Controlling the Crucible: A Novel PvP Recommender Systems Framework for *Destiny*

Rafet Sifa
Fraunhofer IAIS
Sankt Augustin, Germany
rafet.sifa@iais.fraunhofer.de

Eric Pawlakos
Northwestern University
Evanston, IL, USA
ericpawlakos2017@u.northwestern.edu

Kevin Zhai
Northwestern University
Evanston, IL, USA
kevinzhai2017@u.northwestern.edu

Sai Haran
Northwestern University
Evanston, IL, USA
saiharan2017@u.northwestern.edu

Rohan Jha
Northwestern University
Evanston, IL, USA
rohanjha2017@u.northwestern.edu

Diego Klabjan
Northwestern University
Evanston, IL, USA
d-klabjan@northwestern.edu

Anders Drachen
Digital Creativity Labs, University of York
York, United Kingdom
anders.drachen@york.ac.uk

ABSTRACT

Compared to conventional retail games, today's Massively Multiplayer Online Games (MMOGs) have become progressively more complex and volatile, living in a highly competitive market. Consumable resources in such games are nearly unlimited, making decisions to improve levels of engagement more challenging. Intelligent information filtering methods here can help players make smarter decisions, thereby improving performance, increasing level of engagement, and reducing the likelihood of early departure. In this paper, a novel approach towards building a hybrid multi-profile based recommender system for player-versus-player (PvP) content in the MMOG *Destiny* is presented. The framework groups the players based on three distinct traced behavioral aspects: base stats, cooldown stats, and weapon playstyle. Different combinations of these profiles are considered to make behavioral recommendations. An online evaluation was performed to investigate the usefulness of the proposed recommender framework to players of *Destiny*.

CCS CONCEPTS

• **Computing methodologies → Supervised learning by classification**; • **Applied computing → Computer games**;

KEYWORDS

Destiny, profiling, behavior, game analytics, recommender systems, recommendation, MMOG, MMOFPS

ACM Reference Format:
Rafet Sifa, Eric Pawlakos, Kevin Zhai, Sai Haran, Rohan Jha, Diego Klabjan, and Anders Drachen. 2018. Controlling the Crucible: A Novel PvP Recommender Systems Framework for *Destiny*. In *Proceedings of Australasian Computer Science Week 2018 (ACSW 2018)*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3167918.3167926

1 INTRODUCTION

Massively Multiplayer Online Games (MMOGs) have become increasingly more complex as gaming culture and technology mature. MMOGs are constantly introducing new gameplay features and updates, leading to an environment where players have a considerable number of choices about how to play the game. This can make it challenging for players to understand specific ways in which to improve their skills. Improving can be defined differently based on the genre of game and goals of the player, ranging from raising kill-death ratios in the online first person shooter game *Counter-Strike* to scoring higher damage per second in the Multi-player Online Battle Arena (MOBA) game *League of Legends*. A recommender system built for these types of environments would impact how players think about their gameplay and

ACSW 2018, January 29-February 2, 2018, Brisbane, QLD, Australia

ACM ISBN 978-1-4503-5436-3/18/01.
https://doi.org/10.1145/3167918.3167926

might allow them to engage more with the games. These systems are not only useful for players, but for game developers as well. For persistent online games and MMOGs like *Destiny* that are constantly updated, commercial success rests on the game's ability to keep a community engaged for long periods of time. Having an accurate recommendation system advising players on how to improve can create more incentive to continue playing, since players understand their goals and how to achieve them better [10, 19].

In this paper, a multi-profile recommendation framework is introduced to address the unique properties of the gaming domain, specifically for the MMOG, or more specifically Massively Multi-Player Online First Person Shooter (MMOFPS) *Destiny*. *Destiny* is fundamentally a MMOG but the core gameplay revolves around first-person combat and there are other differences from the traditional MMOG format, such as the limited socialization tools.

In MMOFPSs like *Destiny*, players typically have an array of choices in how to build their character and which weapons to use. Given this array of options, there is a largely untapped opportunity for adding recommender systems to assist players with advice on these choices. This is notably the case for those players who are not aware of the *metagame*. The metagame of a game is the ongoing formation of strategies and tactics towards maximizing success chances. In eSports games as well as persistent online titles like *Destiny*, the metagame changes as players invent new strategies. The metagame also commonly changes around the release of new patches or updates, which can alter the balance of the game. Metagame in *Destiny* commonly refers to discussions around the best character builds, skills, weapon modification and strategies. A robust recommender for guns would not merely suggest the guns that dominate the metagame, but rather the guns that uniquely fit the playstyle of the individual.

Robust recommendation systems for MMOGs have not been thoroughly explored previously outside of industry work on pricing of virtual items or systems for recommending new games to players. Existing recommender approaches such as traditional collaborative filtering are not enough for the MMOG setting, since consideration needs to be given to a variety of different metrics and player preferences. A player who primarily favors shotguns would not benefit from a recommender suggesting sniper rifles. Therefore, a hybrid recommender system is proposed to address these unique problems. The purpose of the system proposed here is to help players perform better, as well as give them inspiration to try out variations on builds they are already familiar with. The framework presented provides flexible recommendations on multiple aspects of the game and can in principle be migrated to games of similar complexity, especially games in the eSports space.

2 RELATED WORK

The review of the state-of-the-art across behavioral profiling and recommender systems in games will be limited to the key references in the field of Game Analytics, which is the specialized domain of Business Intelligence that specifically focuses on games [10].

