University of Glasgow | School of Computing Science

A quantitative analysis of the RPGLite mobile game logged usage

Jeong Woo Yang

School of Computing Science

Sir Alwyn Williams Building

University of Glasgow

G12 8RZ

A dissertation presented in part fulfillment of the requirements of the Degree of Master of Science at the University of Glasgow

02-09-2022

Abstract

In this dissertation, log data from a gaming application made for research purposes is analyzed to observe user behavior such as engagement and development of mastery in gaming skills. Three groups of players were observed in this study- players who drop out while playing the game, players who develop mastery as they play, and players who do not develop mastery despite long playtime. This problem is addressed through observation of user trends by segment and utilization of machine learning methods. This paper addresses the question of gaming mastery in quantitative terms, by using the pre-defined skill point metrics or processed metrics and using them in machine learning classification methods. Furthermore, this paper attempts to address the issue of game balancing and player preferences in terms of both machine learning and quantitative analysis. Finally, the paper concludes that data through user logs could be engineered and used to balance competitive games and get insights about user behavior.

Education Use Consent

I hereby give my permission for this project to be shown to other University of Glasgow students and to be distributed in an electronic form.

Name: Jeong Woo Yang   Signature: Yang

Acknowledgements

This research has been extremely enjoyable. From the day I got the dissertation topic, I was joking to my friends that the assignment found the right person who is a core gamer. Through my research, the people around me made it a fun experience.

First and foremost, my supervisor **Prof. Oana Andrei** for bringing up this interesting topic and offering me guidance through the whole process.

My friends and colleagues at the University of Glasgow, who provided support and fun discussions related to gaming. Notable mentions to **Harsh Jain, Manikanth Sai** and **Sristi Bhat**.

Contents

Chapter 1 Introduction

1.1 Motivation

The free-to-play business model is the dominant business model of today's multiplayer video game industry. On Steam, the world's largest digital game software distribution service, free-to-play games such as *Counter-Strike: Global Offensive*, *DOTA 2* and *Lost Ark* have consistently been in the top spot for monthly active users [1]. To make profit, the free-to-play model requires users to be consistently engaged with the game and make microtransactions. Therefore, user retention has become a gist in game design. Game companies try to implement Bushnell's Law, the idea that games should be 'easy to learn, hard to master' [2].

By designing a game to be 'easy to learn, hard to master', gaming companies try to motivate users with the concept of '**mastery**'. 'Mastery' is the concept that one's knowledge and skills at a particular video game is better than other players [3]. However, this could be a double-edged blade. In a competitive multiplayer game, players tend to show polarized responses as they progress throughout the game. Some players exploit their knowledge of the game mechanics and compete for top spots in competitive modes. Others grow weary of the contents or the competition and drop out. Many popular competitive video games such as *DOTA 2* see a decline in new user influx as the competition between skilled users intensify. *DOTA 2*"s user count has been stagnating at 10 million users for 5 years since 2015 [1].

Today's multiplayer games offer a variety of **game material**, or the elements of the game which the player can choose to interact with, whether be it in forms of character classes, items or stages. As the users interact with the game material [4], they develop their mastery, and gameplay would become streamlined. This streamlined state is called as the **metagame**. Contemporary competitive multiplayer games such as *League of Legends* or *DOTA 2* try to keep the users on their toes by consistently patching the game and disrupt the metagame. This too, could bring in unforeseen consequences. If done too early, the users could not be given a chance to develop mastery and immerse into the game. If done too late, users could fall off due to boredom from repetition.

1.2 Aims and Objectives

This study will employ data science methods to quantify and identify specific user behaviors that could be attributed to a particular segment of mastery or engagement. The study focuses on the following topics:

i) Identification and classification of users who display their 'mastery' of the game

ii) Identification of users who lose enthusiasm and fall out, and analysis of their behavior pattern.

iii) Employment of machine learning techniques and quantitative methods to improve competitive balance, or the condition of all options being variable in a contest.

1.3 Report Structure

This study is structured into a background introduction (Chapter 2), a discussion of the data and methodological approach (Chapter 3), an overview of its empirical findings and their discussion (Chapter 4), discussion of limitations and future work (Chapter 5) and a conclusion with personal reflections (Chapter 6).

Chapter 2 Background

2.1 RPGLite, the game

RPGLite is a game developed by William Kavanagh and Tom Wallis as a study on game balancing [4]. It has been played by 370 users over a time period of 6 months. Each round is a turn-based game played by two registered users who choose two classes of characters from a pool of eight classes with different mechanics. To keep the players engaged and motivated to play, the creators implemented RPGLite's own versions of achievements and leaderboards. RPGLite has undergone changes such as addition of classes or new features such as forfeiture during the course of game balancing study. Due to the availability of datasets, the version of RPGLite that will be discussed in this study will be RPGLite 3, which is the latest version that has been in service.

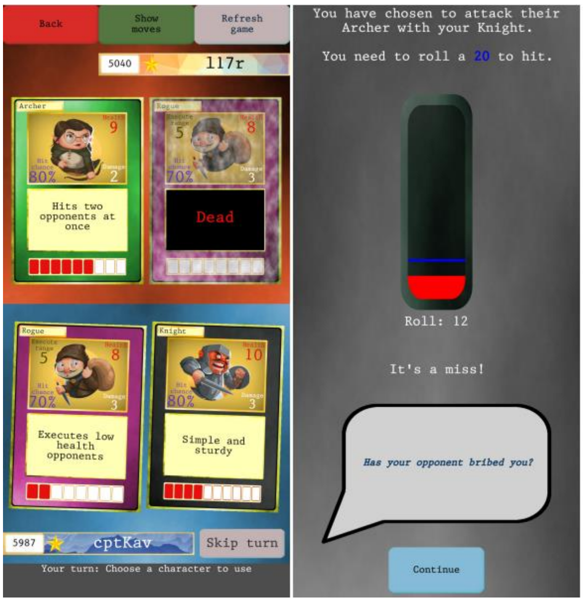


Figure 2.1, Game screen and dice roll animation of RPGLite [4]

To win in RPGLite, the player has to reduce the opponent's characters' 'Health' points to zero. When attacking, the player has to choose a target to attack. The success of the attack is determined by a preset 'Accuracy' parameter, and the attack will do damage according to the 'Damage' parameter. If there is a special ability, it will trigger with a successful attack or become active when a certain condition is met.

Class	Attribute (pre-v1.2)	Attribute (v1.2)	Special mechanism
Knight	H:10, A:60%, D:4	H:10, A:80%, D:3	None
Archer	H:8, A:85%, D:2	H:9, A:80%, D:2	Can attack both targets

Wizard	H:8, A:85%, D:2	H:8, A:85%, D:2	Makes target skip turn when attack hits
Healer	H:10, A:85%, D:2	H:9, A:90%, D:2	Heals self and ally for 1 hp when attack hits
Rogue	H:8, A:75%, D:3	H:8, A:70%, D:3	Deals 5 damage to targets under 5 health
Monk	H:7, A:80%, D:1	H:7, A:75%, D:1	Can strike again if attack hits target
Barbarian	H:10, A:75%, D:3	H:9, A:70%, D:3	Hits for 5 damage when below 5 health
Gunner	H:8, A:75%, D:4	H:8, A:70%, D:4	Deals 1 damage instead of missing

Table 2.1, Summary of basic attributes and special abilities of characters in RPGLite. (H: Health, A: Accuracy, D: Damage) [4]

In their study, Kavanaugh et al. designed the eight classes of RPGLite with different mechanisms and stats, to simulate the concept of competitive balance in multiplayer PvP games [4]. The players initially started with the more basic classes such as the Knight, which could be considered as the most 'stable' option as it has the highest health. Then, with each win, players unlock the more complex character options such as the Healer or the Barbarian. This serves two purposes, to keep players motivated to play to discover new content, and to help them learn about game mechanisms while playing.

2.2 Datasets

The datasets used in the research are the RPGLite Player Data and Lookup Tables [5], which contain data about players, games and interactions that took place during RPGLite's service. The three tables stored in a single JSON file are as follows:

i) Players: contains the players' details such as username, games played, games won, skill points, ELO ratings.
ii) Games: contains details about RPGLite games that took place such as the players, characters chosen, date, and winner
iii) Page hits: contains information about users' activities in RPGLite, such as the screens accessed alongside timestamp.

Full details about the tables are available in Appendix 1.

2.2.1 Exploratory data analysis - Players

The players dataset contained the players' IDs, names, date and time of last login and numerical variables such as skill points, ELO scores, games played and won spread over 15 columns and 360 rows corresponding to each one of 360 players. Some rows such as character count were in JSON, and thus had to be spread into columns for further investigation.

To evaluate the correlation between games played and skill rate, a comparative study was performed.

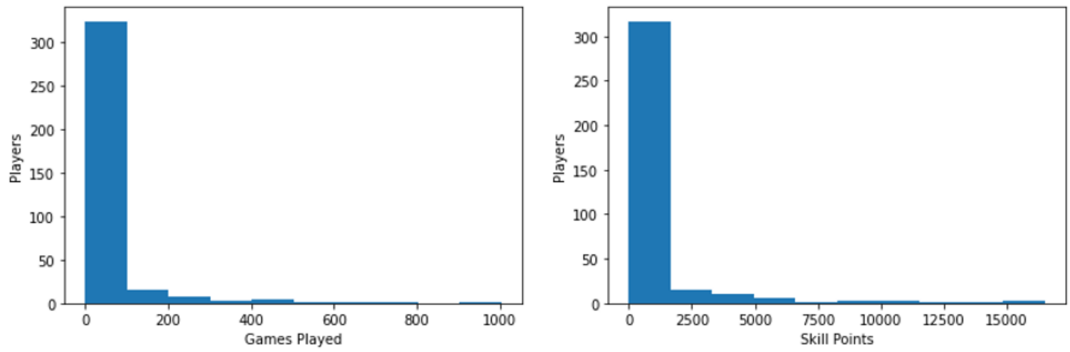


Figure 2.2 The Initial dataset's histogram of distribution of games played and skill points

In Figure 2.2, the distribution of skill points and games played show a very left-skew unimodal distribution. Among the 360 users, 189 of them did not play a single game. Therefore, although the average number of games played by an RPGLite user was 42, the median number was 0 with a standard deviation of 122 games.

Furthermore, the correlation of games played to games won was examined to examine the impact of player experience to actual victories.

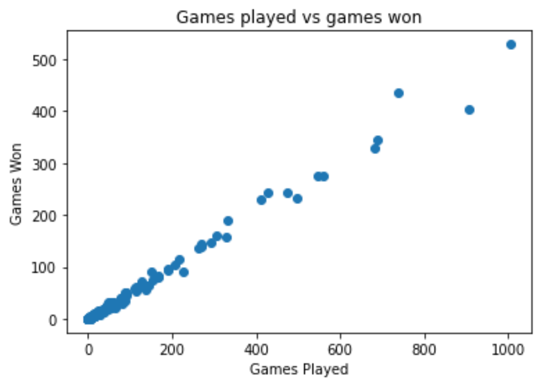


Figure 2.3 The Initial dataset's scatter plot of distribution of games played and games won

The scatter plot shows almost a linear line, and the two variables 'played' and 'win' have a Pearson correlation coefficient of 0.99. However, when the 'win rate' of the player was taken into consideration along with the skill points, the relationship was shown in a nonlinear pattern.

