameterized 3D scene, is passed to a physics simulator where a model of the robot performs the requested action. We distinguish here between *parameters*, i.e. numerical values describing the manner of an action, and *arguments* of the semantic specification, i.e. references to objects that play certain roles in a plan. Control or trajectory parameters, e.g. the exact speed with which to move something, or the exact force with which to push a tool, are rarely specified in requests for action. Moreover, even the tool with which to perform a task is often omitted. In such cases, semantic analysis may know what kind of information – parameter or argument – is missing, but not be able to provide a filler for it.

To fill in control parameters, we use sampling from a probability distribution similar to "action related places" (Stulp et al., 2012): given a collection of episodes of the robot performing the same task, knowledge of the used parameters for an episode, and whether the episode resulted in a successful completion of the task, one can define a generative model for a control parameter by computing the probability for a particular parameter value, conditioned on the episode being successful. The episodes, i.e. NEEMs, are stored in a knowledge base and include, alongside raw sensor and control data, semantic annotations about the performed task, its arguments, and its outcome. Selection of objects to fill in the roles of a plan is done similarly using NEEMs. A second method which we also employ is to use results from human computation elicited by games with a purpose. Specifically, we use the KitchenClash (hidden for review) game to acquire knowledge about which tools are appropriate or preferred for what combinations of tasks and objects being acted on.

The role of simulation in our pipeline is to provide knowledge going beyond the linguistic. In particular, physics simulators are the appropriate tools to provide quantitative information about the physical interactions occurring during the performance of a task and about outcomes under different plan parameterizations. Ultimately, the decision of whether an outcome is acceptable, or an interaction unwanted, is a qualitative decision, which we describe in the next subsection, but this decision must also be informed by relevant physical considerations.

3.4.3 Scene interpretation

A naive approach towards *scene interpretation* would be to treat it as a dual to scene generation, and therefore as a procedure to take data about object trajectories, obtained via physics simulation, and convert this into a qualitative, more abstract description. However there are many ways in which to describe the same trajectories, e.g. by relating the movement of a trajector to every other object in the scene. Clearly, this is not practical, and only some of these descriptions are interesting for an agent. Therefore, we define scene interpretation as checking whether the trajectories observed in a simulation of a scene obey expectations placed on object behaviors by the functional relations which are asserted to hold in the scene. These expectations are formulated in terms of primitive movements, which are descriptions of the movement of a trajector object relative to a relatum object. Examples of primitive movements are RelativeDistancing and RelativeApproaching.

A primitive movement is formalized as a function which takes two trajectories as arguments, and computes based on them a cost depending on how well these trajectories obey the primitive movement. If this cost is above a certain threshold, the trajectories fail to meet the description of the primitive movement. We then distinguish between factual and counterfactual expectations. "Factual" expectations are those which should be met by the simulation of the scene as it actually is, whereas "counterfactual" expectations are defined for alternate scenes where objects from the original scene are successively removed.

3.4.4 Integrating simulation into the NLU pipeline

The basic task of the generation-simulation-interpretation part of our pipeline is to answer whether a qualitatively specified arrangement of objects will behave in a qualitatively specified way. For example: will cooking popcorn kernels stay inside a pot? What if the pot is covered with a heavy/very lightweight item? It is also possible, via counterfactual simulations, to understand which objects contribute to the observed behavior conforming to specification or not. It is then possible to infer using counterfactual simulation that the lid of a pot contributes to keeping cooking popcorn kernels inside the pot, even when no explicit containment relation is asserted between the lid and popcorn.

In this basic use case, the input is a qualitative scene description, which is then converted into a fully specified scene, and based on simulation results it is decided which expectations hold or not. The expectations are tested over the entire simulation timeline. However, usage in the natural language understanding pipeline imposes different requirements: the scene is at least partially known and will include an agent who will act upon it in some way, therefore potentially changing which expectations are in effect at different times. For example, when a robot puts a cup on the table, the expectations for the table supporting the cup are only in force after the cup placement.

The partially known scene is not a difficulty for the approach we presented in the previous subsection: some of the objects are already specified, with scene generation being then restricted only to objects the robot does not have full information about, but expects to be present in the scene. For example, the robot might simulate a scene of itself moving some cups from a cupboard to a table and use scene generation to place simulated cups in the simulated cupboard, before it has actually opened the real one. In our approach the action the robot must perform is modeled as a state transition, with a pre- and a post-scene as the two states on either end of the transition. Each of the two states and the transition correspond to intervals on the simulation timeline, and each carries its own set of expectations:

- pre-scene: expectations defined by the semantic map or other contextual knowledge the robot might have about its environment, e.g. what object supports or contains what other one,

- action: expectations derived from the functional relations the parser infers should hold during an action, e.g. that a cup should preserve its contents during transport,

- post-scene: expectations derived from the functional relations the parser infers should hold after an action completes successfully, e.g. that a cup should be supported by the surface it was placed on.

After completing a simulation, data about object trajectories is split into the corresponding intervals for the pre-/post-scenes and action, and the schematic process of scene interpretation we have previously described is applied to each interval in turn. This allows us to both check that a robot performs an action properly – e.g. not spilling coffee on the way – as well as obtain acceptable results in the world – e.g. the cup it carried does not fall off the edge of the table.

4 Conclusion

In this work we have described a language understanding pipeline that implements the basic principles of cognitive linguistics merged into a larger framework for cognitive robotics. For this we include a semantically driven analysis of the given linguistic input, i.e. instructions, that specifies the basic settings for generating virtual scenes in which required actions are parameterized and explicated to the degree that actual motion planners running on a robotic agent can execute them. While also different in many respects, this basic separation of work can also be observed in the premotor and motor cortex of natural agents.

We consider this work to constitute a significant step towards creating flexible, robust and scalable natural language understanding systems that can be deployed and run in real time on artificial agents that are tasked to carry out everyday activities. As the proposed structures and interface definitions are a part of an ongoing international standardization effort, we expect this approach to become used in multiple research contexts and robotics laboratories, where different reasoning approaches can be combined and tested. We do not claim to have created an optimal assembly of components and approaches, but hope to set up standardized benchmarks for evaluating systems capable of turning vague and underspecified input into executable plans.

References

Stephen Balakirsky, Zeid Kootbally, Thomas Kramer, Anthony Pietromartire, Craig Schlenoff, and Satyandra Gupta. 2013. Knowledge driven robotics for kitting applications. *Robot. Auton. Syst.*, 61(11):1205–1214, November.

Stephen Balakirsky. 2015. Ontology based action planning and verification for agile manufacturing. *Robotics and Computer-Integrated Manufacturing*, 33(Supplement C):21 – 28. Special Issue on Knowledge Driven Robotics and Manufacturing.

John A. Bateman, Joana Hois, Robert Ross, and Thora Tenbrink. 2010. A linguistic ontology of space for natural language processing. *Artif. Intell.*, 174(14):1027–1071, September.

Michael Beetz, Lorenz Mösenlechner, and Moritz Tenorth. 2010. Cram—a cognitive robot abstract machine for everyday manipulation in human environments. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1012–1017. IEEE.

Michael Beetz, Daniel Bessler, Andrei Haidu, Mihai Pomarlan, Asil Kaan Bozcuoglu, and Georg Bartels. 2018. Know Rob 2.0 - A 2nd Generation Knowledge Processing Framework for Cognition-Enabled Robotic Agents. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 512–519.

Benjamin K. Bergen, Shane Lindsay, Teenie Matlock, and Srini Narayanan. 2007. Spatial and linguistic aspects of visual imagery in sentence comprehension. *Cognitive Science*, 31(5):733–764.

John Bryant. 2003. Semantic analysis with ECG. In *Proceedings of 8th International Conference on Cognitive Linguistics*.

Nancy Chang, Jerome Feldman, Robert Porzel, and Keith Sanders. 2002. Scaling cognitive linguistics: Formalisms for language understanding. In *Proceedings of the First International Workshop On Scalable Natural Language Understanding*.

