Archetypal Game Recommender Systems

Rafet Sifa[1], Christian Bauckhage[1,2], and Anders Drachen[3]

[1] Fraunhofer IAIS, Sankt Augustin, Germany
[2] University of Bonn, Bonn, Germany
[3] Aalborg University, Aalborg, Denmark

Abstract. Contemporary users (players, consumers) of digital games have thousands of products to choose from, which makes finding games that fit their interests challenging. Towards addressing this challenge, in this paper two different formulations of Archetypal Analysis for Top-L recommender tasks using implicit feedback are presented: factor- and neighborhood-oriented models. These form the first application of recommender systems to digital games. Both models are tested on a dataset of 500,000 users of the game distribution platform Steam, covering game ownership and playtime data across more than 3000 games. Compared to four other recommender models (nearest neighbor, two popularity models, random baseline), the archetype based models provide the highest recall rates showing that Archetypal Analysis can be successfully applied for Top-L recommendation purposes.

Keywords: Game Data Mining, Recommender Systems, Behavior Analysis, Predictive Analytics, Business Intelligence, Game Analytics, Player Profiling

1 Introduction

Consumers of digital games are inundated with choices. Thousands of games are produced every year and available across a variety of platforms, from PC, console and mobile units, and across a broad design space. While getting solid numbers on the diversity of games and the number of units shipped is difficult due to the lack of disclosure of sales numbers in the game industry, as of June 2014, the game database site *MobyGames.com* has a total of 84,739 games in its archive. However, only commercial titles are included, and according to [13]: *The sheer amount of product ensures that no source can truly be definitive*. This is exemplified by [17] who in 2012 reported over 222,000 active games on the iOS AppStore alone. According to the same source, thousands of games are submitted to the AppStore every month (see Fig. 1). In terms of the games market, the Gartner Group [8] reported a *93 Billion global market for games (including mobile), pushing upwards of a predicted 100 Billion plus in 2014*. Irrespective of the

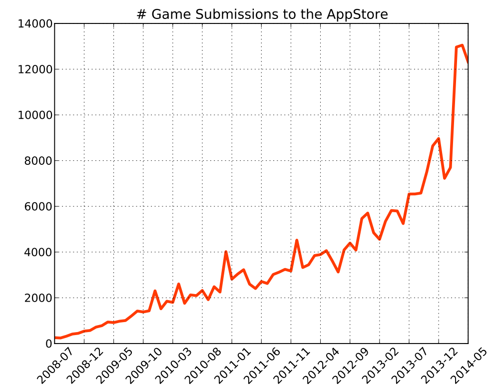

Fig. 1. The number of game applications submitted per month to the iTunes App Store. This is determined by the application release date. Source: www.pocketgamer.biz

specific numbers, there are unprecedented opportunities to meet the needs of a variety of players (consumers) in terms of matching needs with products. According to e.g. [12] this form of recommendation is key to enhance user satisfaction with products. In the context of machine learning, recommender systems are collections of supervised, unsupervised, reinforcement and hybrid learning techniques that aim to predict user's preference or confidence level to a particular set of features based on the previously observed features [11, 10, 14, 18]. In essence, the goal is to provide personalized recommendations that suit the interests of the user in question. Recommender systems are virtually unheard of in digital games beyond basic notions such as popularity rankings. However, recommender systems are a topic of increasing interest, perhaps most importantly due to the recent growth in the number and type of digital games, which has made the issue of information search and selection increasingly serious. Additionally, *user acquisition* costs have increased in the game industry, and with the emergence of Free-to-play (F2P) business models, and the generally low conversion rates of non-paying to paying players, it has become important to devise new strategies to not only acquire but also qualify users, for example via churn analysis [9], and focus acquisition on those channels providing valuable users, e.g. users who are interested in playing the games in question [21]. Recommender systems are perhaps particularly useful for the kind of media products games constitute, because many players will have played the same games, and people often own multiple games [7]. Furthermore, due to the ability to track behavioral telemetry from games, it is possible to obtain implicit information about the level of engagement with a game, for instance, via measuring how much time a given player spends on a game [21]. This means that large-scale data are available on the appeal of specific games to specific customers. Implicit feedback can also be combined with explicit feedback from e.g. user rating systems.