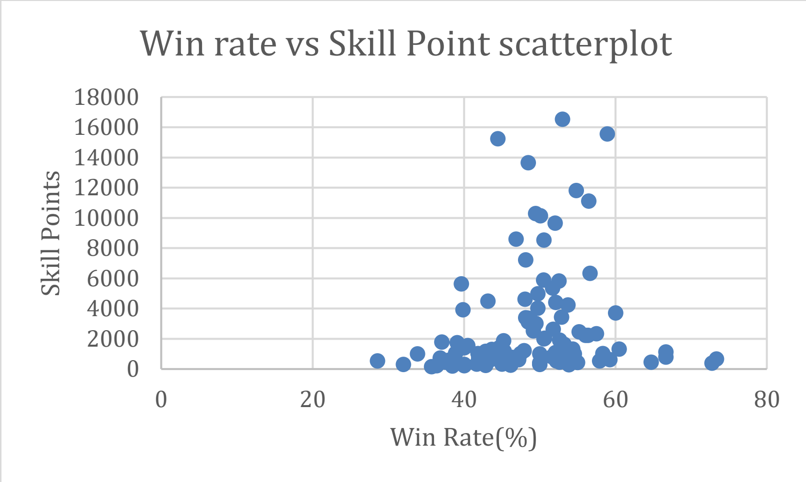


Figure 2.4 The Initial dataset's scatter plot of distribution of win rate vs skill points

Higher win rates did not mean higher skill points. In fact, there were players with extremely high skill points but with win rate below 50%.

2.2.2 Exploratory data analysis - Games

The games dataset contained 14 columns and 7446 rows and contained data such as players involved, balance code, winner, moves made and characters selected. There were 3407 games by 103 players before patch 1.2 and 4039 games by 99 players after patch 1.2. Patch 1.2 was run for almost 3 times the duration of patch 1.0. Thus, the pre-patch timings could be taken as the time when the user influx is high and the users are exploring through the game material, while patch 1.2 could be taken as the period where the users have been given enough time and started to figure out the game material.

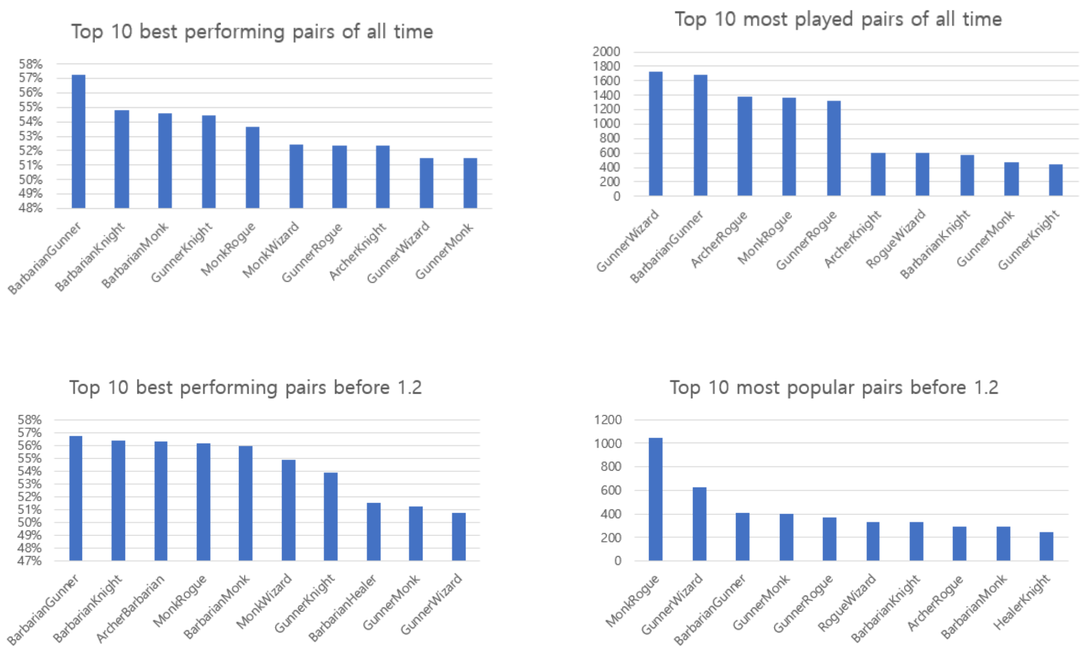


Figure 2.5 The top 10 most successful and most played pairs of all time and pre-patch 1.2

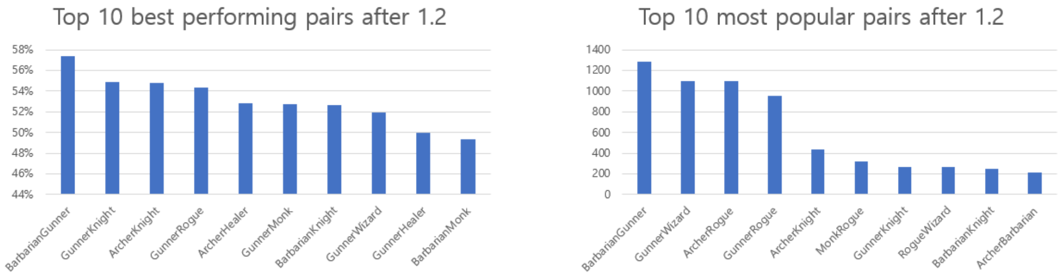


Figure 2.6 The top 10 most successful and most played pairs of patches 1.2

By examining the trend changes before and after patch 1.2(Figure 2.5 vs Figure 2.6) and comparing them with the all-time records, it could be seen that the patch 1.2 changed the landscape of the game. Also, here are some discrepancies between the most popular pairs and best performing pairs.

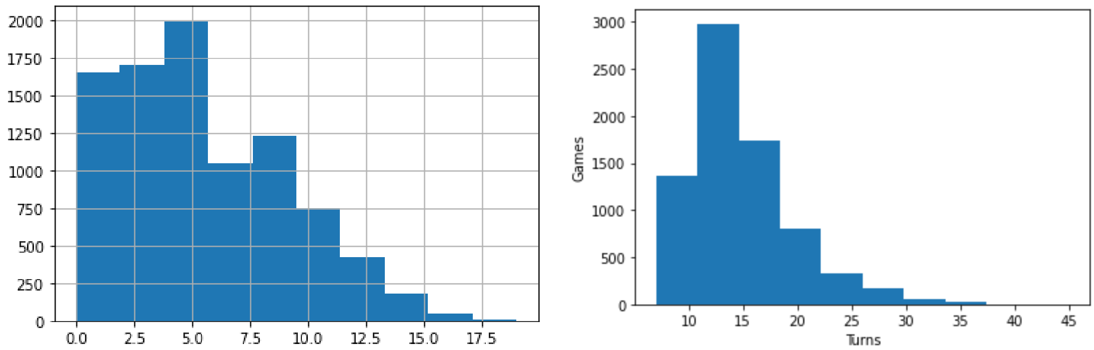


Figure 2.7 Histogram of health difference between players when game concludes (left) and histogram of RPGLite game length(right)

Furthermore, the columns move, p1c1_health, p1c2_health, p2c1_health, p2c2_health were observed to find out how large are the health margins when a game of RPGLite ends. The length of turns was calculated by counting the number of commas used to distinguish between states in the move column of the games dataset and adding 1 for the last turn which would not have a comma in front of it. Most RPGLite games end in 14 turns with around 5 health difference (which means each surviving character has 2 or 3 health).

2.3 Concept of 'mastery' in a competitive video game

In competitive video games which players play against each other, players devise new strategies to outperform their competition. This ability to grasp the game material and perform better than peers is called **Mastery.** Mastery is a motivator in video games, and many competitive games introduce ranked tiers to keep players competing for higher ranks [3].

Mastery can take be presented in different forms in different levels. Some indicators of mastery are quantitative. For example, *DOTA 2* players in higher skill brackets tend to finish the game around 10 minutes earlier than their less skilled peers [6]. Other indicators of skill can be qualitative. For example, in League of Legends Championship Korea 2022, out of the 161 characters provided in *League of Legends*, only 104(62%) were selected by professional players [7]. As various game materials represent various concepts or archetypes, it could be inferred that some archetypes are preferred over the others as the players' knowledge level increase.

Furthermore, mastery can be examined from two different perspectives – 'macro' and 'micro'. This approach was introduced by Donaldson, who performed observation research on *League of Legends* players. Donaldson proposed that mastery in *League of Legends* could be examined in the perspective of 'mechanics' and 'metagame', the former meaning skills of manipulating one's player character, and the latter referring to skills related to making advantageous choices in terms of the big picture such as team composition (Donaldson). To expand Donaldson's perspectives to games which require less mechanics such as turn-based, the terms have been renamed to 'macro' and 'micro' [8].

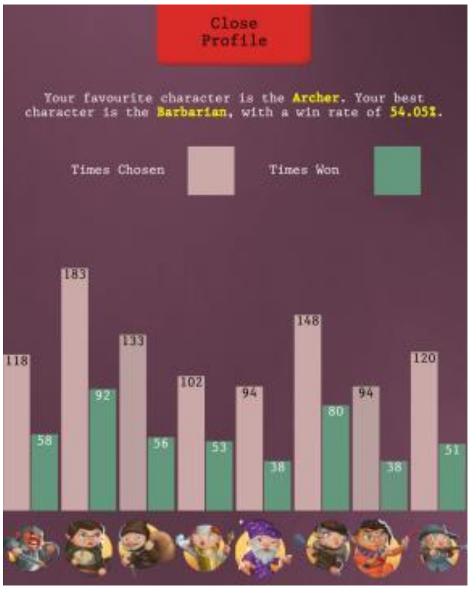


Figure 2.8 RPGLite's profile screen, where players can see their progress and mastery [4]

'Macro' level of mastery refers to mastery shown before the actual player vs player interaction starts. As users progress through the game material, they experiment with various concepts, and discover how different design concepts interact with each other. Then, players come up with a 'strategy' after combining different elements of the game material to overpower their competition. During Kavanaugh et al.'s research, many users took advantage of the Monk class to quickly lower the opponent's characters' health and quickly execute them with the Rogue class that deals bonus damage to targets under a threshold [4]. In this case, the users who were quicker to discover the synergy between the Monk and the Rogue could be considered to have a high level of 'macro' mastery. In RPGLite, the user could evaluate his 'macro' mastery through the profile menu, where all the stats about pick and win rates of the classes are displayed.

'Micro' level of mastery deals with mastery that is made after player vs player interaction has begun in the game, after selecting the game materials to interact with. Some games such as chess provide the same game materials for both the players. Nevertheless, players show difference in skill levels. In the aforementioned example of chess, the players utilize around 80 different openings [9]. This difference in 'tactic' employed to carry out the 'strategy' would be considered as 'micro' mastery. Kavanaugh et al.'s study focused on this perspective, examining how players made mistakes when they could have gone for the win [4].