Philipp Cimiano, Paul Buitelaar, Anette Frank, Stefania Racioppa, Michael Sintek, Malte Kiesel, Massimo Romanelli, Berenike Loos, Thierry Declerck, Ralf Engel, Daniel Sonntag, Vanessa Micelli, and Robert Porzel. 2006. Linginfo: Design and applications of a model for the integration of linguistic information in ontologies. In *Proceedings of the OntoLex Workshop at LREC*, pages 28–32. ELRA, Mai.

Manfred Eppe, Sean Trott, and Jerome Feldman. 2016. Exploiting Deep Semantics and Compositionality of Natural Language for Human-Robot-Interaction. *arXiv:1604.06721 [cs]*, April. arXiv: 1604.06721.

Scott Farrar and Terry Langendoen. 2004. A linguistic ontology for the semantic web. *GLOT International*, 7:97–100, 06.

Gilles Fauconnier. 1985. *Mental Spaces: Aspects of Meaning Construction in Natural Language*. MIT Press/Bradford, Cambridge, Mass. and London.

Jerome A. Feldman. 2008. *From Molecule to Metaphor - A Neural Theory of Language*. MIT Press.

Charles Fillmore and Paul Kay. 1999. *Construction grammar*. CSLI, Stanford, CA.

Charles Fillmore. 1988. The mechanisms of construction grammar. In *Berkeley Linguistics Society*, volume 14, pages 35–55.

Sandro Rama Fiorini, Joel Luis Carbonera, Paulo Gonçalves, Vitor A.M. Jorge, Vítor Fortes Rey, Tamás Haidegger, Mara Abel, Signe A. Redfield, Stephen Balakirsky, Veera Ragavan, Howard Li, Craig Schlenoff, and Edson Prestes. 2015. Extensions to the core ontology for robotics and automation. *Robot. Comput.-Integr. Manuf.*, 33(C):3–11, June.

Aldo Gangemi, Roberto Navigli, and Paola Velardi. 2003. The ontowordnet project: Extension and axiomatization of conceptual relations in wordnet. In Robert Meersman, Zahir Tari, and Douglas C. Schmidt, editors, *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*, pages 820–838, Berlin, Heidelberg. Springer Berlin Heidelberg.

Adele E. Goldberg. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. University of Chicago Press.

Mark Johnson. 1987. *The Body in the Mind Metaphors*. University of Chicago Press.

Z. Kootbally, C. Schlenoff, C. Lawler, T. Kramer, and S.K. Gupta. 2015. Towards robust assembly with knowledge representation for the planning domain definition language (pddl). *Robot. Comput.-Integr. Manuf.*, 33(C):42–55, June.

Lars Kunze, Moritz Tenorth, and Michael Beetz. 2010. Putting people's common sense into knowledge bases of household robots. In *Annual Conference on Artificial Intelligence*, pages 151–159. Springer.

George Lakoff and Mark Johnson. 1980. *Metaphors We Live By*. University of Chicago Press.

George Lakoff. 1987. *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*. University of Chicago Press.

Ronald W. Langacker. 1987. *Foundations of Cognitive Grammar, Vol. 1*. Stanford University Press.

J. Malec, K. Nilsson, and H. Bruyninckx. 2013. Describing assembly tasks in declarative way. In *IEEE/ICRA Workshop on Semantics*.

Claudio Masolo, Stefano Borgo, Aldo Gangemi, Nicola Guarino, and Alessandro Oltramari. 2003a. Wonderweb deliverable d18, ontology library (final). *ICT project*, 33052:31.

Claudio Masolo, Stefano Borgo, Aldo Gangemi, Nicola Guarino, and Alessandro Oltramari. 2003b. WonderWeb deliverable D18 ontology library (final). Technical report, IST Project 2001-33052 WonderWeb: Ontology Infrastructure for the Semantic Web.

Lorenz Mösenlechner and Michael Beetz. 2013. Fast temporal projection using accurate physics-based geometric reasoning. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1821–1827, Karlsruhe, Germany, May 6–10.

Srinivas Narayanan. 1999. Reasoning about actions in narrative understanding. In *Proceedings of the 16th International Joint Conference on Artifical Intelligence - Volume 1*, IJCAI'99, page 350–355, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Rodney D Nielsen, Richard Voyles, Daniel Bolanos, Mohammad H Mahoor, Wilson D Pace, Katie A Siek, and Wayne H Ward. 2010. A platform for human-robot dialog systems research. In *2010 AAAI Fall Symposium Series*.

Ian Niles and Adam Pease. 2001. Towards a standard upper ontology. In *Proceedings of the International Conference on Formal Ontology in Information Systems - Volume 2001*, FOIS '01, page 2–9, New York, NY, USA. Association for Computing Machinery.

Nils J Nilsson. 1984. Shakey the robot. Technical report, SRI INTERNATIONAL MENLO PARK CA.

Alberto Olivares-Alarcos, Daniel Beßler, Alaa Khamis, Paulo Gonçalves, Maki Habib, J. Bermejo, Marcos Barreto, Mohammed Diab, Jan Rosell, João Quintas, Joanna Olszewska, Hirenkumar Nakawala, Edison Pignaton de Freitas, Amelie Gyrard, Stefano Borgo, Guillem Alenyà, Michael Beetz, and Howard Li. 2019. A review and comparison of ontology-based approaches to robot autonomy. *The Knowledge Engineering Review*, 34, 12.

Natalie Paige Parde, Michalis Papakostas, Konstantinos Tsiakas, Maria Dagioglou, Vangelis Karkaletsis, and Rodney D Nielsen. 2015. I spy: An interactive game-based approach to multimodal robot learning. In *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Rajendra Patel, Mikael Hedelind, and Pablo Lozan-Villegas. 2012. Enabling robots in small-part assembly lines: The "rosetta approach" - an industrial perspective. In *ROBOTIK*. VDE-Verlag.

Alexander Perzylo, Nikhil Somani, Stefan Profanter, Ingmar Kessler, Markus Rickert, and Alois Knoll. 2016. Intuitive instruction of industrial robots: Semantic process descriptions for small lot production. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2293–2300.

Athanasios S. Polydoros, Bjarne Großmann, Francesco Rovida, Lazaros Nalpantidis, and Volker Krüger. 2016. Accurate and versatile automation of industrial kitting operations with skiros. In *Towards Autonomous Robotic Systems - 17th Annual Conference (TAROS)*, pages 255–268.

Edson Prestes, Joel Luis Carbonera, Sandro Rama Fiorini, Vitor A. M. Jorge, Mara Abel, Raj Madhavan, Angela Locoro, Paulo Goncalves, Marcos E. Barreto, Maki Habib, Abdelghani Chibani, Sébastien Gérard, Yacine Amirat, and Craig Schlenoff. 2013. Towards a core ontology for robotics and automation. *Robotics and Autonomous Systems*, 61(11):1193 – 1204. Ubiquitous Robotics.

Josef Ruppenhofer and Laura Michaelis. 2010. A constructional account of genre-based argument omissions. *Constructions and Frames*, 2:158–184, 01.

Craig Schlenoff, Edson Prestes, Raj Madhavan, Paulo Goncalves, Howard Li, Stephen Balakirsky, Thomas Kramer, and Emilio Miguelanez. 2012. An IEEE standard ontology for robotics and automation. In *IEEE Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1337–1342.

Stuart M. Shieber. 1986. *An Introduction to Unification-based Approaches to Grammar*. Stanford University/CSLI, Stanford, Cal.

Luc Steels and Joachim De Beule. 2006. Unify and merge in fluid construction grammar. In *Proceedings of the Third International Workshop on the Emergence and Evolution of Linguistic Communication (EELC 2006)*, pages 197–223.

Maj Stenmark, Jacek Malec, Klas Nilsson, and Anders Robertsson. 2015. On distributed knowledge bases for robotized small-batch assembly. *IEEE Transactions on Automation Science and Engineering*, 12(2):519–528.