In the background of this interest is a series of changes the game industry has gone through in recent years, which has added new sources of information about users and -behavior which can be used in recommender systems. These changes include the rapid rise of mobile games to become a substantial part of the marketplace [7], and a break from traditional retail-based business models which are today supplemented with notably F2P models. However, despite a remarkable growth in the number of games and the rapid emergence and advancement of business intelligence methods in game development collectively referred to as *game analytics* [21], the majority of the research on user (player) behavior in digital games has been confined to single games, limiting the broader application of the results, and the knowledge about practices in the industry is limited due to confidentiality issues [21]. Thus, in this paper, to the best knowledge of the authors, the first recommender system developed specifically for digital games is presented.

1.1 Related Work

Over the past few years the domain of game analytics has emerged to support the increasing needs for business intelligence solutions in the game industry [21]. While previous and current work studies touch various aspects of data mining for games, to the best knowledge of the authors there is no prior work done towards player/user based recommender systems for games. Recommender systems form a major topic of investigation outside the games domain, for example the e-commerce movement led by Amazon.com, or for the recommendation of movies, music and physical products [11, 12, 18, 14, 4]. In the below sections, the parts from existing relevant work on recommender systems and their evaluations is explained and related to on an ongoing basis, however, it is also valuable to briefly consider some of the work on large-scale data mining in digital games.

Cross-games analytics is a rare occurrence, in part due to the recent rapid emergence of the practice in the industry, the confidentiality associated with behavioral data and the lack of public datasets. Exceptions exist, such as Bauckhage et al. [1] and Chambers et al. [3], both focusing on analyzing playtime distributions. Within network analysis, an example is Pittman and GauthierDickey [16] who investigated player distribution in the two online games World of Warcraft and Warhammer Online. While analyzing in-game spatial data, Bauckhage et al. [2] propose methods to categorize spatial player behavior which could be later used to learn the transitions between game sectors. Drachen et al. [6] used clustering methods to create behavioral gameplay profiles from a First Person Shooter (FPS) game and a MMOG. Bauckhage et al. [1] observed specific patterns in the playtime distribution across five major commercial game titles, and presented an explanation for why the Weibull distribution model provides good fits on various aspects of player behavior (playtime, session frequency, session length, inter-session time). Following this study, Sifa et al. [23] documented the same patterns for over 3000 games, and furthermore noted the presence of groups of games featuring similar aggregate playtime profiles. A major topic of interest in the game industry is player churn analysis. Having analyzed five F2P mobile game players, Hadiji et al. [9] proposed models to predict player departure, i.e. churn, in order to allow the developers or the studios to take precautions to prevent the players from churning. Runge et al. [19] focused on two F2P games, presenting another method for churn modeling.

1.2 Contribution

This paper takes a step towards addressing the need for recommender systems specialized for games. The first contribution of the work presented two different formulations of archetypal analysis for Top-L recommender task using implicit feedback. These form the first application of recommender systems to digital games. The two models comprise: 1) a factor based model aiming to impute missing values; 2) user-based neighborhood oriented model operating in reduced dimensions. Both systems are evaluated using off-line evaluations following a similar procedure to [4]. The evaluation approach evaluates hit recall ratio that is based on the ranking assigned to a blinded game selected randomly among the games the player in question spends the most of their time on and 100 randomly selected games. The evaluation is then based on where the blinded game occurs in the ranked recommendations. Using our models, we obtain up to 86% recall when predicting mostly played games for players in the cases of Top-5 recommendation and over 97% recall in the cases of Top-30 recommendation.

2 Steam, Data and Pre-processing

Steam is an online, cross-platform game distribution system, with around 75 million active users, about 172 million accounts total, hosting over 3000 games, which makes it an ideal platform for the type of work presented here. The dataset used here was harvested by [23] from public Steam profiles, using the API client provided by Valve. The same dataset is used here to have consistent and comparable results to our previous work. The dataset contains records from over 3200 games and applications, but after running through the preprocessing steps detailed below, the dataset was constrained to 3,007 full games and 6,049,520 Steam players, covering 5,068,434,399 hours of game-play.