Over the past few years, Game Analytics has emerged as a core topic in game design and research, forming a core component of game development today [10]. Behavioral telemetry in major commercial game titles are of large volume, highly varied and typically volatile [3, 7, 8, 16–18, 23]. This is exemplified by *Destiny*, whose back end telemetry servers host over a thousand features for each player, including a daily summary of their performance in the game [6].

Developing behavioral profiles in modern game development can be challenging. However, it creates great benefit by forming condensed, actionable views of the behavior of the player base, which can inform design, track problems, assist matchmaking, and identify players groups with specific characteristics. Drachen et al. [6] developed behavioral profiles for a set of 10,000 players of *Destiny*, focusing on discovering the best performing cluster models for the task of handling high-dimensional behavioral clustering. Working with a set of 4,800 randomly selected players and 41 performance-focused features across four cluster models. The authors concluded that Archetypal Analysis [5] performs best in terms of developing clearly separated and explainable profiles, the latter forming a key quality criteria in games-based behavioral profiling as argued by Drachen et al. [9]. Working with Self-Organizing Networks from a 1,365 player sample from *Tomb Raider: Underworld*, Drachen et al. [9] note how over 95% of the players could be categorized into one of four behavioral profiles. Analyzing a larger sample from the same game, Sifa et al. [17] investigate level-wise behavioral patterns using Archetypal Analysis. Shim and Srivastava [14] utilized segmentation and description to evaluate player behaviors in *EverQuest II*, whereas Thawonmas and Iizuka [21] generated visualizations of player clusters developed using multi-dimensional scaling for the massively multi-player online game (MMOG) *Shen Zhou Online*. Drachen et al. [8] evaluated the fitness of Simplex Volume Maximization (SIVM) and k-means for profiling on data from the MMOG *Tera: Online* and the online team shooter *Battlefield 2*.

While the state-of-the-art of Game Analytics is advancing rapidly, the topic of applying recommender systems in games remains relatively unexplored. Recommender systems initially saw use in games with the focus on training and assisting game AI and are relatively well explored in games for that purpose [23]. However, research on systems for recommending products or behaviors to users are comparatively rare. As the first major academic-based inroads towards using recommender systems, Sifa et al. [15] focused on recommending game titles to players based on the games they had played previously, introducing an Archetypal Analysis [5] based recommender system for game recommendation across a 3000+ game dataset from the game distribution platform Steam. Around the same time, Valve, the company behind Steam, introduced a recommender system to their storefront (the two projects being unrelated). The work focused on recommending games, similar to movie recommendations on platforms such as Netflix or app recommendations on the AppStore [4, 11]. Similarly, Anwar et al. [1] used collaborative filtering to suggest games to players via evaluating the opinions of similar players. Notably, the system was evaluated via a live player sample, an approach that is also adopted here. In addition to recommending which games to play, recommender systems can also be used to recommend behaviors to players during play or which items to buy. An industry case study focused on the latter was described by Weber [22].

3 DESTINY: AN OPEN WORLD MMOFPS GAME

Destiny is a mythic, science-fiction themed online first-person shooter set 700 years in the future. Players assume the role of "Guardians", superpowered beings who defend the last remnants of humanity. *Destiny* is, above all else, an online first-person shooter (FPS) which draws from Bungie's earlier *Halo* series. Most of the game revolves around a player-controlled character using several of the hundreds of weapons available to kill other players or computer-controlled enemies. However, it also incorporates elements from MMOGs such as *World of Warcraft*, which emphasize a social and cooperative element of gameplay, as well as a strong focus on collecting new weapons, armor, and items. Destiny offers both player vs. environment (PvE) and player vs. player (PvP) game modes. PvE game modes allow the player to patrol various planets and attempt solo missions, as well as tackle cooperative missions known as "strikes" and "raids".

Containing role-playing elements, Destiny offers a wide variety of customization options. Starting with character initialization, a player may choose to be male or female, one of three races, and one of three classes (Titan, Hunter, or Warlock). These choices are mostly cosmetic, having no inherent impact on a player's performance. Our recommender focuses on three customizable areas that greatly affect performance: weapon playstyle/weapon class, base stats, and cooldown stats. Weapon class is the kind of weapon a player chooses to use, ranging from shotguns to hand cannons. Furthermore, each class contains a "skill tree" which lets players choose special abilities to augment their agility, armor, and recovery, collectively known as base stats. Respectively, base stats affect how fast a player's character moves, how much damage they can take, and how quickly they can recover health when damaged. Finally, the type of armor equipped may alter strength, discipline, and intelligence. These are the cooldown stats since they affect how often a player may use melees, grenades, and superpowers respectively.

The core of our analysis is centered on PvP, occurring in a game mode called the Crucible. Given the closely competitive nature of Crucible matches, players work to build their character in a way that suits their preferences, such as equipping more powerful weapons and armor or changing their character's base and cooldown stat allocations. Given the vast amount of variability in how players choose to play, it is important to consider several aspects of the gameplay when offering a recommendation, rather than homing in on only one. A player may not be interested in a recommendation to change their weapon, but would enjoy advice on which stat allocation to choose, or vice versa. The multi-profile recommendation framework that is proposed here aims to address this challenge of inherent player preferences in gaming recommendations.