Freek Stulp, Andreas Fedrizzi, Lorenz Mösenlechner, and Michael Beetz. 2012. Learning and Reasoning with Action-Related Places for Robust Mobile Manipulation. *Journal of Artificial Intelligence Research (JAIR)*, 43:1–42.

Leonard Talmy. 2000. *Toward a Cognitive Semantics. Volume 2: Typology and Process in Concept Structuring*. Language, Speech, and Communication. MIT Press, Cambridge, MA.

Moritz Tenorth and Michael Beetz. 2015. Representations for robot knowledge in the knowrob framework. *Artificial Intelligence*.

Benjamin Walther-Franks, Jan Smeddinck, Peter Szmidt, Andrei Haidu, Michael Beetz, and Rainer Malaka. 2015. Robots, pancakes, and computer games: designing serious games for robot imitation learning. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3623–3632. ACM.

Jan Winkler, Asil Kaan Bozcuoğlu, Mihai Pomarlan, and Michael Beetz. 2017. Task parametrization through multi-modal analysis of robot experiences. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, AAMAS ’17, pages 1754–1756, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

References

Giovanni Acampora, Vincenzo Loia, and Autilia Vitiello. Improving game bot behaviours through timed emotional intelligence. *Knowledge-Based Systems*, 34:97–113, 2012.

Ernest Adams. Balancing games with positive feedback. *Gamasutra. com, January*, 4, 2002.

Sabbir Ahmad, Andy Bryant, Erica Kleinman, Zhaoqing Teng, Truong-Huy D Nguyen, and Magy Seif El-Nasr. Modeling individual and team behavior through spatio-temporal analysis. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pages 601–612. ACM, 2019.

Gustavo Andrade, Geber Ramalho, Hugo Santana, and Vincent Corruble. Extending reinforcement learning to provide dynamic game balancing. In *Proceedings of the Workshop on Reasoning, Representation, and Learning in Computer Games, 19th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 7–12, 2005.

Dennis Ang and Alex Mitchell. Comparing effects of dynamic difficulty adjustment systems on video game experience. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pages 317–327. ACM, 2017.

Dennis Ang and Alex Mitchell. Representation and frequency of player choice in player-oriented dynamic difficulty adjustment systems. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pages 589–600, 2019.

ArenaNet. *Guild Wars*. Game [PC], April 2005. ArenaNet, Bellevue, WA.

Mehrdad Bahrini, Nima Zargham, Johannes Pfau, Stella Lemke, Karsten Sohr, and Rainer Malaka. Good vs. evil: Investigating the effect of game premise in a smart home security educational game. In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '20, page 182–187, New York, NY, USA, 2020a. Association for Computing Machinery. doi: 10.1145/3383668.3419887.

Mehrdad Bahrini, Nima Zargham, Johannes Pfau, Stella Lemke, Karsten Sohr, and Rainer Malaka. Enhancing game-based learning through infographics in the context of smart home security. In *Joint International Conference on Entertainment Computing and Serious Games*. Springer, 2020b.

R.J.S. Baker and P.I. Cowling. Bayesian Opponent Modeling in a Simple Poker Environment. In *2007 IEEE Symposium on Computational Intelligence and Games*, pages 125–131, April 2007. doi: 10.1109/CIG.2007.368088. ISSN: 2325-4289.

Sander CJ Bakkes, Pieter HM Spronck, and H Jaap Van Den Herik. Opponent modelling for case-based adaptive game ai. *Entertainment Computing*, 1(1):27–37, 2009.

C Bauckhage, B Gorman, C Thurau, and M Humphrys. Learning human behavior from analyzing activities in virtual environments. *MMI-Interaktiv*, 12:3–17, 2007.

Christian Bauckhage, Christian Thurau, and Gerhard Sagerer. Learning Human-Like Opponent Behavior for Interactive Computer Games. In Bernd Michaelis and Gerald Krell, editors, *Pattern Recognition*, Lecture Notes in Computer Science, pages 148–155, Berlin, Heidelberg, 2003. Springer. ISBN 978-3-540-45243-0. doi: 10.1007/978-3-540-45243-0_20.

Leonard E Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, 37(6):1554–1563, 1966.

Philipp Beau and Sander Bakkes. Automated game balancing of asymmetric video games. In *2016 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–8. IEEE, 2016.

Jennifer Hernández Bécares, Luis Costero Valero, and Pedro Pablo Gómez Martín. An approach to automated videogame beta testing. *Entertainment Computing*, 18:79–92, 2017.

Hans J Berliner. Backgammon computer program beats world champion. *Artificial Intelligence*, 14(2):205–220, 1980.

Darse Billings, Denis Papp, Jonathan Schaeffer, and Duane Szafron. Opponent modeling in poker. *Aaai/iaai*, 493:499, 1998.

Jason M. Bindewald, Gilbert L. Peterson, and Michael E. Miller. Clustering-Based Online Player Modeling. In Tristan Cazenave, Mark H.M. Winands, Stefan Edelkamp, Stephan Schiffel, Michael Thielscher, and Julian Togelius, editors, *Computer Games*, Communications in Computer and Information Science, pages 86–100, Cham, 2017. Springer International Publishing. ISBN 978-3-319-57969-6. doi: 10.1007/978-3-319-57969-6_7.

BioWare. *Mass Effect 3*. Game [PC, XBox360, PS3, WiiU], March 2012. BioWare, Edmonton, Canada.

Blizzard Entertainment. *Heroes of the Storm*. Game [PC], June 2015. Blizzard Entertainment, Irvine, CA, USA. Played 2017.

Blizzard Entertainment. *Overwatch*. Game [PC,PS4,XboxOne,Switch], May 2016.

Charles L Bouton. Nim, a game with a complete mathematical theory. *Annals of Mathematics*, 3(1/4):35–39, 1901.

Leo Breiman, Joseph H Friedman, R. A. Olshen, and C. J. Stone. Classification and regression trees. 1983.

Christian Buhl and Fazeel Gareeboo. Automated testing: A key factor for success in video game development. case study and lessons learned. In *proceedings of Pacific NW Software Quality Conferences*, pages 1–15, 2012.

Bungie. *Halo: Combat Evolved*. Game [XBox, PC], November 2001. Bungie, Chicago, IL, USA.

Paolo Burelli and Georgios N. Yannakakis. Adapting virtual camera behaviour through player modelling. *User Modeling and User-Adapted Interaction*, 25(2):155–183, June 2015. ISSN 1573-1391. doi: 10.1007/s11257-015-9156-4.

Elizabeth Camilleri, Georgios N. Yannakakis, and Antonios Liapis. Towards general models of player affect. In *2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 333–339, October 2017. doi: 10.1109/ACII.2017.8273621. ISSN: 2156-8111.

Murray Campbell, A Joseph Hoane Jr, and Feng-hsiung Hsu. Deep blue. *Artificial intelligence*, 134(1-2): 57–83, 2002.

Alessandro Canossa and Anders Drachen. Play-Personas: Behaviours and Belief Systems in User-Centred Game Design. In Tom Gross, Jan Gulliksen, Paula Kotzé, Lars Oestreicher, Philippe Palanque, Raquel Oliveira Prates, and Marco Winckler, editors, *Human-Computer Interaction – INTERACT 2009*, Lecture Notes in Computer Science, pages 510–523, Berlin, Heidelberg, 2009. Springer. ISBN 978-3-642-03658-3. doi: 10.1007/978-3-642-03658-3_55.

Fabio Reis Cecin, Rodrigo Real, Rafael de Oliveira Jannone, CF Resin Geyer, Marcio Garcia Martins, and JL Victoria Barbosa. Freemmg: A scalable and cheat-resistant distribution model for internet games. In *Eighth IEEE International Symposium on Distributed Simulation and Real-Time Applications*, pages 83–90. IEEE, 2004.

Ben Chan, Jörg Denzinger, Darryl Gates, Kevin Loose, and John Buchanan. Evolutionary behavior testing of commercial computer games. In *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*, volume 1, pages 125–132. IEEE, 2004.