The public profiles and the corresponding players were selected form the most populous 3500 communities on Steam, and their IDs are anonymized by random hashing. The data was harvested in the Spring of 2014 and contains information on playtime for the games owned by the user. For some players, the dataset may not cover their full player histories (i.e. still active players), and may bias results towards showing shorter playtimes than they actually are. It is also important to note that the tracking of player behavior in the Steam platform started after March 2009[4], which eliminates the playtime of the users before this time. A series of preprocessing steps have been performed: all game demos and Software Development Kits (SDKs) were removed, as were games not played by

[4] http://forums.steampowered.com/forums/showthread.php?p=10247483

at least 25 people. Furthermore, there was a small set of games with no playtime information, i.e. games that do not save the information about whether it has been downloaded and not played. These games were also eliminated. It is also important to note that the dataset only covers playtime on the Steam platform. We tested our recommender models that predict the mostly played games using a sample of players of the game Warframe[5], a F2P combat game (3rd most played F2P game on Steam). Warframe was randomly selected from the 25 most played games in the dataset, to ensure a sufficiently large sample size of players: 772,068 of the 6 million people in the dataset has played Warframe, leading to a combined playtime of over 22 million hours. The game is furthermore a good candidate for the work presented here because it, by the time of writing, is only about one year old, which means there is theoretically less chance of tracking corruption as compared to older games. Of the Warframe players, over 540,000 have played at least 5 games and 100,000 players were randomly sampled from this pool to evaluate our models.

3 Archetypal Analysis

In this section we briefly introduce Archetypal Analysis and it's properties. Archetypal Analysis [5] is a matrix decomposition technique based on decomposing the given matrix into a collection of extreme entities, called *archetypes*, and stochastic coefficient vectors to represent each data point as a convex combination of the found archetypes. Archetypal Analysis has been extensively used in the gamemining research to cluster players [6, 24], analyze social group activities [25], group games based on gameplay-interest models [23] and to generate human-like game bots [22]. Formally given an m dimensional data matrix with n data points as $\mathbf{X} \in \mathbb{R}^{m \times n}$ and an integer k, archetypal analysis finds k archetypes $\mathbf{Z} \in \mathbb{R}^{m \times k}$ and a non-negative column stochastic coefficient vectors $\mathbf{A} \in \mathbb{R}^{k \times n}$, that satisfy

$$||\mathbf{a}_j||_1 = 1 \ \forall j \in [1, 2, \ldots, n]. \tag{1}$$

Having found the appropriate archetypes, the data points can be represented as convex combinations of the archetypes as

$$\mathbf{x}_j \approx \mathbf{Z}\mathbf{a}_j = \sum_{i=1}^{k} \mathbf{z}_i a_{ij}. \tag{2}$$

Additionally, in [5] archetypes were defined as convex combinations of the actual data points which can be represented as

$$\mathbf{z}_i = \mathbf{X}\mathbf{b}_i = \sum_{j=1}^{n} \mathbf{x}_j b_{ji} \tag{3}$$

[5] http://store.steampowered.com/app/230410/

where $\mathbf{B} \in \mathbb{R}^{n \times k}$ are non-negative column stochastic coefficient vectors satisfying the following

$$||\mathbf{b}_i||_1 = 1 \; \forall i \in [1, 2, \ldots, k]. \tag{4}$$

Representing this as matrix reconstruction problem, Archetypal Analysis finds the optimal archetypes and the coefficient matrices to reduce the representation error that can be quantified by the Frobenius Norm as

$$\left\|\mathbf{X} - \mathbf{XBA}\right\|_F = \left\|\mathbf{X} - \mathbf{ZA}\right\|_F = \left\|\mathbf{E}\right\|_F = \sqrt{\sum_{u=1}^{n}\sum_{y=1}^{m} |e_{uy}|}. \tag{5}$$

Identifying the possible values of k, Cutler and Breiman [5] show that for $k = 1$ the minimizer of (5) is the data mean, whereas for values between 1 and n, i.e. $1 < k < n$, the archetypes are in the data convex hull and finally, for the case where $k = n$, having the data elements as archetypes is the global minimizer of (5). Various methods have been studied to find archetypes that reduce (5) by keeping, relaxing or strengthening the above constraints. Cutler and Breiman [5] have proposed an alternating algorithm that solves convex least squares problems to find optimal matrices $\mathbf{A}$ and $\mathbf{B}$. Thurau et al. [26] increase the speed of finding the archetypes by constraining the archetypes to be lying in the data convex hull. Restricting the archetypes to be data points, Thurau et al. [27] found the archetypes that maximize the volume of the data simplex. For a detailed explanation of archetypal analysis and it's applications we refer the reader to [5, 26, 27].