4 DATA AND PRE-PROCESSING

The datasets that are generated are based on a random sample of 10,000 players from the available pool of total players as of September 2016. The sample was extracted by Bungie, the developer of *Destiny*, to ensure random sampling. The only requirement placed on the players, was that any given player had to have played the game for more than 2 hours, to avoid players who did not get past the introduction tutorial. The sample is thus large enough to enable inference to the population of players; however, population-level analysis is not the focus of the current work. Via the Bungie API, data about player behavior were extracted and stored as large JSON files. These datasets were pulled during *The Taken King* expansion, released on September 15, 2015. Parallel extractions were performed to ensure initial data quality. While only a subset of players are used here, it is important to note that *Destiny* passed 30 million active players in 2016 [12], and has been running since 2014, which means that the dataset from the game is of substantial scale by now. Furthermore, it is important to note that any profiles generated in the game are by their nature of limited shelflife as accurate representations of the players, since *Destiny* is constantly patched and updated. However, as a testing base for recommender systems, the dataset is highly representative and varied.

First, data was extracted for the Crucible matches of the players, providing a total of 930,000 matches. Each time a player enters a PvP match, Bungie tracks information about that player in addition to any other players in the match. Within PvP matches, Bungie is primarily collecting "performance" data. Performance data gives us information on how the player behaved and what they did during the match.

This includes metrics related to their score (kills, deaths, assists, total points, etc.) and metrics related to their behavior (amount of kills with a specific weapon, which weapons they used, their average time alive per life, etc.). In total, Bungie tracks 46 metrics for each player in a match. Any metric not related directly to a player's PvP performance, such as the match ID, was removed from the analysis. Within a PvP match, a player can kill an opponent in 15 separate ways, based on the class of the weapon used. Every individual weapon belongs to one of the 15 weapon classes. The kills earned by each player with each of the weapon classes were converted into proportions. By doing so, the issue of players having different number of matches and number of kills is avoided. Proportions also give us more information about a player's preferred weapon class overall. The usage of specific weapons per player was aggregated in order to find a given player's most used weapons. After parsing all the matches the aggregated dataset consisted of 8,873 characters and 38 features.

A second component of the dataset covers the core information about each of the 10,000 players. This includes aggregated information on almost everything related to the player's characters. This includes a player's appearance, gear, level, weapons, and much more. It is important to note that this information was aggregated across the lifetime of a player. As such, this is a "snapshot" of the player's current status at the time the data was pulled. Within this dataset, the most relevant information was in the "base stats" and the "cooldown stats" of the players. A more detailed explanation of what these stats are is included in the feature definitions. Since these stats affect various aspects of combat, a player's stat distribution is reflective of how they play the game. After parsing the dataset, the stats were converted into proportions. This is important due to the varying level of the players. Taking the proportion allows us to normalize the issue of varying levels and quality of gear, which will give a player more raw stats. After parsing the data, the second dataset consisted of 24,116 characters and 6 features in total. Given that the goal of this analysis is introducing a recommendation system for players to improve, it is critical to select a feature that allows us to determine which players are "good" players. Candidates for this feature are character level, light level, and combat rating. *Character level* ranges from 1-40 and players can increase their character level by playing the game more and earning "experience points". This is not appropriate since a player may reach level 40 simply by advancing through the game, not by playing optimally. On the other hand, *light level* is calculated from a player's equipment stats. Again, the same reason for not using character level is applied here. *Combat Rating*, which is discussed in more detail in the feature explanations (see also [20]), is instead used as an overall measure of a player's skill. Since

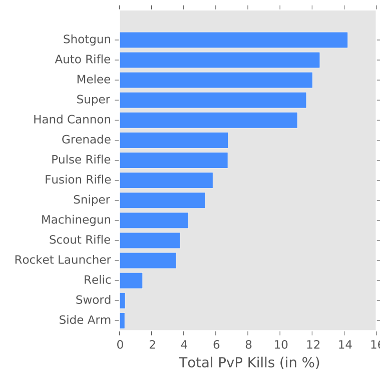


Figure 1: Distribution of kills (in %) for each weapon class. We can see that auto rifles, hand cannons, melee, shotguns and supers are all fairly popular, with each accounting for about 12 percent of overall kills (and 60 percent in total). The remaining weapon classes are less popular, with each accounting for about 4-6 percent of overall kills, excluding side arms and swords which account for less than 1 percent of overall kills combined. Notice that the more popular classes require less accuracy to use compared to the less popular classes. Low accuracy classes, such as the shotgun and auto rifle, require less skill to use than high accuracy classes, such as scout rifles and sniper rifles.

Combat Rating is unique to the PvP mode, it is fitting for our analysis.

As discussed above, lightlevel is calculated from a player's equipment and requires time and skill to increase. At the time this data was taken (during the *Taken King Expansion*), the maximum light level attainable in the game was 335. By considering those with a light level above 200 (the top 40 percent of players), we ensure that the players in our dataset have enough playtime and have at least a degree of freedom of choice in their equipment. This decision was made since low-level players will not have played the game long enough to have earned their desired gear and often lack choices for their gear. The choice of which portion of the players of a game to train models on should in general be based on a consideration of the goal of the analysis, as is the case here.

After merging the two components of the dataset, the initial pool of characters decreased from 24,116 to 8,873. Since the analysis is focused on PvP, only characters that had appeared in the 930,000 tracked PvP matches were considered. Additionally, since *Destiny* tracks all their players quite extensively, we were able to create a concise subset of the overall data. After merging the initial subset based on light level and the initial feature extraction, the final dataset consisted of 2,153 characters and 32 features (from the initial random sample of 10,000 players and 24,116 characters). These features are as follows:

- **Combat Rating:** Combat Rating (CR) is a metric designed by Bungie used to assign a single number which is representative of a given player's overall skill. Although the exact calculation of Combat Rating is hidden by Bungie, we know generally how Combat Rating changes. If a player wins a match, their CR will increase. Similarly if a player loses a match, their CR will decrease. Many online games with matchmaking have some variant of this ranking system. Combat Rating, like other ranking systems, is quite important for a game's matchmaking system to produce balanced matches where all the players are of similar skill levels [20].
- **Proportion Base Stats:** Here we are dealing with the proportion of points placed into Agility, Armor, and Recovery. Agility is used to increase a player's overall movement speed and jump. To understand armor and recovery, we briefly discuss how health works in *Destiny*. A player's overall "lifebar" is split into two segments: actual health and shield. Every player has the same amount of health and shield regardless of what their stats are. Armor can be thought of as damage reduction *in addition to* a player's base defenses. I.e. when the shields go down, a player with higher armor will lose less actual health per hit relative to a player with lower armor. Recovery, on the other hand, effects how fast shields recharge, and reduces the delay of recharge (the time between a shield going down and starting to "recharge"). Additionally, each character created starts with a bonus to one of these three stats. E.g. if a player chooses to be a Hunter, their character receives a +5 bonus to agility.
- **Proportion Cooldown Stats:** Similar to the Base Stats we also consider the proportion of points placed into Discipline, Intellect, and Strength. In PvP, there are 3 specific attacks that are on "charge" and require time to recharge after use. These three attacks are a character's grenade, super, and melee attacks. Discipline helps grenade attacks recharge faster, Intellect helps super attacks recharge faster, and Strength helps melee attacks recharge faster. Note that proportions were used for the Base and Cooldown stats to normalize the effect of gear. Better gear means larger value of raw stats compared to players with worse gear.
- **Inventory List:** To determine a player's favorite weapons, the inventory list is an aggregate of the specific weapons used by a player throughout all PvP matches. After parsing and aggregating 930,000 PvP matches, each character is associated with their own list of specific weapon usage. *Please note: This feature is used solely for weapon recommendations.*
- **Kills-Death Ratio:** One of the de facto first person shooter player ranking features is the kill(s)-death(K/D) ratio [8], which is the ratio of a player's total kills to their total deaths in a given match.
- **Average Score Per Life/Per Kill:** This ratio comprises the player's average score per life (each time they die) and per kill (their average score at the time of a kill). A player's score is a combination of their kills, assists, and any other in-game actions such as capturing an objective. These features help to distinguish players with similar kill-death ratios. The higher the score the larger the impact on the game.
- **Resurrection:** Whenever a player dies, there is the option to "revive" the dead player. A living player must interact with the dead player and take time to revive the dead player. If this action is performed successfully, the previously dead player will be alive and able to resume playing in the current match again. If a dead player is not revived, they will have to wait until the match has ended.
- **Proportion Offensive/Defensive Kills:** In the PvP matches, there are specific match types that are objective-based, such as "Control", where players work together to gain control of an objective/area on the map. During these matches, offensive and defensive kills represent the player's kills that haven taken place either capturing or defending the objective.
- **Average Kill Distance:** To consider proximity preferences of users we incorporate the average kill distance as a feature as well. This keeps track of how far the player is from the other players that are killed. Players who prefer long range weapons, such as snipers, will have a much higher average kill distance than players who prefer close range weapons, such as shotguns.
- **Proportion Weapon Kills**: This composite feature consists of 15 separate features. The proportion of weapon kills represents the proportion of kills that a player got with a weapon class. In Destiny, a player has the freedom to change their weapon load-out after each death, and each weapon belongs to one of these weapon classes. As such, the proportion of weapon kills provides reliable information on how a player chooses to play the game. The possible weapon classes a player can get a kill with are as follows: Auto Rifle, Fusion Rifle, Grenade, Hand Cannon, Machine-gun, Melee, Pulse Rifle, Relic, Rocket Launcher, Scout Rifle, Shotgun, Side Arm, Sniper, Super, and Sword. The weapon classes all have varying levels of power, firing rate, and effective distance. Fig. 1 illustrates the distribution of players killed by the various weapon classes. This distribution allows us to see weapon classes that the overall community uses to get kills.

Table 1: Base Stat Profiles

Cluster	Profile Name	Description
1	Tank	High Armor/Recovery & Low Agility
2	Speedster	Maxed Agility & Low Armor/Recovery
3	Bruiser	High Agility/Armor & Low Recovery
4	Guerrilla	Maxed Recovery & Low Agility/Armor

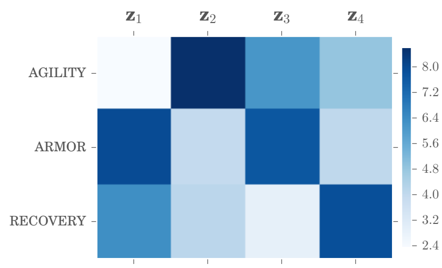


Figure 2: Results on clustering base stats. The results show two clusters (1 and 3) are high on two base stats and two clusters (2 and 4) are maxed out on one stat, but low in the other stats. Players tend to have a preference for one or two base stats as opposed to equally allocating to all three.