Darryl Charles and Michaela Black. Dynamic player modeling: A framework for player-centered digital games. In *Proc. of the International Conference on Computer Games: Artificial Intelligence, Design and Education*, pages 29–35, 2004.

Kuan-Ta Chen, Jhih-Wei Jiang, Polly Huang, Hao-Hua Chu, Chin-Laung Lei, and Wen-Chin Chen. Identifying mmorpg bots: A traffic analysis approach. *EURASIP Journal on Advances in Signal Processing*, 2009 (1):797159, 2008.

Zhao Chen and Darvin Yi. The Game Imitation: Deep Supervised Convolutional Networks for Quick Video Game AI. *arXiv:1702.05663 [cs]*, February 2017. arXiv: 1702.05663.

EU Condon. The nimatron. *The American Mathematical Monthly*, 49(5):330–332, 1942.

Thomas Constant and Guillaume Levieux. Dynamic difficulty adjustment impact on players' confidence. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, page 463. ACM, 2019.

Mihaly Csikszentmihalyi, Sami Abuhamdeh, Jeanne Nakamura, et al. Flow, 1990.

Vickie Curtis. Motivation to participate in an online citizen science game: A study of foldit. *Science Communication*, 37(6):723–746, 2015.

Daedalic Entertainment. *Anna's Quest*. Game [PC], July 2015. Daedalic Entertainment, Hamburg, Germany. Played 2017.

Aaron Davidson, Darse Billings, Jonathan Schaeffer, and Duane Szafron. Improved opponent modeling in poker. In *International Conference on Artificial Intelligence, ICAI'00*, pages 1467–1473, 2000.

Edirlei Soares de Lima, Bruno Feijó, and Antonio L Furtado. Player behavior and personality modeling for interactive storytelling in games. *Entertainment Computing*, 28:32–48, 2018.

Fernando de Mesentier Silva, Scott Lee, Julian Togelius, and Andy Nealen. Ai as evaluator: Search driven playtesting of modern board games. In *AAAI Workshops*, 2017.

Thomas Debeauvais. *Challenge and retention in games*. PhD thesis, UC Irvine, 2016.

Simon Demediuk, Marco Tamassia, Xiaodong Li, and William L Raffe. Challenging ai: Evaluating the effect of mcts-driven dynamic difficulty adjustment on player enjoyment. In *ACSW*, pages 43–1, 2019.

Shawn M Doherty, Devin Liskey, Christopher M Via, Christina Frederick, Jason P Kring, and Dahai Liu. An analysis of expressed cheating behaviors in video games. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 58, pages 2393–2396. SAGE Publications Sage CA: Los Angeles, CA, 2014.

Anders Drachen, Alessandro Canossa, and Georgios N. Yannakakis. Player modeling using self-organization in Tomb Raider: Underworld. In *2009 IEEE Symposium on Computational Intelligence and Games*, pages 1–8, September 2009. doi: 10.1109/CIG.2009.5286500. ISSN: 2325-4289.

Anders Drachen, Rafet Sifa, Christian Bauckhage, and Christian Thurau. Guns, swords and data: Clustering of player behavior in computer games in the wild. In *2012 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 163–170, September 2012. doi: 10.1109/CIG.2012.6374152. ISSN: 2325-4289.

Epic Games. *Fortnite*. Game [PC,Switch,PS4,XboxOne,iOS,Android], July 2017.

William Rao Fernandes and Guillaume Levieux. δ-logit: Dynamic difficulty adjustment using few data points. In *Joint International Conference on Entertainment Computing and Serious Games*, pages 158–171. Springer, 2019.

Firaxis Games. *Civilization V*. Game [PC], September 2010. Firaxis Games, Hunt Valley, Maryland. Played 2017.

Julian Frommel, Fabian Fischbach, Katja Rogers, and Michael Weber. Emotion-based dynamic difficulty adjustment using parameterized difficulty and self-reports of emotion. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*, pages 163–171. ACM, 2018.

Pablo García-Sánchez, Alberto Tonda, Antonio M Mora, Giovanni Squillero, and Juan Julián Merelo. Automated playtesting in collectible card games using evolutionary algorithms: A case study in hearthstone. *Knowledge-Based Systems*, 153:133–146, 2018.

Quentin Gemine, Firas Safadi, Raphaël Fonteneau, and Damien Ernst. Imitative learning for real-time strategy games. In *2012 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 424–429, September 2012. doi: 10.1109/CIG.2012.6374186. ISSN: 2325-4289.

Bernard Gorman, Christian Thurau, Christian Bauckhage, and Mark Humphrys. Believability Testing and Bayesian Imitation in Interactive Computer Games. In Stefano Nolfi, Gianluca Baldassarre, Raffaele Calabretta, John C. T. Hallam, Davide Marocco, Jean-Arcady Meyer, Orazio Miglino, and Domenico Parisi, editors, *From Animals to Animats 9*, Lecture Notes in Computer Science, pages 655–666, Berlin, Heidelberg, 2006. Springer. ISBN 978-3-540-38615-5. doi: 10.1007/11840541_54.

Ícaro Goulart, Aline Paes, and Esteban Clua. Learning how to play bomberman with deep reinforcement and imitation learning. In *Joint International Conference on Entertainment Computing and Serious Games*, pages 121–133. Springer, 2019.

Chengjie Gu, Shunyi Zhang, Xiaozhen Xue, and He Huang. Online wireless mesh network traffic classification using machine learning. volume 7, pages 1524–1532, 2011.

Stefan Freyr Gudmundsson, Philipp Eisen, Erik Poromaa, Alex Nodet, Sami Purmonen, Bartlomiej Kozakowski, Richard Meurling, and Lele Cao. Human-like playtesting with deep learning. In *2018 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–8. IEEE, 2018.

Yong Guo, Siqi Shen, Otto Visser, and Alexandru Iosup. An analysis of online match-based games. In *2012 IEEE International Workshop on Haptic Audio Visual Environments and Games (HAVE 2012) Proceedings*, pages 134–139. IEEE, 2012.

Brian Guthrie, Kevin Reuter, Michael Barkdoll, and Henry Hexmoor. Small team group dynamics in online games. *COOS: Scope and theme*, page 42, 2014.

Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

He He, Jordan Boyd-Graber, Kevin Kwok, and Hal Daumé III. Opponent modeling in deep reinforcement learning. In *International Conference on Machine Learning*, pages 1804–1813, 2016.

Hello Games. *No Man's Sky*. Game [PC, PS4, XBoxOne], August 2016. Hello Games, Guildford, UK.

Todd Hester, Matej Vecerik, Olivier Pietquin, Marc Lanctot, Tom Schaul, Bilal Piot, Dan Horgan, John Quan, Andrew Sendonaris, Ian Osband, Gabriel Dulac-Arnold, John Agapiou, Joel Z. Leibo, and Audrunas Gruslys. Deep Q-learning From Demonstrations. In *Thirty-Second AAAI Conference on Artificial Intelligence*, April 2018. URL `https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16976`.

Rob High. The era of cognitive systems: An inside look at ibm watson and how it works. *IBM Corporation, Redbooks*, 2012.

Sylvain Hilaire, Hyun-chul Kim, and Chong-kwon Kim. How to deal with bot scum in mmorpgs? In *2010 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR 2010)*, pages 1–6. IEEE, 2010.

Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE, 1995.

Nadia Hocine, Abdelkader Gouaich, and Stefano A Cerri. Dynamic difficulty adaptation in serious games for motor rehabilitation. In *International Conference on Serious Games*, pages 115–128. Springer, 2014.

Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N Yannakakis. Generative agents for player decision modeling in games. In *FDG*. Citeseer, 2014.

Christoffer Holmgard, Michael Cerny Green, Antonios Liapis, and Julian Togelius. Automated playtesting with procedural personas with evolved heuristics. *IEEE Transactions on Games*, 2018.