Kavanaugh et al. tried to quantify this concept of mastery using two measures. One was the traditional ELO rating system used in games such as Chess. However, ELO rating was found to be a poor rating of success in RPGLite, and thus the researchers developed his own skill rankings, that start from 1200 and are altered accordingly by the formula below [4].

$$
\begin{aligned}
new(S_1) &= S_1 + 40 + \delta \\
new(S_2) &= S_2 - 10 + \delta
\end{aligned}
\quad \text{where} \quad
\delta = \begin{cases}
5 & \text{if } S_1 - S_2 > 500 \\
4 & \text{if } 500 \geq S_1 - S_2 > 400 \\
3 & \text{if } 400 \geq S_1 - S_2 > 300 \\
2 & \text{if } 300 \geq S_1 - S_2 > 200 \\
1 & \text{if } 200 \geq S_1 - S_2 > 100 \\
0 & \text{if } |S_1 - S_2| \leq 100 \\
-1 & \text{if } 200 \geq S_2 - S_1 > 100 \\
-2 & \text{if } 300 \geq S_2 - S_1 > 200 \\
-3 & \text{if } 400 \geq S_2 - S_1 > 300 \\
-4 & \text{if } 500 \geq S_2 - S_1 > 400 \\
-5 & \text{if } S_2 - S_1 > 500
\end{cases}
$$

Figure 2.9 Initial skill point equation displaying how skill point of player 1(S_1) changes after a victory over player 2(S_2) [4]

2.4 The concept and importance of competitive balance

In competitive video games, keeping '**balance**', or providing the users with the most competitively viable game materials possible is an important factor to keep the users playing. According to Yee's study, mastery leads to the feeling of 'power', or the feeling that the player is more powerful than his or her peers [3]. However, lack of options in the game material could lead to quicker depletion of content as Viljanen noted in his study [10]. This could become a reason for user drop-off. As Yee stated, the feeling of 'power', that one is in control of the situation whether it be the opponent or the game is one of the biggest motivations for gaming. Thus, feeling of powerlessness would definitely become a factor for dropping off.

One of previous studies dealing with RPGLite was on using chained strategy generation to balance the application. Kavanaugh et al.'s initial strategy was to design the game so that each strategy becomes viable by 'countering', or having a strong win rate against another [4]. However, as game theory suggests, players will stick with the stable option that will bring the desirable output in most situations [10]. For example, if option A is advantageous against 7 strategies out of 10, while option B can oust option A but is advantageous against only 4 out of 10 strategies, then players will opt for option A.

Commercial gaming companies, especially those in the domain of esports, are highly sensitive of this concept of competitive balance. To provide spectators with a varying gaming landscape and prevent stagnation in the competitive scene, companies introduce patches on a regular basis. Patches are usually performed in the form of **buffs**, which strengthen a gaming material's numerical statistics, or **nerfs** which weaken the numerical statistics.

2.5 Related Works

Along with the growth of the video game industry and competitive esports, many researches have been performed to study the behavior of video game users. In 2015, there was a study that examined into people's inspiration for playing video games regardless of genre and culture. The study concluded that there were six types of motivations, that have their own different reason to play [11]. Although RPGLite is solely for the 'competitors' [4], there are possibilities that some other reasons that could have demotivated players from playing.

Furthermore, there have been attempts to try and predict user behavior in competitive video games. Research by Nascimento Junior involved clustering *League of Legends* players according to the stats they produce (kills, deaths, gold earned, towers taken, etc.) through the course of their game. A notable trait in this study is that although there are many stats involved, there is a high correlation coefficient between certain features. Furthermore, Nascimento discovered that certain behaviors in the game act as notable distinguishments between users of different skill levels [12].

Building on the idea that numerical statistics can be an indicator of skill, there were attempts to determine whether machine learning models can be used to predict outcomes of *League of Legends* matches. A study by Ani et al. attempted using ensemble methods to predict victory in professional esports matches, and achieved an accuracy of over 90% [13]. Another research by Do et al. applied the same principals for uncoordinated amateur settings where players do not have pre-match information about each other. Machine learning methods with an accuracy over 70% for predicting winners based only on quantitative metrics [14].

Building on the idea that victory in a competitive video game can be predicted on machine learning, this research attempts to discover metrics that could be used for finding a user's mastery based on basic information. Furthermore, this research tries to address competitive balance using machine learning by finding when to disrupt stagnating competitive balance.

Chapter 3 Methodology

3.1 Software and Hardware specifications

This project was performed with Python 3.8.5, Pandas 1.1.3, SQLite 2.6.0, with most of the analysis done on Scikit-learn 1.1.2 and MS Excel pivot tables.

The system used in this research was a gaming laptop with GTX 1650, Intel i7 and 16 GB RAM.

3.2 Data Preprocessing

Some of the features in the dataset were removed as they had little correlation to the variables that were involved in determining mastery or player involvement. Features such as playerid, selected and badge progression were removed. Skill points have been normalized for more efficient access and ease of modeling. Refer to Appendix for how each data has been removed.

3.3 Definition of user cohorts

This study focused on the behavior of three user groups. Due to the distribution of the users (Figure 2.2), the minimum threshold of the users to be considered for research was very low. The median of the number of games played by the users was 10, with a standard deviation of 122 games. Thus 10 was taken as the threshold.

The first group of users is the '**dropouts**'. They played more than 10 games, but they stopped playing before the patch 1.2 was applied. They were taken as sample for analysis of users who dropped out of the game for various reasons. Among the 370 players, 23 users were considered as dropouts.

The second group of users is the '**experts**', the players who have played from the first version to version 1.2., while keeping a win rate over 50%. These users were taken as samples of users whose mastery, whether macro or micro, have developed during the course of the game. There were 10 users who were chosen as the sample group of experts, in the order of most games played. During the course of the patch 1.2, the experts featured in 3362 matches.

The third group of users is the '**stagnants**', which is a group of players who have played from the first version until the latest, but with a win rate below 50%. These users represent a group of players who have failed to develop mastery despite the number of games that they have played. There were 10 users who were chosen as the stagnants, in the order of most games played. Throughout patch 1.2, the stagnants appeared in 3846 matches.

3.4 Machine learning methodology

As machine learning models were able to predict the success of teams in more complex games, they were also used to predict winners in RPGLite as well.

Examining the performance of different machine learning algorithms and feature importance could bring insight into the significance of metrics used to measure player performance in RPGLite. As the research is focused on predicting true positives (winners), the metric chosen to measure all machine learning algorithms was accuracy. Furthermore, linear regression was employed for attempts to predict trends of user behavior. For all machine learning algorithms, the dataset was divided into 70% training and 30% test set, with random stratified sampling with balance_code(patch version) as the stratification variable. Balance_code was chosen as the game's state actually changed after Kavanagh's 1.2 patch (Kavanagh).

3.4.1 Classification – k Nearest Neighbors

The k-Nearest Neighbors(kNN) algorithm compares a record with the k closest records in the same dataset. For this model, the number of neighbors used (n_neighbors parameter) was 3, as the sample size of selected players was rather small, and increasing neighbors above 4 decreased the accuracy.

3.4.2 Classification – Random Forest

The Random Forest (RF) algorithm is an ensemble method, or an algorithm that creates multiple decision trees to classify and combines the result into a single model. The advantage that Random Forest holds over decision trees is that Random Forests usually have less variance, and thus are less vulnerable to the risk of overfitting. The parameters for scikit-learn random forest in this experiment were 200 estimators with default setting as the sample size was not too large.

3.4.3 Classification – AdaBoost boosted support vector machine

AdaBoost (Adaptive Boosting) is another ensemble method that combines multiple weak decision tree classifiers to build a stronger classifier. Adaboost is applied with another classifier. In this research, support vector machine with the default gaussian radial basis function (RBF) kernel was used, with n_estimators =10 and learning rate 1.

3.4.4 Classification – Gaussian Naïve Bayes

Gaussian Naïve Bayes classifier is a classifier based on the Bayes theorem. Gaussian Naïve Bayes was used as there were 2 target labels (player 1 wins, player 2 wins) to predict, and the default values were used for the priors and var_smoothing parameters.

Chapter 4 Analysis

4.1 Overview

This chapter discusses detailed evaluation of user actions, analysis of metrics used for determining mastery. The analysis is guided by the questions below as well as by insights gained incrementally from tackling each of them:

RQ1: What are the demotivating factors in a competitive video game?

RQ2: What are the indicators of mastery in a competitive video game?

RQ3: Could machine learning be used to predict user success? If so, how could an efficient metric be designed to reflect mastery?

RQ4: Could there be a quantitative way to find patch timings?

RQ5: What are the other factors that affect players' decisions?

4.2 RQ1: What are the demotivating factors in a competitive video game?

Since RPGLite began service in April 3 2020, there were 243 among 370 users that did not play over 10 games, and 189 of them did not play a single game. This could have been a limitation from the player base, as this game was made for research purposes and most of the players were research students, their friends, or family members. Therefore, they could not be bothered to play the game after installation. This is supported by the evidence that although there is a feature to disable the push alarms that would be sent in case of a match [4], a majority of the players (177 players) did not disable this feature. There are chances that such users would have not even played a game of RPGLite in the first place.

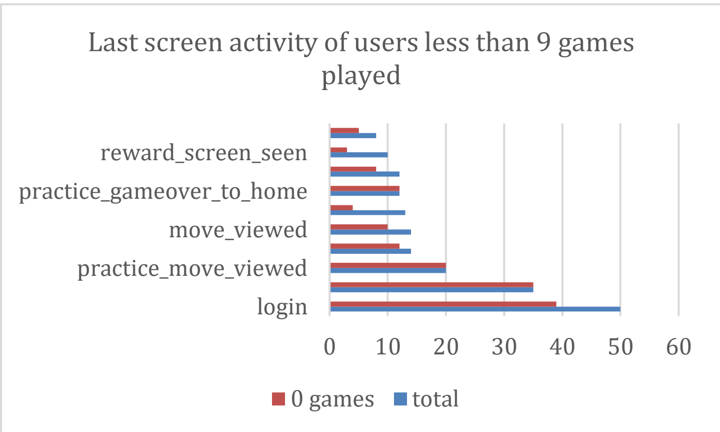


Figure 4.1 The last screen users that played less than 9 games have visited

The login screen, with a screen count of 50 in total, accounted for the highest number of last screens. Among the 50 users, 35 users have not played a single game of RPGLite, meaning that they have given up at the first screen due to lack of motivation or other reasons such as lost login credentials. Users who seemed to have lost their credentials did not attempt to reset their password, as there was only one user who accessed the change_password screen. Another notable fact from the screen data is that all the players who quitted on home_to_tutorial, practice_gameover_to_home and practice_move_viewed screens are players who played 0 games. Those three screens are screens related to the tutorial. Perhaps, the 67 players who stopped playing at those screens were the ones who found the game unappealing, whether by its looks or by its genre(turn-based).