5 A PVP RECOMMENDER SYSTEMS FRAMEWORK

The goal was to come up with a novel way to recommend in-game items and stats allocation to Destiny players. Instead of using a single recommender profile, a multi-dimensional alternative approach to player profiling was conceptualized and used as a framework for the final recommendation model. The basic tools we used for the profiling based recommender systems framework are based on factorizing the analyzed data matrices into combinations of low ranked matrices [2, 17]. Formally given a column data matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ (containing m dimensional n data points, which in our case player records) and a rank k, the primary objective of matrix factorization is to come up with a basis matrix $\mathbf{Z} \in \mathbb{R}^{m \times k}$ and a mixing coefficient matrix $\mathbf{H} \in \mathbb{R}^{n \times k}$ to represent the data matrix as $\mathbf{X} \approx \mathbf{Z}\mathbf{H}^T$. Appropriate factor matrices $\mathbf{Z}$ and $\mathbf{H}$ can be found by minimizing the squared matrix norm $\left\|\mathbf{X}-\mathbf{Z}\mathbf{H}^T\right\|^2$. The minimization process usually follows an alternating optimization scheme that iteratively updates each factor matrix individually keeping the other fixed [2, 4, 18]. It is important to note that, due to interpretability of the resulting factor matrices and regularization of the data representation we usually impose constraints on the factor matrices [2, 18, 19]. To illustrate, for the popular k-means clustering algorithm $\mathbf{Z}$ contains the mean vectors and $\mathbf{H}$ contains the binary indicators to each of the clusters whereas for Archetypal analysis $\mathbf{Z}$ contains the extreme prototypical points (that are convex combination of *certain* data points from $\mathbf{X}$) and $\mathbf{H}$ contains row stochastic mixing (or belongingness) coefficient values to indicate proximity to particular archetypes in $\mathbf{Z}$.

Table 2: Cooldown Stat Profiles

Cluster	Profile	Description
1	DISC/INT	High on Discipline and Intellect
2	DISC/STR	High on Discipline and Strength
3	STR/INT	High on Strength and Intellect

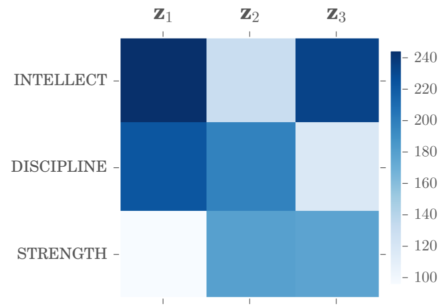


Figure 3: Results on clustering cooldown stats. The results show clusters that are high on two stats and low on the other. Players tend to prefer having very low cooldowns on two abilities instead of equally spreading across all three.

Player Profiling with k-means Clustering

A popular technique to group similar players together in game analytics is the k-means clustering algorithm [2, 8]. This method was chosen as it provides an efficient way to characterize the different behaviors of players on average. K-means clustering groups a given dataset into a fixed number of k clusters. The algorithm focuses on calculating centroids for each of the cluster and assigns each data point to the nearest centroid. This process is done iteratively until the centroids converge to their final values. It results in minimizing in-cluster variance and maximizing inter-cluster variance, which is exactly what was desired when it came to classifying players in Destiny. Traditionally, k-means performs well for grouping average tendencies in the dataset and is not the best approach if trying to find clusters that define extreme behaviors of players. As explained later in this work, Archetypal Analysis [5] was used when it was desired to cluster players based on their game-play styles. When it came to analyzing the base stats and cooldown stats of players, the extreme allocations would just be maxing out on one of the stats which does not help in the classification process. Hence, it was reasonable to utilize k-means to come up with the common configurations the players were using for their characters.

Profiling Base Stats: The game has three base stats that were focused on namely, Agility, Armor & Recovery. Players customize their characters by allocating points to each of these base stats to complement their class and game-play style. After analyzing the results from k-means for 3, 4 and

5 clusters, the 4 cluster results were chosen to be the best balance between granularity and interpretable clusters. The cluster definitions can be visualized using Fig. 2. Using domain knowledge on Destiny and other games, each of the clusters was assigned a profile to reflect the thought process of the players behind their allocations. The four profiles created based on base stats are shown in Table 1.

Profiling Cooldown Stats: The game also has three other stats that players can allocate to improve the cooldown times of various abilities like special, grenade, etc. These stats could also serve as potential profiling metrics to characterize players and their play-styles. k-means clustering was performed over the three cooldown stats, viz. Strength, Discipline & Intellect. In the case of cooldown stats, it was reasonable to have 3 clusters as more often than not, the players would max out on 2 of the 3 stats based on their requirements. Allocating equally to all 3 is much sub-optimal and is rarely done by the high-level players. The cluster definitions and profile assignments can be seen in Fig. 3 and Table 2 respectively.

Player Profiling with Archetypal Analysis

In *Destiny*, players are constantly changing their playstyle, whether to try out something new or to keep up with the metagame. As such, we wanted to identify the main playstyles in the game. Archetypal analysis is used to determine the extreme entities, the *archetypes*, in a given dataset. These archetypes are prototypical points that will represent a given population. Once the archetypes have been identified, every player in the dataset can be represented as a convex combination of these extremes [15, 17].

The archetypes are typically not manifestations of actual players, but rather are manifestations of extreme behavior qualities. Thus, players typically have less extreme values relative to the archetypes. After calculating the archetypes for each of the players in the dataset, players were assigned to the archetype with the largest belongingness value, resulting in archetypal clusters. Since Archetypal analysis is focused on the extreme entities, there is a more pronounced difference between the archetypal clusters relative to the difference in centroid based clustering algorithms.

The results from our archetypal analysis clustering is shown in Fig. 4. The optimal number of archetypes was 6, based on our scree plot analysis and the distribution of players falling into each archetype. Additionally, domain knowledge was used to pick clusters assignments that were not arbitrarily broad or specific. Most of the archetypes refer to players with specific weapon preferences.