Christoffer Holmgård, Julian Togelius, and Georgios N. Yannakakis. Decision Making Styles as Deviation from Rational Action: A Super Mario Case Study. In *Ninth Artificial Intelligence and Interactive Digital Entertainment Conference*, November 2013. URL `https://www.aaai.org/ocs/index.php/AIIDE/AIIDE13/paper/view/7391`.

Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis. Evolving personas for player decision modeling. In *2014 IEEE Conference on Computational Intelligence and Games*, pages 1–8, August 2014a. doi: 10.1109/CIG.2014.6932911. ISSN: 2325-4270.

Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis. Personas versus Clones for Player Decision Modeling. In Yusuf Pisan, Nikitas M. Sgouros, and Tim Marsh, editors, *Entertainment Computing – ICEC 2014*, Lecture Notes in Computer Science, pages 159–166, Berlin, Heidelberg, 2014b. Springer. ISBN 978-3-662-45212-7. doi: 10.1007/978-3-662-45212-7_20.

Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis. Monte-Carlo Tree Search for Persona Based Player Modeling. In *Eleventh Artificial Intelligence and Interactive Digital Entertainment Conference*, September 2015.

Christoffer Holmgård, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis. Evolving models of player decision making: Personas versus clones. *Entertainment Computing*, 16:95–104, July 2016. ISSN 1875-9521. doi: 10.1016/j.entcom.2015.09.002.

Shay Horovitz and Danny Dolev. Collabrium: Active traffic pattern prediction for boosting p2p collaboration. In *2009 18th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises*, pages 116–121. IEEE, 2009.

Ryan Houle. Player modeling for adaptive games. 2006.

Dayana Hristova. Dynamic difficulty adjustment (dda) in first person shooter (fps) games, 2017.

Kenneth Hullett, Nachiappan Nagappan, Eric Schuh, and John Hopson. Empirical analysis of user data in game software development. In *Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, pages 89–98. IEEE, 2012.

Robin Hunicke. The case for dynamic difficulty adjustment in games. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pages 429–433. ACM, 2005.

Sidra Iftikhar, Muhammad Zohaib Iqbal, Muhammad Uzair Khan, and Wardah Mahmood. An automated model based testing approach for platform games. In *2015 ACM/IEEE 18th International Conference on Model Driven Engineering Languages and Systems (MODELS)*, pages 426–435. IEEE, 2015.

Max Jaderberg, Wojciech M Czarnecki, Iain Dunning, Luke Marris, Guy Lever, Antonio Garcia Castaneda, Charles Beattie, Neil C Rabinowitz, Ari S Morcos, Avraham Ruderman, et al. Human-level performance in 3d multiplayer games with population-based reinforcement learning. *Science*, 364(6443):859–865, 2019.

Alexander Jaffe, Alex Miller, Erik Andersen, Yun-En Liu, Anna Karlin, and Zoran Popovic. Evaluating competitive game balance with restricted play. In *Eighth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2012.

W Lewis Johnson, Hannes Högni Vilhjálmsson, and Stacy Marsella. Serious games for language learning: How much game, how much ai? In *AIED*, volume 125, pages 306–313, 2005.

Niels Justesen and Sebastian Risi. Learning macromanagement in starcraft from replays using deep learning. In *2017 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 162–169. IEEE, 2017.

Arnaud Kaiser, Dario Maggiorini, Nadjib Achir, and Khaled Boussetta. On the objective evaluation of real-time networked games. In *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference*, pages 1–5. IEEE, 2009.

Igor V. Karpov, Jacob Schrum, and Risto Miikkulainen. Believable Bot Navigation via Playback of Human Traces. In Philip Hingston, editor, *Believable Bots: Can Computers Play Like People?*, pages 151–170. Springer, Berlin, Heidelberg, 2012. ISBN 978-3-642-32323-2. doi: 10.1007/978-3-642-32323-2_6. URL `https://doi.org/10.1007/978-3-642-32323-2_6`.

Donald Kehoe. Designing artificial intelligence for games. *URL https://software. intel. com/en-us/articles/designing-artificial-intelligence-for-gamespart-1*, 2009.

Ahmed Khalifa, Aaron Isaksen, Julian Togelius, and Andy Nealen. Modifying mcts for human-like general video game playing. In *IJCAI*, pages 2514–2520, 2016.

Hyungil Kim, Sungwoo Hong, and Juntae Kim. Detection of auto programs for mmorpgs. In *Australasian Joint Conference on Artificial Intelligence*, pages 1281–1284. Springer, 2005.

Larian Studios. *Divinity: Original Sin 2*. Game [PC,XBoxOne,PS4,Switch], September 2017. Larian Studios, Gent, Belgium.

Geoffrey Lee, Min Luo, Fabio Zambetta, and Xiaodong Li. Learning a Super Mario controller from examples of human play. In *2014 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8, July 2014. doi: 10.1109/CEC.2014.6900246. ISSN: 1941-0026.

Ryan Leigh, Justin Schonfeld, and Sushil J Louis. Using coevolution to understand and validate game balance in continuous games. In *Proceedings of the 10th annual conference on Genetic and evolutionary computation*, pages 1563–1570. ACM, 2008.

Chris Lewis and Noah Wardrip-Fruin. Mining game statistics from web services: a world of warcraft armory case study. In *Proceedings of the Fifth International Conference on the Foundations of Digital Games*, pages 100–107. Citeseer, 2010.

Nicholas Liao, Matthew Guzdial, and Mark Riedl. Deep convolutional player modeling on log and level data. In *Proceedings of the 12th International Conference on the Foundations of Digital Games*, page 41. ACM, 2017.

Antonios Liapis, Georgios N. Yannakakis, and Julian Togelius. Designer Modeling for Personalized Game Content Creation Tools. In *Ninth Artificial Intelligence and Interactive Digital Entertainment Conference*, November 2013. URL `https://www.aaai.org/ocs/index.php/AIIDE/AIIDE13/paper/view/7448`.

Antonios Liapis, Christoffer Holmgård, Georgios N. Yannakakis, and Julian Togelius. Procedural Personas as Critics for Dungeon Generation. In Antonio M. Mora and Giovanni Squillero, editors, *Applications of Evolutionary Computation*, Lecture Notes in Computer Science, pages 331–343, Cham, 2015. Springer International Publishing. ISBN 978-3-319-16549-3. doi: 10.1007/978-3-319-16549-3_27.

Dayi Lin, Cor-Paul Bezemer, and Ahmed E Hassan. Studying the urgent updates of popular games on the steam platform. *Empirical Software Engineering*, 22(4):2095–2126, 2017.

Lionhead Studios. *Black & White*. Game [PC], April 2001. Lionhead Studios, Guildford, UK.

Changchun Liu, Pramila Agrawal, Nilanjan Sarkar, and Shuo Chen. Dynamic difficulty adjustment in computer games through real-time anxiety-based affective feedback. *International Journal of Human-Computer Interaction*, 25(6):506–529, 2009.

Sarayut Lueangrueangroj and Vishnu Kotrajaras. Real-time imitation based learning for commercial fighting games. *Proc. of Computer Games, Multimedia and Allied Technology*, 9:1–3, 2009.

Tobias Mahlmann, Anders Drachen, Julian Togelius, Alessandro Canossa, and Georgios N. Yannakakis. Predicting player behavior in Tomb Raider: Underworld. In *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games*, pages 178–185, August 2010. doi: 10.1109/ITW.2010.5593355. ISSN: 2325-4289.

Tobias Mahlmann, Julian Togelius, and Georgios N Yannakakis. Evolving card sets towards balancing dominion. In *2012 IEEE Congress on Evolutionary Computation*, pages 1–8. IEEE, 2012.

Hector P. Martinez, Yoshua Bengio, and Georgios N. Yannakakis. Learning deep physiological models of affect. *IEEE Computational Intelligence Magazine*, 8(2):20–33, May 2013. ISSN 1556-6048. doi: 10.1109/MCI.2013.2247823.

Maxis. *Spore*. Game [PC], May 2008. Maxis, Redwood Shores, CA, USA.