4 Archetypal Top-L Recommender Systems

In this section we describe how Archetypal Analysis can be used to recommend $L \in \mathbb{N}$ items to users given that we know which items they own. That is, we will describe two different data representation and recommendation approaches to increase recommendation accuracy. As the main intention of recommender engines is to provide the users useful recommendations that are personalized according to their tastes, our scenario is based on proposing the user, called *active user*, a list of games, denoted by $\mathcal{L}$, that they might be interested in. Formally, representing the playtime data as a game-player matrix $\mathbf{T} \in \mathbb{R}^{m \times n}$, where we have m games and n players, our aim is to come up with a list of games that a player i have not yet played and might be interested in playing, based on the games they have played, which we group under the set $\mathcal{D} = [j \; | j \; \forall \; t_{ji} > 0]$. It is important to note that the solutions we provide here are inclined to fully observed datasets, where the zero values in the data matrix are not treated as missing values. This is an important aspect to distinguish when we analyze implicit playing behavior measures, such as playtime or login time distributions, number of in-app purchases, number of friends and so on. In fact, unlike the explicit movie ratings, this sensory, or telemetry, data represents the confidence of the user to the particular attribute rather than being a direct indicator of interest [1, 23, 10].

4.1 Factor Oriented Model

The first recommendation model we propose in this work is a factor based model that aims to impute the missing value by refactorization. Namely, having selected an integer k and factorized the playtime matrix $\mathbf{T}$ using Archetypal Analysis we obtain:

$$\mathbf{T} \approx \mathbf{G}^T \mathbf{P}. \tag{6}$$

In this equation $\mathbf{G}^T$ represents the extreme game player profiles that form the *game factor matrix* and $\mathbf{P}$, the *player factor matrix*, contains the stochastic coefficient vectors that are used to represent each player as a convex combination of the extreme entities in $\mathbf{G}^T$. Prediction at this point is done similar to the other latent factor models by multiplying the respective (active) player and game factors. Namely the estimation of association between game j and player i is found as

$$\hat{t}_{ji} = \mathbf{g}_j \mathbf{p}_i = \sum_{u=1}^{k} g_{uj} p_{ui}. \tag{7}$$

As a final step, as shown in (8), our main aim is now to find the top L games for the player i that have the highest estimated values of the associations.

$$\mathcal{L} = \operatorname*{argmax}_{l \notin D} \hat{t}_{li} \tag{8}$$

Compared to the neighborhood oriented models, the factor model provides a very fast and a scalable way to make recommendations, as we do not have to calculate similarities between the available users, but directly multiply the corresponding two factor vectors of the active player and the active game.

4.2 Neighborhood Oriented Models

Another use of Archetypal Analysis for Top-L recommender systems is through neighborhood methods in the reduced dimensional spaces. Having found the appropriate archetypes and coefficient vectors, for each user, we can make recommendation based on neighboring items or users in the calculated coefficient space. An obvious advantage of this set of methods is the representation of the players or the games in the reduced dimensions, k-simplex, which reduces the neighborhood calculation time significantly, as $k \ll n$. Initially, we factorize the player-game matrix as in (6), where we obtain the simplex embedding of the players in the matrix $\mathbf{P}$.

At each recommendation step where we are required to recommend L games to the user i, we first find the most similar players to player i that maximize the similarity, or minimize the distance, in the reduced dimensional player factor space represented by $\mathbf{P}$. Variety of distance metrics can be used to find similar entities in the reduced dimensional space those include, cosine similarity, Pearson's correlation index or entropy based similarity measures. Throughout our experiments we used cosine similarity measure to calculate the similarities between users as it provided us a fast and efficient solution through vectorized operations. Cosine similarity is an angular similarity measure, that is based on finding the cosine value of the angle between two vectors. We can calculate the Cosine similarity between two vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ by finding the dot product between the vectors after normalizing them as in

$$sim(\mathbf{v}_1, \mathbf{v}_2) = \frac{\mathbf{v}_1 \mathbf{v}_2}{||\mathbf{v}_1|| \, ||\mathbf{v}_2||}. \tag{9}$$