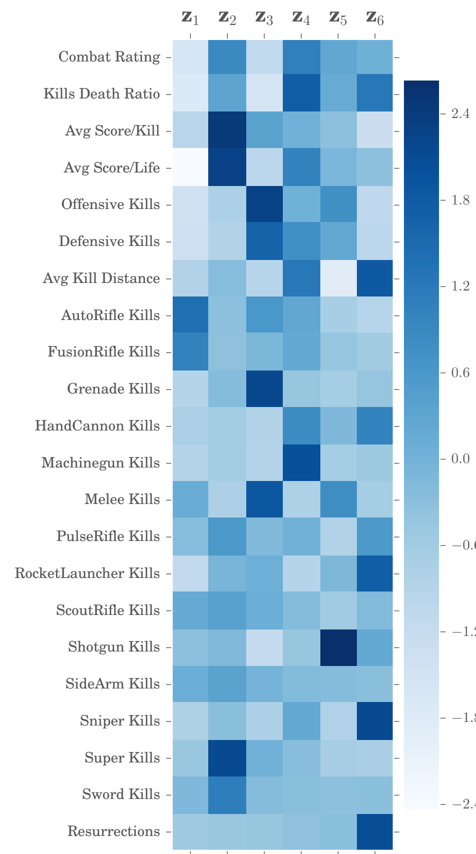


Figure 4: Illustration of six distinct archetypes of playstyles. Six archetypes were chosen based on the interpretability and distinctiveness of each archetype. Some archetypes are defined by specific weapon usage, such as z_1, z_5, and z_6 for Auto Rifles, Shotguns, and Sniper Rifles respectively. Other archetypes represent a general playstyle, such as z_2 being a player who relies on timing their super ability to score massive amounts of points.

The Recommender System

Rather than relying on a single behavioral dimension for building the recommender system, all the three different player profiles across base stats, cooldown stats and in-game performance were used. The recommender approach was two-pronged: 1) Recommend weapon loadouts to players based on similar players and: 2) Recommend optimal allocations for both base stats and cooldown stats.

The main idea of the multi-profile recommendation framework is illustrated in Fig. 5. For each of the profiles represented as circles, there are clusters within each profile that a player falls into. Each intersection represents the pool of players that can be considered for recommending on. For example, lets say Player X wanted recommendations on how to improve. Intersection 1 represents players that are most similar to X across all three profiles since they fall into the same cluster/archetype assignments. However, Player X may wish to know how players similar to him across two profiles, but different in the third, are doing. Intersections 2,3,4

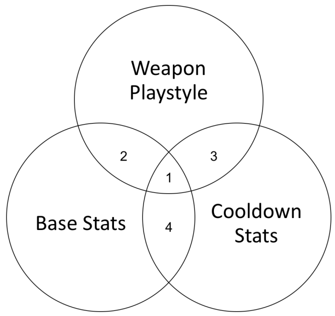


Figure 5: Illustration of the three different player profile perspectives we use to generate our recommendations.

represent players that are different in a third profile. For example, taking the players at intersection 2 to recommend on would give players that have a high variation in cooldown stats compared with player X. This recommendation framework provides a flexible way to consider different aspects of gameplay and take into account what the player is willing to change. *Weapon Recommendations:* For a given player, the first step was to find similar players using the three profiles. The 3-way intersection set (region 1 in Fig. 5) of players having same profile assignments as the target player was found. From these set of similar players, we filtered out two players - the best player & the closest (most similar) player. The best player was found by simply finding the one having the maximum value for combat rating. The closest player on the other hand was found using the k-nearest Neighbors technique. The system then recommends weapon loadouts for both of these players respectively labeling them as loadouts for best and closest player. *Stats Allocations:* For recommending optimal stats allocation, a different approach was required, as they act as one of the three profiling dimensions. Due to this reason, when computing intersection sets of similar players the dimension to be recommended is left out. For instance, when recommending optimal cooldown stats allocation, the 2-way intersection (region 1 + region 2 in Fig. 5) between base stats and playstyles is computed. Also, as the allocation of stats is closely tied to the class of the character, an additional filtering was added to keep only players belonging to the same class as that of the target player. On top of this, only similar players that had a higher combat rating than the target player were kept. Taking these measures ensured that the recommendations were reasonable and would be useful to the player.

After finding the desired set of similar players, the distribution of players was calculated on the recommendation dimension. Continuing from the previous example of recommending optimal cooldown stats allocation, the distribution of the similar players was calculated across the three cooldown profiles. The profile containing the maximum number of players was then compared with the target's cooldown profile and an appropriate recommendation to move points across the three stats was provided.

Table 3: Summary Statistics of Reddit User Sample

Measure	Mean	Max	Min
Time Played (Hours)	112.4	122.1	106.2
Light Level	384.7	400	209
Combat Rating	94.9	144.4	52.4
Kills+Assists/Death Ratio	1.2	2.1	.1

6 EVALUATION AND RESULTS

Recommender systems are usually evaluated in offline and online fashion[4, 11, 13, 15]. Offline evaluations provide an ability to gauge the accuracy of the algorithm without having to test the system with live users. Instead they utilize existing data with some removed information [11, 15] to *simulate* live systems. The recommender algorithm is evaluated by its ability to recommend the missing information. After applying the recommendation, the difference between the recommended information and the actual information is calculated via a loss function [4, 11, 13]. While usually robust for a wide variety of recommenders, this approach was not appropriate for multi-profile recommendation, as one its main components is weapon information. Weapons in Destiny are, by nature, highly substitutable by other weapons. For example, while one shotgun may be used by a slight majority of top tier players, another shotgun may be just as deadly in the hands of slightly different, but indistinguishable to the algorithm, players. For this reason, calculating loss off of the recommendations would be next to impossible [1, 4, 11, 13, 15]. For this reason, an evaluation via a user study as defined by Shani and Gunawardana [13] was instead performed on real *Destiny* players (a similar general approach also adopted by Anwar et al. [1]).

