Group Name :

RUNIF

Group Member :

Brian J Park (bjpark2), Min Gu Kim (mkim136), Youngseok Hahm (yhahm3)

Project Title :

Application: Predicting the Results of a League of Legends Game, and Visual Statistics

Phase of the Project :

Proposal

1. Introduction

I. What problem or topic are you addressing?

The most fundamental and ultimate interest in most competitive games is to win the game against opponents. The project focuses on the game "League of Legends" which is classified under the Multiplayer Online Battle Arena (MOBA) genre.

The game starts with 10 players split into 2 teams of 5 players each, and each player within a team chooses a "champion" from a finite pool without overlapping with other players' champions. Then, the game is played on one "map", where each player can interact with others through their chosen "champion" and "lane". A "lane" is the term used to describe a set portion of the playable map that each player defends and attacks with dedicated responsibility.

The goal of a game of League of Legends is to advance into enemy-controlled territory, destroying enemy buildings, and ultimately destroying the enemy "nexus", which are on the top right and bottom left corners on the map shown in **Figure 1** in the addendum.

The objective of this project is to construct an application that predicts the result of a game when the combination of champions are given, in addition to providing visual representations of important statistics. As there are many servers dedicated to the different regions around the world, and the difficulties associated with working with datasets from multiple regions, our project will be confined to League of Legends games played in the North America, or NA, servers.

II. Why is it interesting or important? In particular, what evidence supports this conclusion? (Cite papers or reputable sources that back up this claim.)

League of Legends has been available for the public to play for 8 years, since 2009. The game has been developed by the company Riot. Riot works diligently to keep the game relevant, by frequently fixing any bugs reported and changing champions' performances to balance the state of the game. Due partly to Riot's efforts, League of Legends remains one of the most popular electronic games to this day despite its age, boasting a daily player base of over 12 million (Paul 2017).

Moreover, the Esports industry has seen rapid international expansion, and many different countries are broadcasting these games (Warman 2017). Since this game has become popular internationally, there is much existing historical game data that can be used in statistical analyses. The statistical result obtained from this project is anticipated to be both relevant and interesting for a player of this game.

III. Where did the problem or topic come from?

As with any game, the end goal of any player of League of Legends is to win. Thus, players are interested in general statistics of champions picked in a game as well as individual players' performances with each champion. Some websites provide raw data obtained from API's and some statistically computed data pertaining to each user and champion. However, such websites fall short on providing statistical inferences based on context. Thus, our group chose to interpret this data to deliver statistical inferences in an actual game, namely, the predicted chance each team has at winning.

IV. What is your idea for addressing the problem or topic?

This project will not analyze the events that occur once the game starts; it will only consider the combination of champions chosen, each champion's performance overall, and each player's performance with the respective champion, to predict end-game results.

The application will take in a single in-game username. With the given username as a parameter, the application will request data regarding a currently on-going game from Riot's API. If the current game is at the stage where all 10 players' champion picks are locked, the application will find out each player's recent win rates with their chosen champion from Riot's API and combine this data with overall champion statistics, which will be obtained from a separate API. Then, using the methodologies discussed in **Section 3**, we will provide a statistically derived prediction of the game's outcomes.

V. How does your idea match with the course's focus on statistical programming?

So far in this course, we have learned simple statistical functions and general concepts about R. In the remaining parts of this course, we anticipate to learn more about how to "pipe" the data into R, and how to "package" our code. This project will encompass all of the topics of this course, by piping dynamic data through API's, as opposed to working with static local data files. Moreover, the project will also implement "Shiny" for R, so that our work can be packaged into a standalone, visually appealing, and easy to use application.

2. Related Work

I. Overview of pre-existing solutions.

"www.manaless.com" is a blog with various statistical analyses pertaining to League of Legends are posted. There are 38 analyses of different types of League of Legend games and among those analyses, we will use a post named "Predicting ARAM Outcome Based on the Champions Selected." ARAM, abbreviated for "All Random All Middle", is a game type within League of Legends that escapes the scopes of this project. However, the objective of the said analysis, which is to predict the outcome of an ARAM game, share some obvious resemblance to our project. The said analysis predicts the outcome of a game with an accuracy of approximately 66% (Sufficiency 2016).

In addition, a project on GitHub concerning League of Legends predicts winning rates based on in-game events (Vincentqb 2016) that occur after the game starts. Although the events that occur after the beginning of a game once again escapes the scopes of this project, the author's methods used in obtaining data from API's and the methodologies used to statistically evaluate the in-game events are relevant to our project.