Having selected a similarity measure, which we denote as sim, the estimation of the association for recommending any game j can be calculated as a weighted average between the values of the closest (with respect to distance in the reduced space) U players grouped in $\mathcal{U}$. Formally, we find the estimation of the association between player i and game j as

$$\hat{t}_{ji} = \frac{\sum_{u \in \mathcal{U}} sim(\mathbf{p}_i, \mathbf{p}_u) \, \mathbf{t}_{ju}}{\sum_{u \in \mathcal{U}} sim(\mathbf{p}_i, \mathbf{p}_u)}. \tag{10}$$

Having calculated the estimation for the association values, the top L games are recommended by returning the ones that have the highest predicted playtime association values, as done in (8).

5 Evaluation and Results

Variety of methods in the recommender systems literature have been proposed to evaluate recommender systems [4, 12, 11]. In this study we follow a user based off-line evaluation where we randomly blind one of the mostly played games for each player to form a test set and use the rest of the data set to train our models. We follow an evaluation method similar to [4], that is, our aim is to evaluate the hit recall based on the ranking assigned to the blinded (active) games and randomly selected 100 games for each test case. The addition of the randomly selected games to the evaluation process is made to observe that, given the history of the active player, if the system returns the active game to the user, i.e. if a high association value is assigned to the very game. An ideal recall hit, 100%, will be achieved if all of the blinded games in the test set are returned, or ranked, in the recommended top-L list, and oppositely, 0 % recall will be achieved if non of the games in the test set are returned in any of the recommendation steps. Therefore, for each testing instance we calculate the association between the active player and the 101 games and find out if the recommended top L games contain the blinded game to get a hit. We repeat the same procedure for different numbers of returned games, i.e. top-L games, and report the hit ratio for each L value to obtain the recall curves. We preferred to solely use recall based evaluation so as to simulate the specif game recommendation scenarios where the player is offered to purchase or download a collection of games that are presented without any particular order.

We have tested the factor oriented and the player based neighborhood oriented methods (which we named **AAF** and **AANeP** respectively) to predict the games that are likely to be played by each Steam player for a relatively long time. Fig. 2 shows the recommendation results from our models with different parametrization where the recall values at the y-axis represents the hit ratio of the sought game and the x-axis represents the number of recommended games.

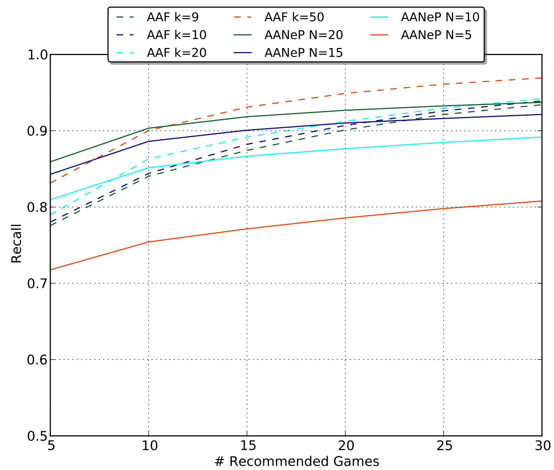

Fig. 2. Top-L Game recommendation results for different parametrization of **AAF** and **AANeP**. The recall value at the y-axis represents the hit ratio of the sought game and the x-axis represents the number of recommended games. Best seen in color.

So as to compare our models with other recommender systems, we used the following four recommendation methods:

- **Rand**: A random recommender that recommends L randomly selected (from uniform distribution) games to each user (among the games that are not played by the user)
- **PurPop**: A popularity based recommender system that recommends users unplayed games that are sorted with respect to the global frequency of the ownership (for free-to-play games) or purchases (for paid games)
- **TimePop**: A popularity based recommender system that recommends users unplayed games that are sorted with respect to the total playtime of the games
- **NN**: An item-based nearest neighbor recommender that estimates player-game associations based on player's playtime spent on similar games.