User Study Evaluation

To evaluate the potential uses of the recommender system, general sentiment and feedback was sought from the active users on the Reddit community /r/DestinyTheGame during March 2017. The benefit of asking this community to evaluate the recommender is the experience that came with the users. Each has extensive experience with the game and its metagame and they were able to provide educated feedback about the performance of the system. The drawback of using the Reddit community, however, is that the sample of users surveyed were biased. The users were already enthusiastic about *Destiny*, and may have responded more positively than a randomly selected sample. See Table 3 for sample statistics.

Base Stats:

Profile Name	Description
Tank	High Armor/Recovery, low Agility
Speedster	Maxed Agility, lower Armor/Recovery
Bruiser	High Agility/Armor, low Recovery
Guerilla	Maxed Recovery, lower Agility/Armor

Your player is a **Tank**.

Figure 6: Section 1 of the personalized player report. Players are given descriptions of each cluster within each profile, and told which cluster their character falls into, in this case a tank (high armor and ability to withstand damage).

This is the best player's most common loadout:
Primary: Haakon's Hatchet, Auto Rifle
Special: Strongbow-D, Shotgun
Heavy: Gjallarhorn, Rocket Launcher
Combat Rating: 134.45

This is the loadout of the player MOST similar to you (your nearest neighbor):
Primary: Harrowed Smite of Merain, Pulse Rifle
Special: Queenbreakers' Bow, Fusion Rifle
Heavy: Harrowed Elulim's Frenzy, Rocket Launcher
Combat Rating:105.03

Figure 7: Section 2 of the personalized player report. Players are told the top weapon loadout of the best player in their intersection, by combat rating, as well as the top weapon loadout of their nearest neighbor.

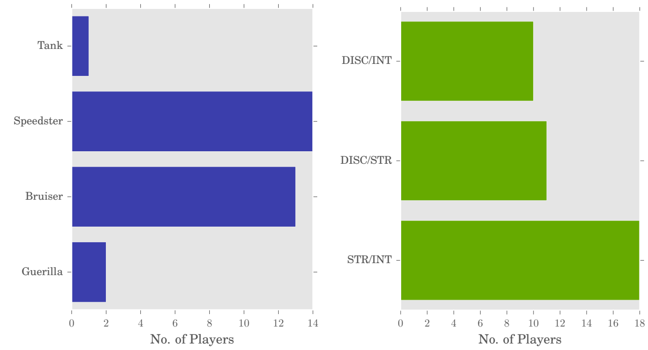
For each player from Reddit, their data was pulled from the API and run through our system. We then created personalized recommendations for each user. Contents of the reports included four sections:

Profile Assignments (Fig. 6) - Describe each profile (base stat, cooldown attribute, playstyle) and tell the user which cluster they fall into under each profile.

Weapons (Fig. 7) - Give the user the top weapon loadout (Primary, Special, and Heavy Weapon) for the best player, as well as the top weapon loadout for the user's nearest neighbor.

Stats (Fig. 8) - Show the user how players with higher combat ratings allocate their stats. Two histograms are shown visualizing the distributions of players in two sets of profiles: one for base stats, and the other for cooldown attributes.

Recommendation (Fig. 9) - Based on the weapon usage of players better than the user, up to three suggested weapons are shown as recommendations, as well as a suggestion on how to reallocate stats (if necessary). Average combat rating of the players using the recommended weapons and stat allocations is shown to reinforce the validity of the recommendation.



(a) Base Stats Profiles **(b) Cool Down Stats Profiles**

Figure 8: Section 3 of the personalized player report. Given a user's base stat and cooldown stat allocations, the distribution of how other similar, but better, players allocate their stats is shown.

Based on similar players who have a higher combat rating, we recommend that you try the following weapons:

1000-Yard Stare, Sniper Rifle
Harrowed Elulim's Frenzy, Rocket Launcher
Harrowed Qullim's Terminus, Machine Gun

We also recommend you stick with **Recovery** and **Discipline/Strength**.

The players doing these things have on average a **11.96** better combat rating.

Figure 9: Section 4 of the personalized player report. A final soft recommendation is delivered based on the current stat allocation and weapon choice of the user's character. Validation of the recommendation is given by telling the user that players who have made these choices have a better combat rating.

These reports were sent to each user with a survey attached, asking questions about their opinion on the usefulness of the recommendation and if they would actually act on it, assuming the data was up to date.

Out of 50 the original respondents, 30 responded to the survey sent along with the personalized reports. The response was generally positive: 80% of players stated that they found the recommendations helpful and that they would act on them. When asked "Would you like to see this implemented into a service for you to use?", over 90% said yes. However, one responder also noted that the recommendations were not very helpful since players all have different preferences. This brings us back to the problem of player motivations. Some players simply want to get better through practice and not through weapon recommendations. The main takeaway here is that players are hungry to know about which specific areas of the game they can improve on, but perhaps veterans of the game feel like they do not need suggestions.

Given the nature of the recommendations, the generally positive response is encouraging for the potential application of the algorithm. Further online testing of our framework is necessary to provide tangible results.

7 CONCLUSION AND FUTURE WORK

In this paper, a multi-profile recommendation framework was developed for *Destiny* across three distinct game play features: base stats, cool down stats, and weapon play style. This framework allows for flexibility in choosing which features to recommend on and how much variability is desired for those features. A basic expert sample, online evaluation of the system through Reddit revealed the recommendations were interesting and valuable to players. Furthermore, players indicated an interest in acting on the recommendations in order to see if their performance would improve. Future work regarding this system involves longitudinal live testing on the recommendation framework, meaning select players would be followed and game telemetry would be analyzed to see if these players improved from the recommendations they were given. With the recent release of *Destiny 2*, the focus of applying our recommender framework will transfer to this game. Although many gameplay features have changed in *Destiny 2*, such as the omission of cooldown stats, a similar system can be built for the sequel.