II. What other ideas have been attempted

In the prior version of this project, our goal was to find out which champion a player should choose based on the comprehensive win rates of champions and player data.

Another scrapped idea for this project was to factor in correlations of champions. The motivation behind this approach was that certain champions synergize well with others, whereas, in some cases, certain champions perform exceptionally poorly when opposed by champions on the other team. However, this idea was scrapped due to the amount of effort it would consume, and as well as the fact that a case-based reasoning model, explained in **Section 3**, would make the need for the calculation of these correlations obsolete.

III. Why is your team's idea original compared to prior work?

Compared to other ideas attempted to predict the result of a game, this project takes individuals' performances into account to predict the result. Other projects tend to analyze the history of many games and with combined data. However, this project will categorize the data in detail so that each data can

provide more precise results. Moreover, this project deals with dynamic data because user's performances always fluctuate at some degree so that this project can be used whenever the user wants.

.

3. Method

This primary objective of this project is to deliver a user-friendly and standalone program that takes in a user's in-game nickname as its parameter and returns: a prediction of a current game's outcome and graphical representations of important statistics.

Although the goal is the deliver a standalone program, the program itself will be built in R. The predictive model will rely on two datasets. The first set of data is that of the total player pool, which can be considered as the "population" data in the statistical sense. The second is the dataset pertaining to individual players. We believe that these two datasets will be sufficient to provide a relatively accurate representation of the outcome of a game, which can be considered as a random variable. Then, the outcome of a specific game can be considered an individual observation from the random variable.

To this end, the main efforts in the development of this project can be largely divided into 3 parts, as follows:

I. Data Piping

As mentioned above, we will need two different sets of data - that of the player pool, which includes statistics such as the win rate of each champion over the entire player pool, and that of individual players, which includes statistics such as an individual's win rate with each champion. In the early experimentation stages, we have been able to "pipe" these datasets using API's provided by "Champion.gg" and "Riot.inc", and relevant R packages such as "jsonlite". We have also experimented with web scraping, using R packages such as "rvest" and "XML", but found that our efforts are much more efficiently spent with API's. At this stage, we believe that API keys each group member has acquired for both the Riot API and the Champion.gg API will be sufficient to support the frequencies of our API requests. As such, our current plans are to embed one of our API keys into the program.

II. Data Cleaning

Due in large to the nature of R's API handling, data fetched from the two API's will need to be "cleaned" rigorously. In our current experimental stages, the fetched data was in a very large "list" data structure. In order to render this data usable in the later sections of this project, we will restructure this data into a data frame, so that each variable (win rate, for example) of each observation (of champion "A") can be accessed within R. This step will also allow for our "case base" to be constructed readily, explained in the following part.

III. Predictive Model

The main methodology used within our predictive model will be a case-based reasoning model, henceforth referred to as CBR, that, when given a set of champions in a match, compares the champion set with those in a built-in database of finished matches which we know the outcome of (case base), selects matches that most resemble the champion set, and estimates the probability of each team winning. The emphasis of our project is less on building a statistically rigorous predictive model than on the data piping, data cleaning, and interfacing parts. With that in mind, this part of the project is one that we can always expand on. Time allowed, we may choose to implement methodologies to further increase the accuracy of the predictions made by our model. Some of such methodologies are the genetic algorithm methodology (GA), the analytic hierarchy process (AHP), and the neural network methodology.

IV. Graphics and Interfacing

With the above parts completed, the final part of our project is to wrap the R codes into a standalone web app using R's "shiny" package. We will strive to make the app as easy to use and as graphically pleasing as possible. The user would only need to input a League of Legends username. Given this username, the app will give a prediction of the game's outcome if the username given is associated with an on-going game. The app will also provide various graphical representations of relevant statistics, such as a radar chart of each player's history with the player's chosen champion.

4. Feasibility

I. Is this project able to be completed before the end of the semester?

The time required for this project will largely depend on the depth of our analysis. If we decide to do only the basic analysis and at most a simple prediction, it will surely be completed before the end of the semester. However, if we decide to add more objectives on our project to make the better prediction, the project is highly unlikely to be completed before the end of the semester.