For the players who have sufficiently engaged with RPGLite, the win rate and the feeling of 'power' might have been a reason to quit playing. The 54 players who played at least one game but did not progress over 10 games had an average win rate of 43%, winning only 83 games out of the 203 games they participated in. The 23 players who were considered as dropouts had an average win rate of 49%, with 10 of them having a win rate over 50%. The players who had win rates below 50% played around 10 more games on average, thus it could be inferred that the players gave up from exhaustion after efforts to try and improve their game.

On the other hand, players who quit the game despite having a good win rate over 50% could be inferred that they stopped due to boredom from repetition. As Yee stated, the feeling of 'mastery' and 'power' comes from the gamer overcoming difficult challenges the game materials present [3]. For example, the user 'BestWilliam', who had the highest win rate of 64% among the dropouts, played for a very short period of 3 days. Early depletion of content could be another reason as 'BestWilliam' had his greatest success with the Knight-Rogue combination and the Rogue-Wizard combination which accounted for more than half of his games. He played 7 of his last 10 games with the Rogue-Wizard and even his final game was a win with the same combination. It could be hypothesized that 'BestWilliam' left the game after concluding that he has found out the strongest strategy, and he has no more content to explore.

Summary: players' first motivation to play a game is personal taste, such as genre or aesthetics. After starting, players fall off due to disappointment from poor performance or fatigue from repetitive gameplay.

4.3 RQ2: What are the indicators of mastery in a competitive video game?

When a new video game is released, users explore through the game material. They test out different tactics, and modify them if the tactic seems ineffective. The users showed similar patterns in week 1 of RPGLite, as many different strategies were prevalent in both the groups. One difference that the expert users showed is that they had more tactical diversity (13 vs 10).

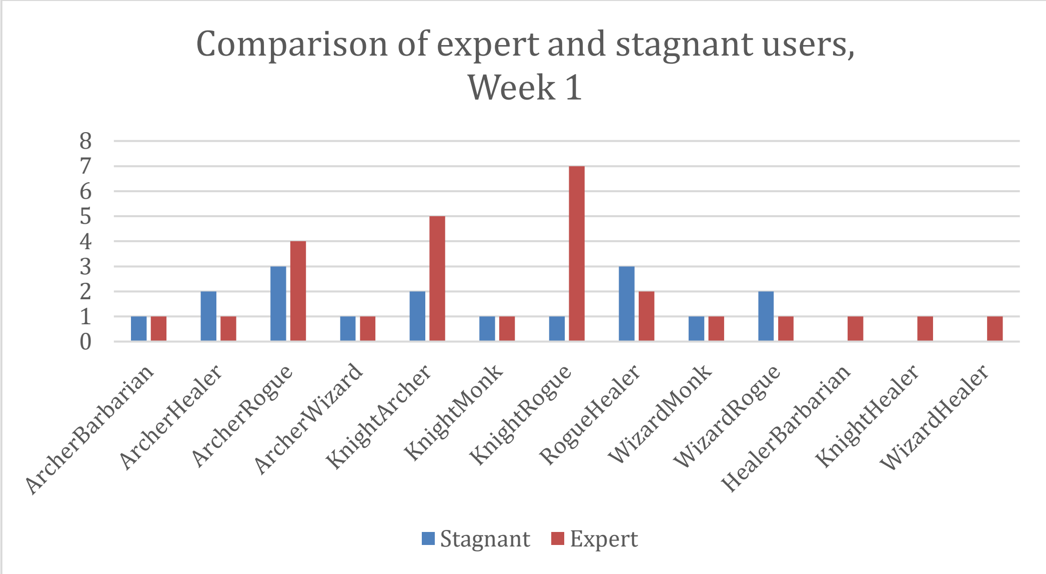


Figure 4.2 Choices made by the two user groups in Week 1 of RPGLite

As players progress throughout the game material, they figure out which strategies work out best in the current metagame. There were only 13 combinations used in week 1, but soon the users began to employ all 28 possible combinations from week 2 onwards. However, as RPGLite progressed, the users showed differences in employment of certain classes. This difference is highlighted in the following graphs, when the expert group show more than 150% or less than 50% when compared to the stagnant group.

The experts favored the Barbarian, Gunner and the Monk more than the stagnant players, while the stagnant players favored the Wizard. In both player groups, the Rogue was highly favored, with the Rogue-Monk combination appearing in 14% of games played in both groups.

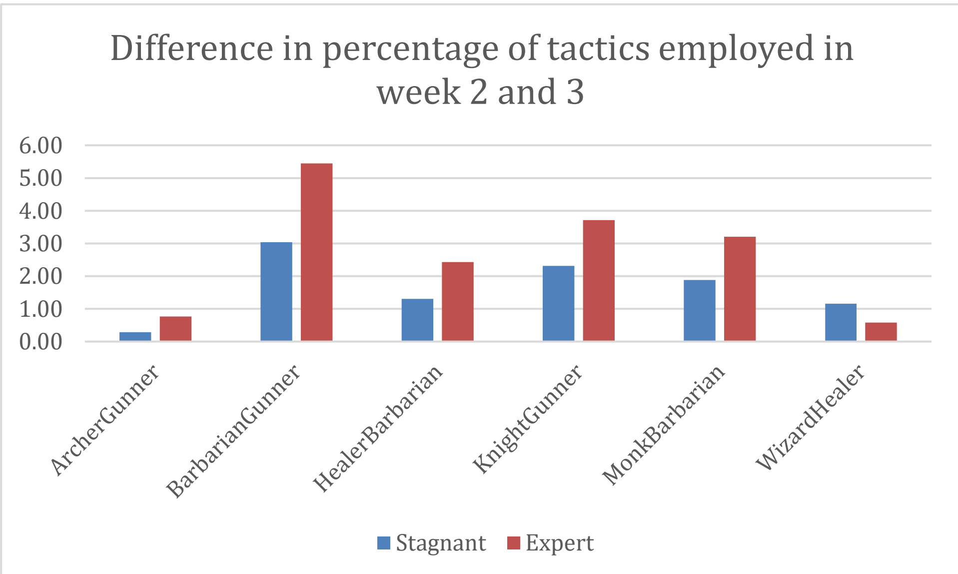


Figure 4.3 Week 2 and 3 metagame of the stagnant and expert users

In the last weeks of pre-1.2 RPGLite, the two player groups started to diverge. In weeks 2 and 3, only 6 combinations showed notable difference in preference (Figure

4.3). However, this difference increased to 9 combinations in weeks 4 and 5(Figure 4.4). The stagnant group of players preferred the Archer, who was the underdog of the patch with a win rate of 42% (412 games) from 974 games. On the contrary, the group of users that progress to become experts preferred the Barbarian, which was one of the stronger characters that appeared in 79% (2709 games) of the 3407 games and won 54% of them.

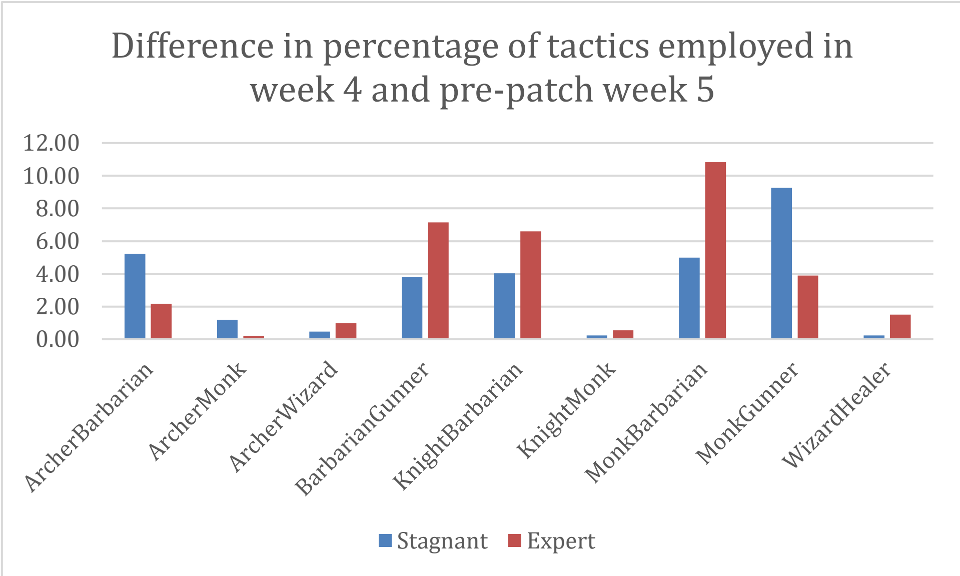


Figure 4.4 Week 4 and pre-patch week 5 metagame of the stagnant and expert users

After Week 5, RPGLite was patched to promote balance in pick rates. Due to the prevalence of the Rogue-Monk combination, there were measures taken to weaken those two classes, while also strengthen the archer, which was the biggest underdog of RPGLite before patch 1.2. Other classes such as the Barbarian and the Gunner were nerfed as well (Table 2.1). As patch 1.2 was announced, users explored the game material, increasing the number of combinations with notably different preferences to 13 (Figure 4.5), which is almost half of the 28 pairs possible.

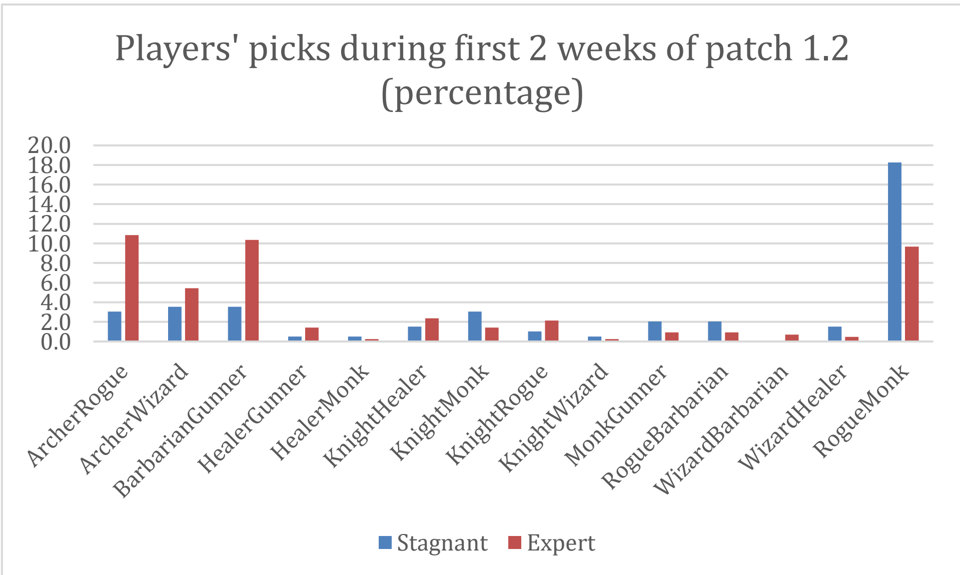


Figure 4.5 Contrast of players’ picks in the first two weeks of patch 1.2

The expert group were quick to adapt to the patches. The popularities of the pairs that contained the nerfed Monk class dwindled in both brackets. Before patch 1.2, the monk class appeared in 33% of games in the two brackets, but this percentage shrunk to 21% right after patch 1.2 was applied.