While specific to *Destiny*, the framework of the recommender system presented here can be applied in similar titles, notably MMOGs and team-based eSports titles. Additionally, the adoption of multiple profiles for each player represents a novel angle on previous work in behavioral profiling, which has generally focused on developing one set of profiles for the entire game, or specific sections thereof. While a system of three profiles were presented here, the methodology is in principle generalizable to n number of profiles. Doing this would create numerous distinct intersections to build the recommendation on, encompassing any desired complexity of any game. E.g. a four profile-system could be built for a *League of Legends* where the profiles are item builds, mastery trees, rune pages, and ability leveling. This has potentially significant implications in the eSports scene, an environment where strategies and the metagame are rapidly evolving.

Acknowledgements

Part of this work was conducted in the Digital Creativity Labs (www.digitalcreativity.ac.uk), jointly funded by EPSRC/AHRC/InnovateUK under grant no EP/M023265/1.

REFERENCES

[1] Syed M. Anwar, Talha Shahzad, Zunaira Sattar, Rahma Khan, and Muhammad Majid. 2017. A game recommender system using collaborative filtering (GAMBIT). In *IEEE Applied Science and Technologies.*

[2] Christian Bauckhage, Anders Drachen, and Rafet Sifa. 2015. Clustering game behavior data. *IEEE Transactions on Computational Intelligence and AI in Games* 7, 3 (2015), 266–278.

[3] John Bohannon. 2010. Game-Miners Grapple with Massive Data. *Science* 330, 6000 (2010), 30–31.

[4] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. 2010. Performance of Recommender Algorithms on Top-N Recommendation Tasks. In *Proceedings of ACM Recommender Systems Conference.*

[5] Adele Cutler and Leo Breiman. 1994. Archetypal Analysis. *Technometrics* 36, 4 (1994), 338–347.

[6] Anders Drachen, James Green, Chester Gray, Elie Harik, Patty Lu, Rafet Sifa, and Diego Klabjan. 2016. Guns and Guardians: Comparative cluster analysis and behavioral profiling in Destiny. In *Proc. IEEE CIG.*

[7] Anders Drachen, Eric Lunquist, Yungyen Kung, Pranav Rao, Diego Klabjan, Rafet Sifa, and Julian Runge. 2016. Rapid Prediction of Player Retention in Free-to-Play Mobile Games. In *Proc. of AAAI AIIDE.*

[8] 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 *Proc. of IEEE CIG.*

[9] Anders Drachen, Georgios N. Yannakakis, Alessandro Canossa, and Julian Togelius. 2009. Player Modeling using Self-Organization in Tomb Raider: Underworld. In *Proc of IEEE CIG.*

[10] Magy S. El-Nasr, Anders Drachen, and Alessandro Canossa. 2013. *Game Analytics - Maximizing the Value of Player Data.* Springer.

[11] Paul Kantor, Lior Rokach, Francesco Ricci, and Bracha Shapira. 2011. *Recommender Systems Handbook.* Springer.

[12] Mike Minotti. 2016. Destiny passes 30 million registered players. https://venturebeat.com/2016/05/05/destiny-now-has-over-30-million-registered-players/. *VentureBeat* (2016).

[13] Guy Shani and Asela Gunawardana. 2009. A Survey of Accuracy Evaluation Metrics of Recommendation Tasks. In *JMLR.*

[14] Kyong J. Shim and Jaideep Srivastava. 2010. Behavioral Profiles of Character Types in EverQuest II. In *Proceedings of IEEE CIG.*

[15] Rafet Sifa, Christian Bauckhage, and Anders Drachen. 2014. Archetypal Game Recommender Systems. *Proc. of KDML-LWA.*

[16] Rafet Sifa, Christian Bauckhage, and Anders Drachen. 2014. The Playtime Principle: Large-scale Cross-games Interest Modeling. In *Proc. of IEEE CIG.*

[17] Rafet Sifa, Anders Drachen, Christian Bauckhage, Christian Thurau, and Alessandro Canossa. 2013. Behavior Evolution in Tomb Raider: Underworld. In *Proc. of IEEE CIG.*

[18] Rafet Sifa, Cesar Ojeda, and Christian Bauckhage. 2015. User Churn Migration Analysis with DEDICOM. In *Proc. of ACM RecSys.*

[19] Rafet Sifa, Sridev Srikanth, Anders Drachen, Cesar Ojeda, and Christian Bauckhage. 2016. Predicting Retention in Sandbox Games with Tensor Factorization-based Representation Learning. In *Proc. of IEEE CIG.*

[20] Tom Stafford, Sam Devlin, Rafet Sifa, and Anders Drachen. 2017. Exploration and Skill Acquisition in a Major Online Game. In *The Annual Meeting of the Cognitive Science Society.*

[21] Ruck Thawonmas, Keisuke Yoshida, Jing-Kai Lou, , and Kuan-Ta Chen. 2011. Analysis of revisitations in online games. *Entertainment Computing* 2 (2011), 215–221. Issue 4.

[22] Ben Weber. 2015. Building a Recommendation System for EverQuest Landmark's Marketplace. *Game Developers Conference* (2015).

[23] Georgios N. Yannakakis. 2012. Game AI Revisited. In *Proc. of ACM Computing Frontiers Conference.* 285–292.