Philipp Mayring. Qualitative inhaltsanalyse. In *Handbuch qualitative Forschung in der Psychologie*, pages 601–613. Springer, 2010.

Nazanin Mehrasa, Yatao Zhong, Frederick Tung, Luke Bornn, and Greg Mori. Deep learning of player trajectory representations for team activity analysis. In *11th MIT Sloan Sports Analytics Conference*, 2018.

David Melhart, Ahmad Azadvar, Alessandro Canossa, Antonios Liapis, and Georgios N. Yannakakis. Your Gameplay Says It All: Modelling Motivation in Tom Clancy's The Division. *arXiv:1902.00040 [cs, stat]*, May 2019. arXiv: 1902.00040.

Cale Michael. https://dotesports.com/, Jul 2019. URL `https://dotesports.com/dota-2/news/at-least-one-player-leaves-in-11-7-percent-of-all-dota-2-matches`.

Philip Mildner, Tonio Triebel, Stephan Kopf, and Wolfgang Effelsberg. A scalable peer-to-peer-overlay for real-time massively multiplayer online games. In *Proceedings of the 4th international ICST conference on simulation tools and techniques*, pages 304–311. ICST (Institute for Computer Sciences, Social-Informatics and . . . , 2011.

Maximiliano Miranda, Antonio A Sánchez-Ruiz, and Federico Peinado. A neuroevolution approach to imitating human-like play in ms. pac-man video game. In *CoSECivi*, pages 113–124, 2016.

Yutaro Mishima, Kensuke Fukuda, and Hiroshi Esaki. An analysis of players and bots behaviors in mmorpg. In *2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA)*, pages 870–876. IEEE, 2013.

Olana Missura and Thomas Gärtner. Player Modeling for Intelligent Difficulty Adjustment. In João Gama, Vítor Santos Costa, Alípio Mário Jorge, and Pavel B. Brazdil, editors, *Discovery Science*, Lecture Notes in Computer Science, pages 197–211, Berlin, Heidelberg, 2009. Springer. ISBN 978-3-642-04747-3. doi: 10.1007/978-3-642-04747-3_17.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Ryan M Moeller, Bruce Esplin, Steven Conway, et al. Cheesers, pullers, and glitchers: The rhetoric of sportsmanship and the discourse of online sports gamers. *Game Studies*, 9(2), 2009.

Mojang. *Minecraft*. Game [PC], November 2011. Mojang, Stockholm, Sweden.

K Mørch. Cheating in online games-threats and solutions. *Publication No: DART/01/03. January*, 2003.

Mihail Morosan and Riccardo Poli. Automated game balancing in ms pacman and starcraft using evolutionary algorithms. In *European Conference on the Applications of Evolutionary Computation*, pages 377–392. Springer, 2017.

NCSoft. *Lineage II*. Game [PC], October 2003.

NCsoft. *Aion*. Game [PC], September 2008. NCSoft, Seongnam, South Korea. Played August 2019.

Allen Newell, Herbert Alexander Simon, et al. *Human problem solving*, volume 104. Prentice-hall Englewood Cliffs, NJ, 1972.

Nintendo EAD. *Zelda II: The Adventure of Link*. Game [NES], January 1987. Nintendo EAD, Kyoto, Japan.

Nintendo EAD. *Mario Kart 8*. Game [WiiU,Switch], May 2014. Nintendo EAD, Kyoto, Japan. Played 2019.

Nintendo R&D1. *Metroid Fusion*. Game [GBA], November 2002. Nintendo RD1, Kyoto, Japan.

Pedro A Nogueira, Vasco Torres, Rui Rodrigues, Eugénio Oliveira, and Lennart E Nacke. Vanishing scares: biofeedback modulation of affective player experiences in a procedural horror game. *Journal on Multimodal User Interfaces*, 10(1):31–62, 2016.

In-Seok Oh, Ho-Chul Cho, and Kyung-Joong Kim. Imitation learning for combat system in RTS games with application to starcraft. In *2014 IEEE Conference on Computational Intelligence and Games*, pages 1–2, August 2014. doi: 10.1109/CIG.2014.6932919. ISSN: 2325-4270.

Jehwan Oh, Zoheb Hassan Borbora, and Jaideep Srivastava. Automatic detection of compromised accounts in mmorpgs. In *2012 International Conference on Social Informatics*, pages 222–227. IEEE, 2012.

Jehwan Oh, Zoheb Hassan Borbora, Dhruv Sharma, and Jaideep Srivastava. Bot detection based on social interactions in mmorpgs. In *2013 International Conference on Social Computing*, pages 536–543. IEEE, 2013.

Jacob Kaae Olesen, Georgios N Yannakakis, and John Hallam. Real-time challenge balance in an rts game using rtneat. In *2008 IEEE Symposium On Computational Intelligence and Games*, pages 87–94. IEEE, 2008.

Juan Ortega, Noor Shaker, Julian Togelius, and Georgios N. Yannakakis. Imitating human playing styles in Super Mario Bros. *Entertainment Computing*, 4(2):93–104, April 2013. ISSN 1875-9521. doi: 10.1016/j.entcom.2012.10.001.

Michail Ostrowski and Samir Aroudj. Automated regression testing within video game development. *GSTF Journal on Computing (JoC)*, 3(2):1–5, 2013.

Nathan Partlan, Abdelrahman Madkour, Chaima Jemmali, Josh Aaron Miller, Christoffer Holmgård, and Magy Seif El-Nasr. Player imitation for build actions in a real-time strategy game. 2019.

Christopher Pedersen, Julian Togelius, and Georgios N. Yannakakis. Modeling Player Experience for Content Creation. *IEEE Transactions on Computational Intelligence and AI in Games*, 2(1):54–67, March 2010. ISSN 1943-0698. doi: 10.1109/TCIAIG.2010.2043950.

Johannes Pfau and Rainer Malaka. Can you rely on human computation? a large-scale analysis of disruptive behavior in games with a purpose. In *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*, CHI PLAY '19 Extended Abstracts, page 605–610, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3341215.3356297.

Johannes Pfau and Rainer Malaka. We asked 100 people: How would you train our robot? In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '20, page 335–339, New York, NY, USA, 2020. Association for Computing Machinery. doi: 10.1145/3383668.3419864.

Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. Automated game testing with icarus: Intelligent completion of adventure riddles via unsupervised solving. In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '17 Extended Abstracts, page 153–164, New York, NY, USA, 2017. Association for Computing Machinery. doi: 10.1145/3130859.3131439.

Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. Towards deep player behavior models in mmorpgs. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '18, page 381–392, New York, NY, USA, 2018a. Association for Computing Machinery. doi: 10.1145/3242671.3242706.

Johannes Pfau, Jan David Smeddinck, Georg Volkmar, Nina Wenig, and Rainer Malaka. Do you think this is a game? In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI EA '18, page 1–6, New York, NY, USA, 2018b. Association for Computing Machinery. doi: 10.1145/3170427.3188651.

Johannes Pfau, Robert Porzel, Mihai Pomarlan, Vanja Sophie Cangalovic, Supara Grudpan, Sebastian Höffner, John Bateman, and Rainer Malaka. Give meanings to robots with kitchen clash: A vr human computation serious game for world knowledge accumulation. In *Joint International Conference on Entertainment Computing and Serious Games*, pages 85–96. Springer, 2019a.

Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. Deep player behavior models: Evaluating a novel take on dynamic difficulty adjustment. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI EA '19, page 1–6, New York, NY, USA, 2019b. Association for Computing Machinery. doi: 10.1145/3290607.3312899.

Johannes Pfau, Antonios Liapis, Georg Volkmar, Georgios Yannakakis, and Rainer Malaka. Dungeons & replicants: Automated game balancing via deep player behavior modeling. In *2020 IEEE Conference on Games (CoG)*, pages 431–438. IEEE, 2020a. doi: 10.1109/CoG47356.2020.9231958.

Johannes Pfau, Jan David Smeddinck, Ioannis Bikas, and Rainer Malaka. Bot or not? user perceptions of player substitution with deep player behavior models. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–10, New York, NY, USA, 2020b. Association for Computing Machinery. doi: 10.1145/3313831.3376223.

Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. Enemy within: Long-term motivation effects of deep player behavior models for dynamic difficulty adjustment. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–10, New York, NY, USA, 2020c. Association for Computing Machinery. doi: 10.1145/3313831.3376423.

Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. The case for usable ai: What industry professionals make of academic ai in video games. In *Extended Abstracts of the 2020 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '20, page 330–334, New York, NY, USA, 2020d. Association for Computing Machinery. doi: 10.1145/3383668.3419905.

Les A Piegl. B(asic)–spline basics. *Fundamental developments of computer-aided geometric modeling*, 1993.

Jared N Plumb, Sneha Kumar Kasera, and Ryan Stutsman. Hybrid network clusters using common gameplay for massively multiplayer online games. In *Proceedings of the 13th International Conference on the Foundations of Digital Games*, page 2. ACM, 2018.

Robert Porzel, Vanja Cangalovic, Mihai Pomarlan, Sebastian Höffner, Johannes Pfau, John Bateman, and Rainer Malaka. Understanding instructions all the way: A simulation-based approach. In *Proceedings of the 28th International Conference on Computational Linguistics. Under Review*, 2020.

Edward J Powley, Simon Colton, Swen Gaudl, Rob Saunders, and Mark J Nelson. Semi-automated level design via auto-playtesting for handheld casual game creation. In *2016 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 1–8. IEEE, 2016.

Psyonix. *Rocket League*. Game [PC,XboxOne,PS4,Switch], July 2015.

Stefan Radomski and Tim Neubacher. Formal verification of selected game-logic specifications. *on Engineering Interactive Computer Systems with SCXML*, page 30, 2015.

Pramila Rani, Nilanjan Sarkar, and Changchun Liu. Maintaining optimal challenge in computer games through real-time physiological feedback. In *Proceedings of the 11th international conference on human computer interaction*, volume 58, pages 22–27, 2005.

Relic Entertainment. *Company of Heroes 2*. Game [PC], June 2013. Relic Entertainment, Vancouver, Canada. Played 2017.

Scott Rigby and Richard M Ryan. *Glued to games: How video games draw us in and hold us spellbound: How video games draw us in and hold us spellbound*. AbC-CLIo, 2011.

Rosslin John Robles, Sang-Soo Yeo, Young-Deuk Moon, Gilcheol Park, and Seoksoo Kim. Online games and security issues. In *2008 Second International Conference on Future Generation Communication and Networking*, volume 2, pages 145–148. IEEE, 2008.

Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

Richard M Ryan. Control and information in the intrapersonal sphere: An extension of cognitive evaluation theory. *Journal of personality and social psychology*, 43(3):450, 1982.

Richard M Ryan, C Scott Rigby, and Andrew Przybylski. The motivational pull of video games: A self-determination theory approach. *Motivation and emotion*, 30(4):344–360, 2006.

Frederik Schadd, Sander Bakkes, and Pieter Spronck. Opponent modeling in real-time strategy games. In *GAMEON*, pages 61–70, 2007.

Christopher Schaefer, Hyunsook Do, and Brian M Slator. Crushinator: A framework towards game-independent testing. In *Proceedings of the 28th IEEE/ACM International Conference on Automated Software Engineering*, pages 726–729. IEEE Press, 2013.

Jonathan Schaeffer. The history heuristic and alpha-beta search enhancements in practice. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (11):1203–1212, 1989.

Jonathan Schaeffer, Neil Burch, Yngvi Björnsson, Akihiro Kishimoto, Martin Müller, Robert Lake, Paul Lu, and Steve Sutphen. Checkers is solved. *science*, 317(5844):1518–1522, 2007.

Markus Schatten, Igor Tomičić, Bogdan Okreša Đurić, and Nikola Ivković. Towards an agent-based automated testing environment for massively multi-player role playing games. In *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 1149–1154. IEEE, 2017a.

Markus Schatten, Bogdan Okreaša Đurić, Igor Tomičič, and Nikola Ivkovič. Automated mmorpg testing–an agent-based approach. In *International conference on practical applications of agents and multi-agent systems*, pages 359–363. Springer, 2017b.

Ruben Schertler, Simone Kriglstein, and Günter Wallner. User guided movement analysis in games using semantic trajectories. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pages 613–623, 2019.

SethBling. Mari/o - machine learning for video games. June 2015. Retrieved December 12, 2019 from https://www.youtube.com/watch?v=qv6UVOQ0F44.

Mohammad Shaker, Mhd Hasan Sarhan, Ola Al Naameh, Noor Shaker, and Julian Togelius. Automatic generation and analysis of physics-based puzzle games. In *2013 IEEE Conference on Computational Inteligence in Games (CIG)*, pages 1–8. IEEE, 2013.

Manu Sharma, Manish Mehta, Santiago Ontañón, and Ashwin Ram. Player Modeling Evaluation for Interactive Fiction. 2007.

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, and Laurent Sifre et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529:484–489, 2016a.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484, 2016b.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*, 2017.

David Sirlin. Balancing multiplayer competitive games. In *Game Developer's Conference*, 2009.

Adam M. Smith, Chris Lewis, Kenneth Hullet, Gillian Smith, and Anne Sullivan. An Inclusive View of Player Modeling. In *Proceedings of the 6th International Conference on Foundations of Digital Games*, FDG '11, pages 301–303, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0804-5. doi: 10.1145/2159365.2159419. URL `http://doi.acm.org/10.1145/2159365.2159419`. event-place: Bordeaux, France.

BANDAI NAMCO Studios Inc. Sora Ltd. *Super Smash Bros. for Nintendo 3DS / for Wii U*. Game [WiiU,3DS], September 2014. Sora Ltd, BANDAI NAMCO Studios Inc, Tokyo, Japan. Played 2017.

Finnegan Southey, Gang Xiao, Robert C Holte, Mark Trommelen, and John W Buchanan. Semi-automated gameplay analysis by machine learning. In *AIIDE*, pages 123–128, 2005.

Pieter Spronck, Ida Sprinkhuizen-Kuyper, and Eric Postma. Online adaptation of game opponent ai with dynamic scripting. *International Journal of Intelligent Games and Simulation*, 3(1):45–53, 2004.

Shoshannah Tekofsky, Jaap Van Den Herik, Pieter Spronck, and Aske Plaat. Psyops: Personality assessment through gaming behavior. In *In Proceedings of the International Conference on the Foundations of Digital Games*, pages 166–173, 2013.

F. Tence, L. Gaubert, J. Soler, P. De Loor, and C. Buche. Stable growing neural gas: A topology learning algorithm based on player tracking in video games. *Applied Soft Computing*, 13(10):4174–4184, October 2013. ISSN 1568-4946. doi: 10.1016/j.asoc.2013.06.002.

Fabien Tencé, Cédric Buche, Pierre De Loor, and Olivier Marc. The Challenge of Believability in Video Games: Definitions, Agents Models and Imitation Learning. *arXiv:1009.0451 [cs]*, September 2010. arXiv: 1009.0451.

Gerald Tesauro. Td-gammon, a self-teaching backgammon program, achieves master-level play. *Neural computation*, 6(2):215–219, 1994.

Ruck Thawonmas, Masayoshi Kurashige, Keita Iizuka, and Mehmed Kantardzic. Clustering of Online Game Users Based on Their Trails Using Self-organizing Map. In Richard Harper, Matthias Rauterberg, and Marco Combetto, editors, *Entertainment Computing - ICEC 2006*, Lecture Notes in Computer Science, pages 366–369, Berlin, Heidelberg, 2006. Springer. ISBN 978-3-540-45261-4. doi: 10.1007/11872320_51.

Ruck Thawonmas, Yoshitaka Kashifuji, and Kuan-Ta Chen. Detection of mmorpg bots based on behavior analysis. In *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*, pages 91–94. ACM, 2008.