The two brackets of users adapted to the patch at different speeds. The bracket of expert users quickly disposed the Monk class in search for a new alternative, and they have turned to the Archer which received a buff in terms of stability while slightly deterred in its potential to deal damage to both the targets. The Archer-Rogue combination was popular, as it featured in 10.9% of the 424 games that were played by the expert group during that period. On the other hand, the bracket of stagnant users held on to the monk class. The popularity of the Rogue-Monk combination fell from 23.4% to 9.7% in the expert bracket, while it only experienced a slight decrease from 22.8% to 18.3% in the stagnant bracket.

On the other hand, the expert group were less reactive towards certain patches made in 1.2. Although the Gunner, the Barbarian and the Rogue were nerfed, these classes did not see much change in preferences. In fact, the Barbarian-Gunner combination saw a spike in pick rate from 7.2% to 10.4%. This combination remained as the most popular combination regardless of the bracket for 20 weeks until RPGLite ended service. Also, the Rogue class retained some of its popularity, being a part of the combination that would be the expert users' preference until the end of service. Another notable factor was that out of the 1002 games that the Barbarian-Gunner combination was in play, around 616 of the games involved users (mostly in the expert group) attacking first with the gunner, who has a higher expected damage value than the barbarian. A majority of those games (603 out 0f 616) ended in the victory of the Gunner-Barbarian pair.

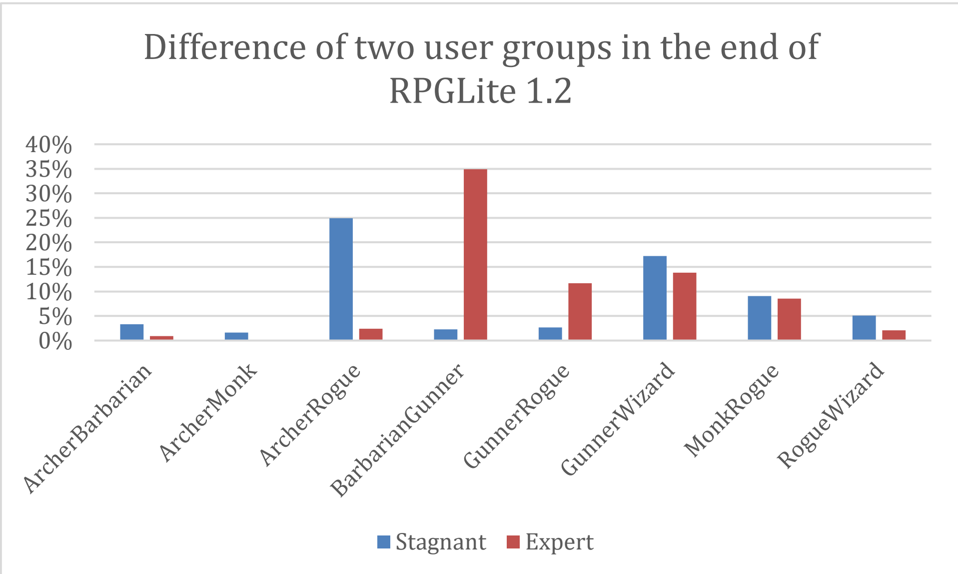


Figure 4.6 Difference in players' preferences in the end of RPGLite 1.2

By the end of RPGLite patch 1.2, there were eight notable combinations in the two user bases (Figure 4.6). Among them, the most notable combinations were the Archer-Rogue combination and the Barbarian-Gunner combination. Although the Archer-Rogue combination was popular in the early weeks of 1.2 patch, many users, especially the experts, have moved away from it. The Archer-Rogue combination was one with a low win rate of 44%. Kavanaugh et al. attributed this phenomenon to users' apathy coming from the long service time of RPGLite and lack of inspirations to win a game that provides no rewards [4].

Summary: Expert users tend to be more adventurous when exploring in-game choices. Users that develop mastery in a video game tend to be more attentive to their in-game performance, and change their choices accordingly.

4.4 RQ3: Could machine learning be used to predict user success? If so, how could an efficient metric be designed to reflect mastery?

4.4.1 Data preparation

To test the predictability of RPGLite match outcomes, machine learning methods ensemble methods were employed. The data was encoded under these hypotheses:

i) A skilled player with a more valid strategy for winning would have a higher chance to win
ii) The player with higher skill points determined with Kavanaugh et al.'s formula (Figure 2.9) would be considered as the 'stronger' player, with a higher chance to win
iii) The strategy that is more popular would have a higher chance to win, as users would intuitively prefer strategies that keep them winning.

Building on these assumptions, the players were encoded as their skill points (stronger players would have higher skill points) and the strategies were encoded as the popularity ranking (from 1~28) or the normalized value of the popularity ranking. The encoded model was tested using Random Forest (RF), AdaBoost, kNN and Gaussian Naïve Bayes.

As the skill point metric had close correlation with games won and played, and the players who had higher win rates tended to have higher skill points. However, encoding the character combinations came out as a challenge. Even with limited information, the players know by intuition that some strategies tend to be more effective than others.

Furthermore, players could be more confident with some strategies than others, as seen from the example of 'BestWilliam' (Appendix D) who was able to score a high win rate while keeping off the Rogue-Monk pair, which was the strongest of pre-1.2 patch RPGLite. This proposed another strategy in encoding the character pairs for modeling. Instead of a using a generic popularity ranking, the character pair would be encoded as the win rate of the player with that specific character pair, so that it represents the user's mastery with the chosen character pair.

Also, the balance code was introduced as a feature in the model to examine the significance of the version in determining the winners, while also evaluating the impact of the patch. In section 4.2, it was revealed that the players do react sensitively to the patch, as they started to prefer the buffed Archer over the nerfed Monk. With this in mind, if the balance code plays a large role in predicting the winner, then it would mean that the patch made a significant change in the game.

Otherwise, it would mean that players did not play according to the makers' patch intentions and played regardless of the version and the patch.

4.4.2 Results of machine learning methods

Method	One-hot encoding	Popularity	Win rate
RF	52.90%	53.40%	62.89%
AdaBoost	49.23%	50.41%	49.60%
kNN	51.80%	52.90%	54.25%
Naïve Bayes	51.84%	53.67%	64.50%

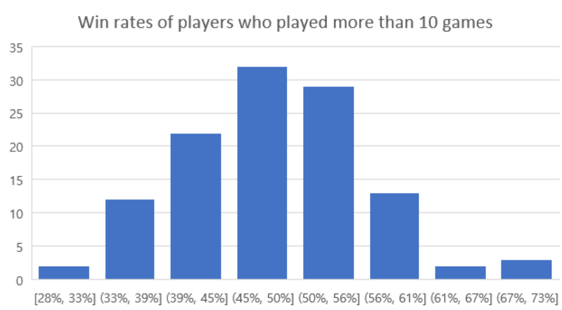
Table 4.1 Performance of different machine learning models in predicting RPGLite winners

Machine learning models had a performance around 50~65% accuracy when predicting the winner of a RPGLite match (Table 4.1). Looking at the feature importance table (Table 4.2), it could be seen that the skill points of the players play the biggest role for the classifier to determine the winner.

Feature	One-hot encoding	Popularity	Win rate
Balance code	0.02	0.02	0.03
P1 skill points	0.30	0.31	0.17
P2 skill points	0.31	0.31	0.18
P1 pair points	0.18	0.19	0.31
P2 pair points	0.19	0.18	0.32

Table 4.2 Performance of different machine learning models in predicting RPGLite winners

However, when the character pairs were encoded as the players' win rates with the particular pair, there was a change in feature importance. The feature importance between the skill points and the player's pair win rate changed. This could be attributed to the fact that the player's win rate with the chosen character pair is a more personalized variable than the other two. Furthermore, the performance of the random forest classifier improved by around 10%, stating that the new metrics are better indicators of distinguishing the players' skills.



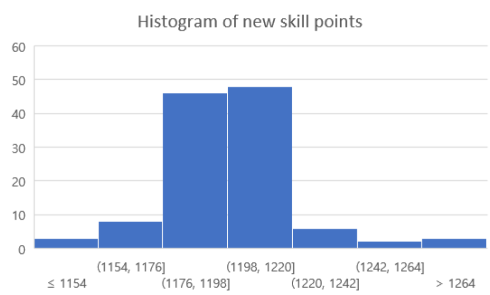


Figure 4.7 Histogram of distributions of players' win rates and newly calculated skill points

Among the machine learning algorithms, Gaussian Naïve Bayes performed the best when it had the player's pair win-rate as the skill score for character pairs. This could be due to the fact that Gaussian Naïve Bayes works best when the data is in a Gaussian distribution as seen in Figure 4.7. Furthermore, it could be due to the fact that the assumptions of the hypotheses are quite similar to a Naïve Bayesian assumption that prior probabilities can explain future trends.

4.4.3 Are skill points reliable?

As seen from Table 4.3, the skill point system proposed by Kavanaugh et al. [4] had little correlation with actual win rate, despite having high correlation coefficient with the number of matches played and won. The skill point system of RPGLite works more in favor of players who have played more matches than those who are more likely to win. For example, 'Becccccca' who had a win rate of around 40% (won 90 out of 227 matches) had a skill point of 5635, much higher than 'Ezzey' who had 3710(who has a win rate of 60% by winning 90 of 150 games).

After performing correlational analysis using Pearson correlational coefficient, it has been discovered that the skill point system devised by Kavanaugh et al. (Figure 2.9) has a high correlation to number of matches won and number of games played, while not having a high correlation rate to the win rate of players. On the other hand, the traditional ELO score did not have much correlation with any other variables such as wins, win rates, and matches played, and thus it was discarded.

	wins	matches	Win rate	Skill points	ELO
wins		0.99	0.17	0.98	0.20
matches	0.99		0.12	0.97	0.12
Win rate	0.17	0.12		0.16	0.19
Skill points	0.98	0.97	0.16		0.13
ELO	0.20	0.12	0.19	0.13	

Table 4.3 Pearson correlation coefficient of wins, matches, win rate, skill points and ELO

Therefore, it could be said that the skill point system was based on the assumption that more games would lead in deeper understanding of the game. However, as RPGLite did not have a more active mechanism to reward the player for being competitive. While designing RPGLite, Kavanaugh et al. kept in mind that "all agents to be driven solely by competition" [4].

One possible way to keep the players' attention and motivate them to win is to punish for the mistakes. As Kavanaugh et al.'s formula in figure 2.9 does not punish losses hard (a win gives 4 times the skill points lost in a loss), a tentative equation for skill points had to be derived. The new skill point equation had a correlation coefficient of 0.49, which is higher than 0.16 of Kavanaugh's. As Kavanaugh's initial skill points did not greatly penalize or reward the outcomes of matches against opponents with skill difference, skill difference was neglected.

$$skill\ point = 1200 + (winrate * wins) - (lossrate * losses * 2)$$

Figure 4.8 Newly devised equation for skill points, penalizing the losses harder

A classification experiment with the new skill point equation (Figure 4.8) was performed, yielding the following results:

Method	One-hot encoding	Popularity	Win rate
RF	51.70%	53.89%	61.59%
AdaBoost	51.52%	49.91%	51.43%
kNN	53.13%	51.66%	55.15%
Naïve Bayes	59.85%	53.85%	65.84%

Table 4.4 Performance of different machine learning models in predicting RPGLite winners with the new skill point formula

As the new metrics were more reflective of the win rates, the performance of the models improved slightly (Table 4.4). However, the gap between the importance of features did not change much. It could be due to the fact that the character pair points are more reflective of the player's skill, as it literally is the player's win rate with that character pair.