The first model is introduced to show how a random model would behave for top-L recommender systems, similar to [4], the following two popularity based recommender models are used to show the effect of global recommendations to find the mostly played games. Finally the last method is used to provide yet another baseline classifier as it has been used commonly in the recommender

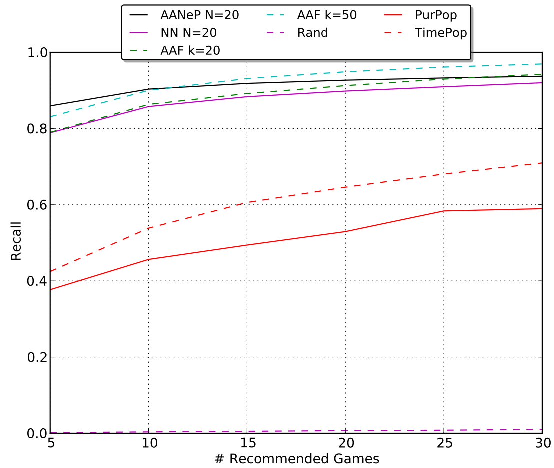

Fig. 3. Top-L game recommendation results for recommending the mostly played games in individual basis. Archetypal Analysis based recommender models performed the best among the methods used in the experiments. Best seen in color

systems literature [20, 10]. Figure 3 illustrates the recall values for all of the tested methods in a single picture where Archetypal Analysis based recommender models performed the best among the methods used in the experiments. We can see that random recommender systems perform poorly when used for top-L recommendation tasks. Additionally, unlike the above random recommender, the results of the popularity based recommender systems show that there are trends for playing games in the steam platform, which indicates, recommending the most popular games increases the recall rates over 50%. Analyzing the results of Archetypal recommender systems we can observe that user based neighborhood oriented method and the factor methods give the best prediction performances among the tested methods reaching up to 86% recall for Top-5 recommendations (for AANeP) and over 97% recall for Top-30 recommendations (for AAF).

6 Discussion and Conclusion

With the steadily increasing number of available digital games, recommender systems have started to gain the interest of the game industry as the issues surrounding information search and selection are becoming increasingly serious. This not only from the viewpoint of the users, who need to find the right games, but also from the perspective of the game developers, who due to high and increasing user acquisition costs, especially for F2P games [21], have a direct need to optimize strategies to find the right users for their games.

In this paper we propose two different recommender methods for Top-L recommender tasks, that are based on archetypal analysis [5] and use implicit feedback via data on playtime and game ownership. It is to the best knowledge of the authors the first application of recommender systems to digital games. The two models are evaluated against a dataset of 500,000 users of the Steam digital game distribution platform, covering more than 3000 games. For L values of 5, recall rates of 86% are reached for AANeP and 84% for AAF, which outperforms the other four recommender models we tested for evaluation. This shows that archetypal analysis can be used for Top-L recommendation purposes. While multiple values of L are tested and results presented in Fig. 3, the L value of 5 is highlighted here as it forms the lower boundary of the classical 7 ± 2 rule from user interface design [15]. The rule basically states that human short term memory can normally handle 7 ± 2 chunks of information. This means that chunking information can increase the short-term memory capacity, but it also means that showing a user for example 10 images of recommended games (for example using box art), will not allow the user to identify these with a glance. On the contrary, if short-term identification is a goal, a maximum of 5 recommended games at a time would appear to be a safer option. This in turn means that there may be an interest in optimizing recommender systems for games towards L values that fall within the 7 ± 2 rule.

The goal here was to use playtime information across all games played by the users on the Steam platform, to predict a game that the player would play for a long time. This is just one potential goal of game recommender systems. Other goals, that will be investigated in future work, include finding players who will spend a lot of money on in-game purchases or have a high social value. Our future work on game recommender systems will also focus on increasing the size of the sample used here. Additionally, features such as social networking impacts will be investigated. Finally, a practical requirement of game recommender systems when it comes to their adoption by the game industry will be the speed at which they can be executed for large sample sizes. While the models used here are relatively fast to compute, optimization forms another venue for future research.

Acknowledgment

The work reported in this paper was carried out within the Fraunhofer / University of Southampton research project *SoFWIReD*. The authors gratefully acknowledge this support.