David Thue, Vadim Bulitko, Marcia Spetch, and Eric Wasylishen. Interactive storytelling: A player modelling approach. In *AIIDE*, pages 43–48, 2007a.

David Thue, Vadim Bulitko, Marcia Spetch, and Eric Wasylishen. Learning player preferences to inform delayed authoring. In *AAAI Fall Symposium: Intelligent Narrative Technologies*, pages 159–162, 2007b.

Christian Thurau, Christian Bauckhage, and Gerhard Sagerer. Imitation learning at all levels of game-ai. In *Proceedings of the international conference on computer games, artificial intelligence, design and education*, volume 5, 2004.

Christian Thurau, Tobias Paczian, and Christian Bauckhage. Is bayesian imitation learning the route to believable gamebots. *Proc. GAME-ON North America*, pages 3–9, 2005.

Christian Thurau, Tobias Paczian, Gerhard Sagerer, and Christian Bauckhage. Bayesian imitation learning in game characters. *International journal of intelligent systems technologies and applications*, 2(2):284, 2007.

Julian Togelius, Renzo De Nardi, and Simon M. Lucas. Towards automatic personalised content creation for racing games. In *2007 IEEE Symposium on Computational Intelligence and Games*, pages 252–259, April 2007. doi: 10.1109/CIG.2007.368106. ISSN: 2325-4289.

Julian Togelius, Noor Shaker, and Georgios N. Yannakakis. Active Player Modelling. *arXiv:1312.2936 [cs]*, December 2013. arXiv: 1312.2936.

Simone Tognetti, Maurizio Garbarino, Andrea Bonarini, and Matteo Matteucci. Modeling enjoyment preference from physiological responses in a car racing game. In *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games*, pages 321–328, August 2010. doi: 10.1109/ITW.2010.5593337. ISSN: 2325-4289.

AM Turing. Mind. *Mind*, 59(236):433–460, 1950.

Turn 10 Studios. *Forza Motorsport*. Game [XBox], May 2005. Turn 10 Studios, Redmond, WA, USA.

Iskander Umarov and Maxim Mozgovoy. Creating believable and effective ai agents for games and simulations: Reviews and case study. In *Contemporary Advancements in Information Technology Development in Dynamic Environments*, pages 33–57. IGI Global, 2014.

Valve. *Left 4 Dead*. Game [PC], November 2008. Valve, Bellevue, WA, USA. Played 2017.

Valve. *Counter-Strike: Global Offensive*. Game [PC,PS3,XBox360], August 2012.

Valve. *Dota2*. Game [PC], July 2013.

Marc Van Kreveld, Maarten Löffler, and Paul Mutser. Automated puzzle difficulty estimation. In *2015 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 415–422. IEEE, 2015.

Giel Van Lankveld, Pieter Spronck, and Matthias Rauterberg. Difficulty scaling through incongruity. In *AIIDE*, 2008.

Giel van Lankveld, Pieter Spronck, Jaap van den Herik, and Arnoud Arntz. Games as personality profiling tools. In *2011 IEEE Conference on Computational Intelligence and Games (CIG'11)*, pages 197–202, August 2011. doi: 10.1109/CIG.2011.6032007. ISSN: 2325-4289.

Simon Varvaressos, Kim Lavoie, Sébastien Gaboury, and Sylvain Hallé. Automated bug finding in video games: A case study for runtime monitoring. *Computers in Entertainment (CIE)*, 15(1):1, 2017.

Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575(7782):350–354, 2019.

Georg Volkmar, Johannes Pfau, Rudolf Teise, and Rainer Malaka. Player types and achievements – using adaptive game design to foster intrinsic motivation. In *Extended Abstracts of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*, CHI PLAY '19 Extended Abstracts, page 747–754, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3341215.3356278.

Luis Von Ahn, Ruoran Liu, and Manuel Blum. Peekaboom: a game for locating objects in images. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 55–64. ACM, 2006.

Michael Washburn Jr, Pavithra Sathiyanarayanan, Meiyappan Nagappan, Thomas Zimmermann, and Christian Bird. What went right and what went wrong: an analysis of 155 postmortems from game development. In *Proceedings of the 38th International Conference on Software Engineering Companion*, pages 280–289, 2016.

Ben G Weber and Michael Mateas. A data mining approach to strategy prediction. In *2009 IEEE Symposium on Computational Intelligence and Games*, pages 140–147. IEEE, 2009.

Jiyoung Woo, Hwa Jae Choi, and Huy Kang Kim. An automatic and proactive identity theft detection model in mmorpgs. *Appl. Math*, 6(1S):291S–302S, 2012.

Steven Woodcock. Game ai: the state of the art industry 2000-2001. *Game Developer*, 8(8):36–44, 2001.

Pieter Wouters, Christof Van Nimwegen, Herre Van Oostendorp, and Erik D Van Der Spek. A meta-analysis of the cognitive and motivational effects of serious games. *Journal of educational psychology*, 105(2):249, 2013.

Amir Yahyavi and Bettina Kemme. Peer-to-peer architectures for massively multiplayer online games: A survey. *ACM Computing Surveys (CSUR)*, 46(1):9, 2013.

Jeff Yan and Brian Randell. A systematic classification of cheating in online games. In *Proceedings of 4th ACM SIGCOMM workshop on Network and system support for games*, pages 1–9. ACM, 2005.

Jeff Yan and Brian Randell. An investigation of cheating in online games. *IEEE Security & Privacy*, 7(3): 37–44, 2009.

Geogios N Yannakakis. Game ai revisited. In *Proceedings of the 9th conference on Computing Frontiers*, pages 285–292, 2012.

Georgios N. Yannakakis. Preference learning for affective modeling. In *2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops*, pages 1–6, September 2009. doi: 10.1109/ACII.2009.5349491. ISSN: 2156-8111.

Georgios N. Yannakakis and John Hallam. Entertainment modeling through physiology in physical play. *International Journal of Human-Computer Studies*, 66(10):741–755, October 2008. ISSN 1071-5819. doi: 10.1016/j.ijhcs.2008.06.004.

Georgios N. Yannakakis and Manolis Maragoudakis. Player Modeling Impact on Player's Entertainment in Computer Games. In Liliana Ardissono, Paul Brna, and Antonija Mitrovic, editors, *User Modeling 2005*, Lecture Notes in Computer Science, pages 74–78, Berlin, Heidelberg, 2005. Springer. ISBN 978-3-540-31878-1. doi: 10.1007/11527886_11.

Georgios N. Yannakakis, Manolis Maragoudakis, and John Hallam. Preference Learning for Cognitive Modeling: A Case Study on Entertainment Preferences. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 39(6):1165–1175, November 2009. ISSN 1558-2426. doi: 10.1109/TSMCA.2009.2028152.

Georgios N Yannakakis, Pieter Spronck, Daniele Loiacono, and Elisabeth André. Player modeling. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013.

George Yee, Larry Korba, Ronggong Song, and Ying-Chieh Chen. Towards designing secure online games. In *20th International Conference on Advanced Information Networking and Applications-Volume 1 (AINA'06)*, volume 2, pages 44–48. IEEE, 2006.

Nima Zargham, Johannes Pfau, Tobias Schnackenberg, and Rainer Malaka. Handle with care: Exploring recognition error handling methodologies for speech-based systems. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. Under review.* ACM, 2020.

Dapeng Zhang, Zhongjie Cai, and Bernhard Nebel. Playing tetris using learning by imitation. *Proceedings of GAMEON*, 10:23–27, 2010.

Yan Zheng, Xiaofei Xie, Ting Su, Lei Ma, Jianye Hao, Zhaopeng Meng, Yang Liu, Ruimin Shen, Yinfeng Chen, and Changjie Fan. Wuji: Automatic online combat game testing using evolutionary deep reinforcement learning. In *Proceedings of the 34th ACM/IEEE International Conference on Automated Software Engineering*, 2019.

Alexander Zook, Eric Fruchter, and Mark O Riedl. Automatic playtesting for game parameter tuning via active learning. *arXiv preprint arXiv:1908.01417*, 2019.