Feature	New skill point formula	Initial skill point formula
Balance code	0.02	0.03
P1 skill points	0.17	0.17
P2 skill points	0.18	0.18
P1 pair points	0.31	0.31
P2 pair points	0.32	0.31

Table 4.5 Feature importance in random forest models before and after changing skill point equation (with the players' win rate as the character pair score)

Summary: Winners of a RPGLite match can be predicted using machine learning techniques, with the players' skill metrics and character pair metric as the features. Among the techniques, Gaussian Naïve-Bayes works best. The models work best when the players' win rates with the specific character pair is used as the character metric. Furthermore, a more reflective skill point metric can improve model accuracy

4.5 RQ4: Could there be a quantitative way to find patch timings?

After patch 1.2, RPGLite was in service for around 20 weeks without changes. During that time period, only four strategies (Archer-Monk, Archer-Rogue, Barbarian-Gunner, Gunner-Wizard) out of 28 available did not reach the weekly pick rate of zero.

A stagnating game could lead to user exits and loss of interests. Therefore, if machine learning based system could warn the game operators to implement a patch before stagnation happens, it would help the operators' business. Therefore, a method based on linear regression was hypothesized and implemented. This yielded different results. Trends of some character pairs were able to be predicted, while other pairs were not.

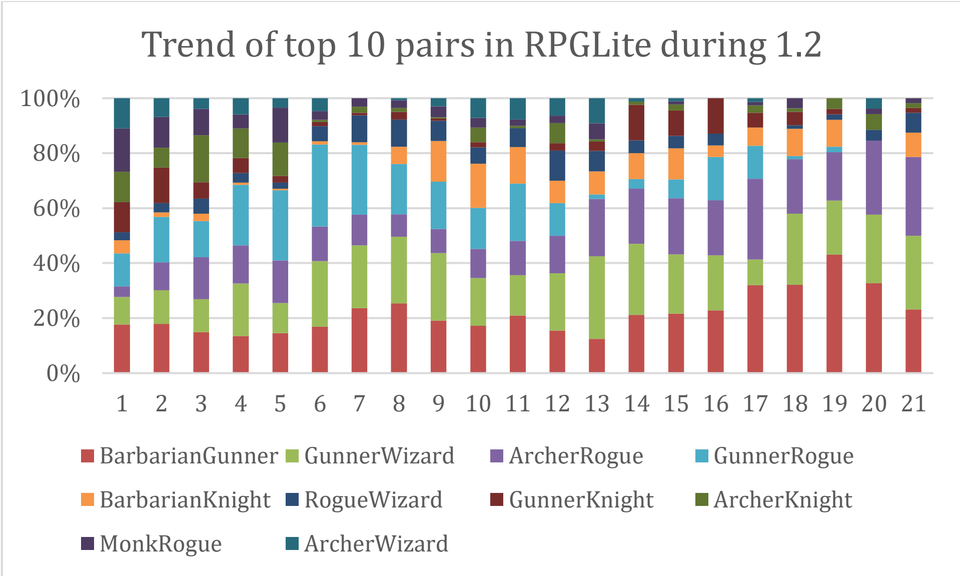


Figure 4.9: Trend of top 10 most popular character pairs during RPGLite v1.2. The row number is the service week of RPGLite v1.2

In Figure 4.9, it could be seen that some strategies such as Barbarian-Gunner stay dominant throughout the service, while some strategies such as Gunner-Knight disappear (week 11) but sometimes show a high spike in (week 14). Machine learning was utilized to try and forecast such trends

The linear regression model only took time and the pick percentage of the character pair as the parameters. It was assumed that time would be the parameter which speaks for the developing knowledge of users. In the experiment, the linear regression model was used to predict trends after week 24(week 7 of patch 1.2), because week 25 was the week that five character pairs were out of the competition and the operators would have been warned of stagnation.

In the experiment, some character pairs such as the Gunner-Knight pair that was consistent but not prevalent were predictable. The Gunner-Knight pair almost saw extinction in week 25, with its pick percentage dropping to 0.76 percent. However, this pair saw a short resurgence in week 26, with a pick rate of 2.22 percent but eventually saw brief extinction in week 29.

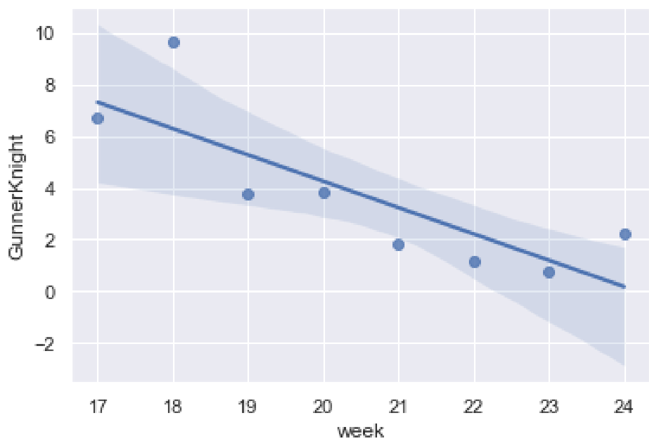


Figure 4.10: Linear regression plot of the Gunner-Knight character pair. Y axis is the pick rate. Week 17 is the starting week of patch 1.2

On the other hand, the Archer-Rogue pair was hard to explain with linear regression. Despite the outliers, the linear regressor predicted the Archer-Rogue combination to increase. However, it had a slight decrease, and spent the next 6 weeks in the 6 to 9 percent range before spiking up to 11 at week 28.

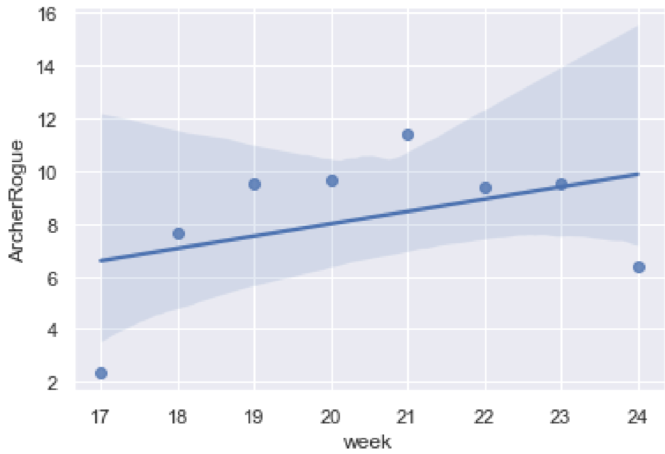


Figure 4.11: Linear regression plot of the Archer-Rogue character pair. Y axis is the pick rate. Week 17 is the starting week of patch 1.2

Although the exact reason of such phenomenon could not be traced to a single reason, one of the most probable causes could be the lack of information among the users. Popular competitive games have websites where users can access dashboards that provide information about the state of the game. However, RPGLite does not have such features. Those sites also provide features that a player can track his or her performance records in detail, but as mentioned, RPGLite lacks such features. Therefore, users do not have the information about what tactic is dominant and would have to rely on their intuition to form strategies. Figures 2.5 and 2.6 show that the top 10 most popular character pairs are not always the most successful ones. The users are not as inclined to streamline their play as quickly as gamers of other commercial games.

Summary: Linear regression on RPGLite character popularity trends failed due to players' inconsistency in streamlining their plays. This inconsistency could be due to players' lack of access to play feedbacks and features that provide feedback.

4.6 RQ5: What are other factors that affect players' decisions?

As seen in the previous section, there were several unexpected behaviors made by the users. Some character pairs were very popular despite the win rate, while other character pairs were unpopular even with a high win rate. The assumption that users made such decisions was due to intuition and fatigue will be discussed.

4.6.1 Why was Archer-Rogue popular?

To balance out the game, Kavanaugh introduced patch 1.2 at May 1st, 2020. He introduced various nerfs and buffs to the game, in order to make some more options viable. However, the results were mixed, with some nerfed classes still remaining dominant while the Healer and the Archer remained weak.

Class	Expected Damage	Expected damage (1.2)	Special mechanism
Knight	2.4	2.4	none
Archer	3.4(double) / 1.7	3.2(double) / 1.6	Can attack both targets
Wizard	1.6	1.7	Makes target skip turn when attack hits (stun)
Healer	1.7	1.8	Heals self and ally for 1 hp when attack hits
Rogue	2.15, 3.75(execute)	2.1, 3.5(execute)	Deals 5 damage to targets under 5 health (execute)
Monk	0.8, with 50% chance of being better than a Knight	0.75, with 40% chance being better than a Knight	Can strike again if attack hits target
Barbarian	2.15, 3.75(rage)	2.1, 3.5(rage)	Hits for 5 damage when below 5 health (rage)
Gunner	3.25	3.1	Deals 1 damage instead of missing (graze)

Table 4.6 Expected damage outputs of the 8 Classes of RPGLite

To summarize the patch, the Healer and the Archer received buffs, the Knight and the Wizard stayed the same, while all other classes were nerfed (Table 4.6).

4.6.2 Was the Archer buffed enough?

In patch 1.2, the Archer received a both a buff and a nerf to make it more relevant in the game. It was buffed in terms of survivability with 1 added health. It received a nerf in terms of damage with its expected damage output decreasing from 3.4 to 3.2, assuming that it hits both targets as intended. In 1.2, this yielded mixed results, with the Archer-Rogue combination becoming one of the most popular combinations in v1.2(Figure 2.6) while being one of the less successful ones (win rate of 45%)

However, when calculating expected damage output, this nerf is quite trivial, as the archer would only deal 2 less damage in 10 turns (even less in practice as the first elimination occurs around turn 5 to 6). But the Archer's base damage itself quite weak to begin with. In theory, the Archer would need around 5 turns to take out the Monk, the most fragile character in RPGLite (Table 4.7). Even after succeeding the first elimination, the Archer becomes a liability, with the weakest single target damage potential in the 1.2 patch and no utility.

	Monk(7HP)	Knight(10HP)
Archer, expected value (1.6)	8 damage dealt with 5 attacks	10 damage dealt with 7 attacks

Archer, assuming all hits (2)	8 damage dealt with 4 attacks	10 damage dealt with 5 attacks
Gunner, expected value (3.1)	9 damage dealt with 3 attacks	12 damage dealt with 4 attacks
Gunner, assuming all hits (4)	8 damage dealt with 2 attacks	12 damage dealt with 3 attacks

Table 4.7 Expected damage calculation of Archer vs Gunner against Monk and Knight

The buff given to the Archer was a 1 health buff, to prevent the Archer from dying too early in the early game when it was supposed to be at the peak of its damage potential when it has 2 targets to hit. This buff did not work out as intended, as the average number of turns of games than involved the Archer decreased by an average of 1 turn, while also the health difference in wins decreased by 1 point. (Appendix E)

However, patch 1.2 was successful in strengthening the Archer as Archer-Knight and Archer-Healer became two of the top 10 most successful character pairs in RPGLite v1.2 (Figure 2.6). However, the more popular combination was the Archer-Rogue, which featured in 1093 games, around 25% of games in RPGLite 1.2 despite a bad win rate of 44%.