References

1. Bauckhage, C., Kersting, K., Sifa, R., Thurau, C., Drachen, A., Canossa, A.: How Players Lose Interest in Playing a Game: An Empirical Study Based on Distributions of Total Playing Times. In: Proc. IEEE CIG (2012)
2. Bauckhage, C., Sifa, R., Drachen, A., Thurau, C., Hadiji, F.: Beyond Heatmaps: Spatio-Temporal Clustering using Behavior-Based Partitioning of Game Levels. In: Proc. IEEE CIG (2014)
3. Chambers, C., F.W.S.S., Saha, D.: Measurement-based Characterization of a Collection of On-line Games. In: Proc. of ACM SIGCOMM Conf. on Internet Measurement (2005)

4. Cremonesi, P., Koren, Y., Turrin, R.: Performance of Recommender Algorithms on Top-N Recommendation Tasks. In: Proc. ACM Recsys (2010)
5. Cutler, A., Breiman, L.: Archetypal Analysis. Technometrics 36(4), 338–347 (1994)
6. Drachen, A., Sifa, R., Bauckhage, C., Thurau, C.: Guns, swords and data: Clustering of player behavior in computer games in the wild. In: Proc. IEEE CIG (2012)
7. Entertainment Software Association: Essential Facts About the Computer and Video Game Industry (2014), `http://www.theesa.com/facts/pdfs/esa_ef_2014.pdf`
8. Gartner Group: Gartner Says Worldwide Video Game Market to Total $93 Billion in 2013 (2013), `http://www.gartner.com/newsroom/id/2614915`
9. Hadiji, F., Sifa, R., Drachen, A., Thurau, C., Kersting, K., Bauckhage, C.: Predicting Player Churn in the Wild. In: Proc. IEEE CIG (2014)
10. Hu, Y., Koren, Y., Volinsky, C.: Collaborative Filtering for Implicit Feedback Datasets. In: Proc. IEEE Int. Conf. Data Mining, ICDM. IEEE (2008)
11. Kantor, P., Rokach, L., Ricci, F., Shapira, B.: Recommender Systems Handbook. Springer (2011)
12. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer 42(8), 30–37 (2009)
13. Laratta, D.: The Death of the Video Game Expert (2010), `http://venturebeat.com/2010/04/08/the-death-of-the-video-game-expert/`
14. Linden, G., Smith, B., York, J.: Amazon.com Recommendations: Item-to-item Collaborative Filtering. IEEE Internet Computing 7(1), 76–80 (2003)
15. Miller, G.A.: The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity to Process Information. Psychological Review 63(2), 81–97 (1956)
16. Pittman, D., GauthierDickey, C.: Characterizing Virtual Populations in Massively Multiplayer Oline Role-playing Games. In: Proc. of Int. Conf. on Advances in Multimedia Modeling (2010)
17. PocketGamer.biz: App Store Metrics (2012), `http://www.pocketgamer.biz/metrics/app-store/categories/`
18. Resnick, P., Varian, H.: Recommender Systems. Communications of the ACM 40(3), 56–58 (1997)
19. Runge, J., Gao, P., Garcin, F., Faltings, B.: Churn Prediction for High-value Players in Casual Social Games. In: Proc. IEEE CIG (2014)
20. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based Collaborative Filtering Recommendation Algorithms. In: Proc. ACM WWW (2001)
21. Seif El-Nasr, M., Drachen, A., Canossa, A.: Game Analytics: Maximizing the Value of Player Data. Springer (2013)
22. Sifa, R., Bauckhage, C.: Archetypical Motion: Supervised Behavior Learning Using Archetypal Analysis. In: Proc. IEEE CIG (2013)
23. Sifa, R., Bauckhage, C., Drachen, A.: The Playtime Principle: Large-scale Cross-games Interest Modeling. In: Proc. IEEE CIG (2014)
24. Sifa, R., Drachen, A., Bauckhage, C., Thurau, C., Canossa, A.: Behavior Evolution in Tomb Raider Underworld. In: Proc. IEEE CIG (2013)
25. Thurau, C., Bauckhage, C.: Analyzing the Evolution of Social Groups in World of Warcraft. In: Proc. IEEE CIG (2010)
26. Thurau, C., Kersting, K., Bauckhage, C.: Convex Non-negative Matrix Factorization in the Wild. In: Proc. ICDM (2009)
27. Thurau, C., Kersting, K., Bauckhage, C.: Yes We Can: Simplex Volume Maximization for Descriptive Web-Scale Matrix Factorization. In: Proc. ACM CIKM (2010)