4.6.3 Intuition and Archer-Rogue

There were other more successful pairs including the Archer such as the Archer-Knight pair with 55%-win rate, and the Archer-Healer pair with 53%-win rate. However, many users preferred to play Archer-Rogue instead. The difference in preference is prevalent when compared to Archer-Healer, which featured in only 193 games.

Character Pair	Average Game Length (turns)	Average win rate	Games played (popularity rank)
Gunner-Rogue	11	54%	958(4)
Archer-Rogue	11	45%	1093(3)
Barbarian-Gunner	12	57%	1283(1)
Archer-Gunner	12	39%	156(14)
Gunner-Wizard	12	52%	1099(2)
Archer-Wizard	12	48%	206(11)
Rogue-Wizard	12	45%	264(8)
Gunner-Knight	12	55%	268(7)
Archer-Barbarian	13	48%	216(10)
Knight-Rogue	13	40%	160(13)

Table 4.8 Top 10 character pairs with shortest game length and their win rate and popularity

This could be attributed to the fact that the Archer-Rogue combination has the shortest average length of games (Table 4.8). However, a majority (58.7%, 642 out of 1093) of these users did make the right move to attack first with the Archer, who has a higher expected damage value than the Rogue when there are 2 targets alive (Table 4.6)

In his study, Kavanaugh attributed this unexpected popularity of the Archer-Rogue combination to "apathy" [4]. As the RPGLite experiment carried on, users were pressured to play (although their motifs are unknown, whether it be from interpersonal pressure or from academic obligations). The expression "apathy" could be accurate as the Archer-Rogue pair is the pair with the shortest game span, and the users are exploiting that fact just to fill the required number of games played.

A notable trend is that most character pairs with the shortest game spans are ranking high in terms of popularity as well, as all but 3 pairs made it in the top 10 in terms of popularity. The Pearson correlation coefficient between average game span and popularity ranking (with the most popular Barbarian-Gunner as 1) is 0.67, stating that there is a positive correlation between the two. Comparing with the Pearson coefficient between popularity ranking and win rates of -0.49, it could be inferred that the popularity is more about how quickly the game ends rather than the actual win rate.

To test this hypothesis, a multivariate linear model has been made as popularity ranking (with 1 as the most popular) as the dependent variable and the percentage win rate and game span as the independent variables (both underwent min-max normalization). The coefficients for the model were 17.69 for average turns, and -12.28 for percentage win rate, confirming the assumption that the longer game span makes the pair less popular, while higher win rates make the pair more popular, but with the former having more impact (as seen from the higher coefficient).

Furthermore, the impression that 'faster game length' leads to 'more sense of power' could also be in consideration. Regardless of outcome, the user might perceive an early win as a sign of power than a win after a long and tenuous struggle. Furthermore, all the other competitors are employing quick game combinations such as Gunner-Rogue, Barbarian-Gunner and Gunner-Wizard that end games in 12~13 turns as well, so the difference in game length would not come as an issue.

A comparable combination with the Archer-Rogue pair is the Gunner-Rogue pair, which has a 9% higher win rate while having a similar number of games featured (1093 vs 958). This pair was widely favored in the expert bracket while overlooked by the stagnant users (Figure 4.6). Although exact reasons are unknown, it could be expected that the expert users had knowledge of expected damage outcomes (Table 4.6), and took advantage of the fact that the Gunner can help the Rogue meet the execute threshold when the attack hits (4 damage puts 4 classes into the execute threshold, which is below 5 health).

4.6.4 Why was the Healer unpopular?

Along with the Archer, the Healer class was another weak class in RPGLite, and has received buffs in version 1.2. The Archer-Healer combination has become the fifth most successful pairs with a win rate of 53%, but none of the character pairs including the Healer has made into the top 10 most popular, with the aforementioned Archer-Healer ranking highest as 13th.

When the Healer is included, the games tend to get longer. It is due to the fact that the Healer is one of the classes with the lowest damage potential, while has an ability to heal ally for 1 health. Therefore, damage dealt to the opponents is low, while damage dealt to allies will be mitigated. This is evidenced by the fact that the games without the healer usually end in 15 turns, while the games with healer have an average game length of 17 turns (Appendix E).

Character Pair	**Average Game Length (turns)**	**Average win rate**	**Games played (popularity rank)**	**Average Health point difference**
Archer-Healer	14	53%	193(12)	6
Gunner-Healer	14	50%	124(15)	6
Healer-Knight	17	47%	115(17)	7
Healer-Rogue	15	38%	104(18)	9
Barbarian-Healer	17	49%	90(21)	6
Healer-Wizard	18	40%	47(27)	6
Healer-Monk	21	41%	27(28)	8

Table 4.9 Character pairs which include the Healer and their popularity and win rate

However, whether the Healer can change the outcome of the game is another question. All other compositions with the Healer other than Archer-Healer has no win rate higher than 50% (Table 4.9). Furthermore, the assumption that 'shorter game length is a more powerful combination' holds somewhat true in this case, as the less popular and less successful character pairs tend to have lower win rates as well.

The problem of the Healer's unpopularity and unsuccessfulness could stem from the fact that it is weak in concept. Successful classes such as the Rogue has classes it can synergize well with (Monk in pre-1.2, Gunner in 1.2), or has a strength of its own like the Gunner (highest unconditional damage output) or the Barbarian (good health combined with high damage potential). However, the Healer's concept is 'irritating', but not 'powerful' at best. As seen in figure 2.7, the majority of RPGLite games end with around 4~5 health difference, which means that each character has around 2~3 health points. However, the most popular characters such as the Gunner, Barbarian and Rogue have an expected damage output around 3 and actual output of at least 4, making the Healer's efforts futile. Adding 1 health to 3 will not save anyone from the Gunner's 4 damage, and definitely not from 5 damage from the Rogue's execute or the Barbarian's rage.

Summary: Without the help of a tool to evaluate their play, players rely on intuition to gauge the strength of their strategy. Match length is a scale of power players often use aside from the actual outcomes of the matches. Some players actually chose quick results and less fatigue over actual win rates.

Chapter 5 Limitations and future work

The biggest limitation was that the game RPGLite was out of service at 2022, and thus first-hand knowledge of the game has become unobtainable. Gameplay could only be hypothesized based on log data, and thus could not be observed firsthand. Furthermore, as user survey has become impossible, so no information about human-computer interaction or competitive gameplay from the users' perspective was able to be surveyed.

Furthermore, the log data lacked some detail. For example, there was no way to find out whether the Archer attacked both the targets or only one target. Even in Kavanaugh's research paper, it was written that "we assume that" player will always target both opponents in the aforementioned situation. Therefore, some indicators of 'micro' level mastery could not be observed.

If the game was still being serviced, other machine learning techniques such as reinforcement learning could have been utilized to see how AI deals with mastery. Reinforcement learning could also provide insights into 'micro' level mastery about optimal ways to operate each character pair. This could open up further research about 'high-level techniques' that users are not able to discover without much experience. An example of such technique would be the technique of skipping a turn when the Barbarian is the sole survivor, as taking damage would trigger rage and enable the Barbarian to eliminate an enemy in the next turn [15].

5.1 Log data analysis for micro control techniques

In Kavanaugh and Miller's research, there were mentions about better decisions using game mechanics such as the aforementioned rage technique. By analyzing the log data further into detail, perhaps a study about 'micro control techniques', or minute tactics employed to switch the game's situation in favorable ways (i.e., chess openings [9]), in RPGLite could be performed.

Therefore, research of those techniques could also be a good topic in mastery. In Do's work, there was a mention of champion mastery scores built in *League of Legends* that had high correlation coefficient with the win rates, and thus made the machine learning model more accurate with predictions [14]. Perhaps the similar could be done with the usage of other mechanics such as Archer's double attack or other special abilities.

5.2 Further development of metrics

Although RPGLite does not have too much room for micro control, there might be users more confident with dealing with a particular character pair than others. (For example, shooting down the Monk first when facing a Monk-Rogue pair, dealing n damage with an Archer, etc.) If win-lose statistics can be more detailed, then new metrics could be formed by inclusion of a user's win rate against the opponent's pair.

Chapter 6 Conclusion and personal reflection

This project employed quantitative machine learning methods to address the question of mastery in video games, and factors that could distract the users from developing mastery.

For a video game, keeping users engaged and providing them feedback is important for user retention. Mastery is a reason for users to play a certain video game, as competition is one of the largest motivation factors for gamers. Competitive gamers aspire to be good at their game, and if they are engaged sufficiently, they would immerse and become static users. However, if the game fails to engage them with suitable challenges such as failure in difficulty control, lack of challenges or lack of content to explore, then the users will drop off. Also, the gamers would have to be more attentive to their plays to be better, rather than rely on intuition alone.

Mastery of gamers can be measured in both qualitative and quantitative ways, as the more masterful users have a different understanding of the game material. More skillful users tend to explore more actively with the game material, and they tend to rely less on intuition when playing. There might also be a psychological difference, as they tend to be more patient and tenacious, and tenacity is definitely something to do with mastery as more plays do correlate to higher mastery. Sufficiently accurate metrics to measure users' understanding of the game material, such as win rates, can lead to accurate prediction of match outcomes involving machine learning techniques.

Furthermore, I was able to delve more into the concept of 'quantitative analyses', and how it would apply in other domains. I have learnt how to make as much inferences and hypotheses from a limited dataset, and gained insights to drawing conclusion from numerical data alone. On a personal side, as a gamer and esports fan myself, I have found this project very relatable and enjoyed this project. I was able to understand why character-by-character mastery scores and metrics in video games such as *DOTA 2* or *League of Legends* exist.

Appendix A RPGLite Dataset details

This appendix contains information about the dataset and how they were used

Table A.1 Player Table from RPGLite Dataset

Data	**Explanation and usage**
Username	Name of the player. Used to track individual playstyles.
Played	Number of games played, used for modeling and correlational analysis
Won	Number of games won, used for modeling and correlational analysis
tag_bg	Background used on player tag, was not used.
Games	Most recently captured game, not used.
Count_character_has_won	Number of wins with character classes. Not used.
Count_character_has _been_played	Number of plays with each character class. Not used.
Accepting_games	Whether or not the player is accepting game invitations. Used for behavior analysis
Skill_points	Skill points of the player. Used for segmentation and machine learning modeling
Badge_progressions	Information about player achievement (badges). Not used.
elo	ELO score used in initial study. Used for correlational study.
Lost_against	Players the player has lost against. Not used.
Current_season_skill	Skill points earned in season. Not used.

Table A.2 Games Table from RPGLite Dataset

Data	**Explanation and usage**
P1	ID of player 1. Used for filtering games according to segment.
P2	ID of player 2. Used for filtering games according to segment
P1_selected	Whether player 1 has selected characters. Not used.
P2_selected	Whether player 2 has selected characters. Not used.
Moves	Moves made throughout the game. Used for tracking player behavior.
Active_player	Player who has the turn. Not used.
P1c1	Player 1 character 1. Used for strategy tracking and modeling
P1c2	Player 1 character 2. Used for strategy tracking and modeling
P2c1	Player 2 character 1. Used for strategy tracking and modeling
P2c2	Player 2 character 2. Used for strategy tracking and modeling
P1c1_health	Health of player 1 character 1. Used for gameplay analysis
P1c2_health	Health of player 1 character 2. Used for gameplay analysis
P2c1_health	Health of player 2 character 1. Used for gameplay analysis
P2c2_health	Health of player 2 character 2. Used for gameplay analysis
usernames	IDs of the players currently involved

Table A.3 Page Hit Table from RPGLite Dataset

Data	**Explanation and usage**
User	User who is accessing a specific screen. Used for behavior analysis.
Kind	The screen that the player has accessed. Used for behavior analysis.
time	Time that the user has accessed the specific screen. Used for behavior analysis.

Appendix B List of players by segments

Table B.1 Table of Dropouts (users who played more than 10 games, but quit during or before second week of patch 1.2 (2020-05-15)

Username	Played	Won	Last Login date
cwallis	189	93	2020-04-27 19:10:39.240000
charlotte	85	34	2020-04-30 18:09:03.054000
bladestoe	57	24	2020-04-20 12:32:23.242000
EarlofSurl	56	31	2020-04-12 18:34:12.302000
norgart	51	23	2020-04-24 23:01:23.620000
BestWilliam	50	32	2020-04-08 09:18:38.737000
Mageofheart	36	14	2020-04-08 07:36:14.388000
versatile	32	17	2020-04-14 16:16:10.974000
meow	32	17	2020-04-17 07:36:55.618000
sidb	30	16	2020-04-14 09:26:39.276000
OKaemii	29	11	2020-04-23 21:32:25.739000
asdf	27	16	2020-04-17 10:38:01.330000
Alliyana	24	11	2020-04-19 13:34:14.023000
totem37	23	12	2020-04-16 18:48:19.612000
Paulverise	21	12	2020-04-10 10:05:39.923000
Rarno	21	10	2020-04-10 12:09:09.465000
cats	19	11	2020-04-14 07:00:13.671000
Kjreid	15	6	2020-04-16 05:08:32.693000
bcslippers	14	6	2020-04-13 17:52:45.775000
yasmin_f	14	6	2020-04-16 17:03:48.425000
Jules_217	13	7	2020-04-13 10:27:17.123000
Felix42	13	6	2020-04-13 21:12:10.070000
mosam1311	10	5	2020-04-27 16:08:41.266000
Ezzey	90	150	2020-05-08 22:47:03.503000

Table B.2 Table of Experts, players who have played since the initial release and played till the end of service while maintaining a win rate above 50%

Username	Games played	Win rate
apropos0	989	53%
basta	267	58%
Frp97	725	66%
Jhannah	110	59%
kubajj	210	56%
Lotua	24	74%
Luca1802	405	61%
probablytom	263	57%
TheMaster	16	74%
timri	425	71%

Table B.3 Table of Stagnants, players who have played since the initial release and played till the end of service but never reached a win rate over 50%

Username	Games played	win rate
l17r	906	45%
Deanerbeck	682	48%
ECDr	561	49%
cptKav	498	47%
DX13	328	48%
Becccccca	228	40%
dalinar	153	48%
DavetheRave	144	44%
Beth	139	47%
alan	138	41%

Appendix C Preference of strategy in the two brackets after v1.2

Character Pair	stagnants	experts
Archer-Barbarian	3%	1%
Archer-Gunner	2%	1%
Archer-Healer	3%	1%
Archer-Knight	3%	1%
Archer-Monk	2%	0%
Archer-Rogue	25%	2%
Archer-Wizard	2%	1%
Barbarian-Gunner	2%	35%
Barbarian-Healer	1%	1%
Barbarian-Knight	2%	5%
Barbarian-Monk	4%	1%
Barbarian-Rogue	1%	1%
Barbarian-Wizard	1%	1%
Gunner-Healer	2%	1%
Gunner-Knight	2%	2%
Gunner-Monk	1%	1%
Gunner-Rogue	3%	12%
Gunner-Wizard	17%	14%
Healer-Knight	1%	3%
Healer-Monk	0%	0%
Healer-Rogue	1%	1%
Healer-Wizard	1%	0%
Knight-Monk	1%	1%
Knight-Rogue	3%	0%
Knight-Wizard	1%	1%
Monk-Rogue	9%	9%
Monk-Wizard	1%	3%
Rogue-Wizard	5%	2%

Appendix D BestWilliam's Play style

Table D.1, BestWilliam's preference of character pairs

Character pair	wins	matches
KnightRogue	14	23
RogueWizard	8	13
BarbarianKnight	3	3
ArcherHealer	0	1
ArcherKnight	1	1
GunnerRogue	1	1
HealerKnight	1	1
HealerMonk	1	1
HealerRogue	0	1
HealerWizard	1	1
MonkRogue	1	1
MonkWizard	1	1

Appendix E Average game length and health difference

Table E.1, Average turn length and health difference of winning games that include the Archer before and after version 1.2

Character Pair	Turns (pre-v1.2)	Health(pre-v1.2)	Turns (1.2)	Health(1.2)
ArcherBarbarian	13.0	5.0	13.0	5.0
ArcherGunner	11.0	6.0	11.0	6.0
ArcherHealer	14.0	7.0	14.0	6.0
ArcherKnight	13.0	6.0	13.0	5.0
ArcherMonk	18.0	6.0	17.0	4.0
ArcherRogue	12.0	5.0	11.0	5.0
ArcherWizard	15.0	6.0	12.0	5.0

Table E.2, Average turn length and health difference of winning games that include the Healer after version 1.2

Character Pair	**Turns**	**Health**
ArcherHealer	14.0	6.0
BarbarianHealer	16.0	5.0
GunnerHealer	14.0	5.0
HealerKnight	18.0	7.0
HealerMonk	22.0	7.0
HealerRogue	14.0	6.0
HealerWizard	21.0	8.0
Average	17	6

Table E.3, Average turn length and health difference of winning games that do not include the Healer after version 1.2

Character Pair	**Turns**	**Health**
ArcherBarbarian	13.0	5.0
ArcherGunner	11.0	6.0
ArcherKnight	13.0	5.0
ArcherMonk	17.0	4.0
ArcherRogue	11.0	5.0
ArcherWizard	12.0	5.0
BarbarianGunner	11.0	4.0
BarbarianKnight	14.0	5.0
BarbarianMonk	18.0	4.0
BarbarianRogue	13.0	4.0
BarbarianWizard	15.0	4.0
GunnerKnight	12.0	6.0
GunnerMonk	15.0	5.0
GunnerRogue	10.0	6.0
GunnerWizard	13.0	6.0
KnightMonk	18.0	4.0
KnightRogue	12.0	5.0
KnightWizard	17.0	6.0
MonkRogue	16.0	7.0
MonkWizard	23.0	7.0
RogueWizard	13.0	6.0
Average	14	5

Bibliography

[1] SteamCharts.com.2022. Steam Charts-Tracking What's Played (2022). Retrieved July 3, 2022 from https://steamcharts.com/

[2] Will Freeman. 2016. Atari co-founder: Mobile games make me want to throw my phone. (April 2016). Retrieved June 27, 2022 from https://www.theguardian.com/technology/2016/apr/28/atari-co-founder-nolan-bushnell-mobile-games-make-me-want-to-throw-my-phone

[3] Nick Yee. 2016. The Gamer Motivation Profile. *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play* (2016). DOI: https://doi.org/10.1145/2967934.2967937

[4] W. Kavanagh, A. Miller, G. Norman and O. Andrei, "Balancing Turn-Based Games With Chained Strategy Generation," in *IEEE Transactions on Games*, vol. 13, no. 2, pp. 113-122, June 2021, DOI: 10.1109/TG.2019.2943227.

[5] W. J. Kavanagh, W. Wallis, and A. Miller, Rpglite player data and lookup tables, http://researchdata.gla.ac.uk/1070/, 2020. DOI: 10.5525/gla.researchdata.1070.

[6] Marc Hammes.2022. What is the average match time in Dota 2? (April 2022). Retrieved August 2, 2022 from https://theglobalgaming.com/gaming/average-match-time-dota-2

[7] Games of Legends.2022. Games of Legends - Pro Games Statistics for LoL (2022). Retrieved July 1, 2022 from https://gol.gg/esports/home/

[8] Scott Donaldson. 2015. Mechanics and Metagame. *Games and Culture* 12, 5 (June 2015), 426-444. DOI:https://doi.org/10.1177/1555412015590063

[9] TheChessWebsite. 2022. Chess Openings (2022). Retrieved August 2, 2022 from https://www.thechesswebsite.com/chess-openings/

[10] Markus Viljanen, Antti Airola, Tapio Pahikkala, and Jukka Heikkonen. 2016. Modelling user retention in Mobile Games. *2016 IEEE Conference on Computational Intelligence and Games (CIG)* (2016). DOI:https://dx.doi.org/10.1109/cig.2016.7860393

[11] Adam S. Kahn et al. 2015. The trojan player typology: A cross-genre, cross-cultural, behaviorally validated scale of video game play motivations. *Computers in Human Behavior* 49 (2015), 354–361. DOI:http://dx.doi.org/10.1016/j.chb.2015.03.018

[12] Fernando Felix Nascimento Junior, Allan Sales Melo, Igor Barbosa da Costa, and Leandro Balby Marinho. 2017. Profiling successful team behaviors in League of Legends. *Proceedings of the 23rd Brazillian Symposium on Multimedia and the Web* (2017). DOI:http://dx.doi.org/10.1145/3126858.3126886

[13] R. Ani, Vishnu Harikumar, Arjun K. Devan, and O.S. Deepa. 2019. Victory prediction in league of legends using feature selection and Ensemble Methods. *2019 International Conference on Intelligent Computing and Control Systems (ICCS)* (2019). DOI:http://dx.doi.org/10.1109/iccs45141.2019.9065758

[14] Tiffany D. Do, Seong Ioi Wang, Dylan S. Yu, Matthew G. McMillian, and Ryan P. McMahan. 2021. Using Machine Learning to Predict Game Outcomes Based on Player-Champion Experience in League of Legends. *The 16th International Conference on the Foundations of Digital Games (FDG) 2021* 1, 1 (2021), 1-5. DOI:https://doi.org/10.1145/3472538.3472579

[15] William Kavanagh and Alice Miller. 2020. Gameplay Analysis of Multiplayer Games with Verified Action-Costs. *The Computer Games Journal* 10, 1 (December 2020), 89-110. DOI:https://doi.org/10.1007/s40869-020-00121-5