II. What steps must occur to complete the project before the end of the semester?

The first step that must occur is obtaining datasets. There will be definitely two datasets we will require: winning rate of each champions and winning rate of randomly selected players from each different levels of gaming skill based on "tiers" which is the divider. Those two datasets will be used for making algorithm to predict the probability of winning for each team and the two datasets will be acquired from APIs of two different websites: https://developer.riotgames.com/ and champions.gg. The API of website https://developer.riotgames.com/ will be used for getting winning rate statistics of each champions based on whole population of the game ,and the API of website champions.gg will be used for comparisons of winning rates between different tiers. While getting the API from the developers of riot games, we will convert ID's of actual players to numeric summoner ID for API using the summoner API on the website and then use the numeric summoner ID for other statistical API. Next step would be making the algorithm to estimate the probability when the user inputs 10 random champions and check if the algorithm works with 100 actual games played. While checking, we will acquire what the algorithm needs to be changed and then will apply as well. When the algorithm produces highly accurate predictions, it will be used for prediction on 100 new actual games played and check if the algorithm is accurate. At the end, we will resemble our algorithm into the R package. If we can manage the project well and be able to make the project go into details, we will need more statistical analysis or datasets such as ban rates of each champions and analytics of objects from games played. These will certainly increase the accuracy of our project but it will definitely take more than a semester.

III. What is the work plan to accomplish the necessary tasks before the end of the semester?

Our original work plan to accomplish the necessary tasks before the end of the semester was obtaining dataset from the API and then do the web-scraping based on the data. However, we have concluded that web-scraping would be too difficult since there was no suitable API of this game to be used for web-scraping and thus, became to acknowledge ourselves that focusing on API instead of web-scraping would be more efficient and will presumably bring a simple outcome. We plan to accomplish the stages of collecting data during the proposal phase. After collecting dataset from API,then acquiring algorithm and R package based on cleaned data and applying the produced algorithm to actual

datas will be accomplished during the progress update phase. Before the project presentation, we will use "shiny package" from R so that the data and algorithm is visualized and be accessible.

5. Conclusion

To the extent of this project as a learning tool, we believe that we will to acquire a comprehensive understanding of R's capabilities in the undertaking of this project. We will not only familiarize ourselves with R's built in capabilities, such as those related to statistical modelling, but also data piping and interfacing of end product. We anticipate that, through this project, our skillset with R will be extended from a purely academic one usually acquired in a classroom environment to one that better resembles the skillset required by the industry.

Regarding this project's intrinsic values, the utility the end product is planned to deliver will likely have nontrivial relevance to the League of Legends community. Although much work has already been completed and is currently underway that resembles our own, our work is unique in the way that it takes the results of some of the said previous work, and delivers visualizations and predictions using these results. With success, this project may expand to a mobile application.

6. References

"Champion.gg - LoL Champion Stats, Builds, Runes, Masteries, Counters and Matchups!" 2017. Champion.gg - LoL Champion Stats, Builds, Runes, Masteries, Counters and Matchups! Accessed July 7. http://champion.gg/.

Jeroen Ooms (2014). The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R objects. arXiv:1403.2805[stat.CO] URL https://arxiv.org/abs/1403.2805.

Pau, Jeremiah. 2017. "By The Numbers: Most Popular Online Games Right Now". Nowloading. https://nowloading.co/posts/3916216,

Smith, Dave. 2015. "How the 'Michael Jordan of eSports' annihilated his opponents to win the 'League of Legends' World Championships - in GIFs." Business Insider. Business Insider. November 12. http://www.businessinsider.com/how-faker-skt-won-league-of-legends-world-championships-2015-2015-11.

Sufficiency. 2016. "Data Analytics for League of Legends." Data Analytics for League of Legends. http://www.manaless.com/.

Vincentqb. "Vincentqb/league-of-legends." *GitHub*. N.p., 18 June 2016. Web. 07 July 2017. https://github.com/vincentqb/league-of-legends/

Warman, Peter. "Esports revenues will reach $696M in 2017 and $1.5Bn by 2020." *Newzoo*. N.p., n.d. Web. 07 July 2017. https://newzoo.com/insights/articles/esports-revenues-will-reach-696-million-in-2017/

Winston Chang, Joe Cheng, JJ Allaire, Yihui Xie and Jonathan McPherson (2017).shiny: Web Application Framework for R. R package version 1.0.3. https://CRAN.R-project.org/package=shiny

Zhang, Qianyun. "R code for Riot League of Legend API Challenge." *Qianyun (Poppy) Zhang*. N.p., n.d. Web. 07 July 2017. http://www.poppy-zhang.com/r-coding/using-riot-league-of-legend-api-to-pull-out-game-history-data/

7. Addendum

I. Contributions

The project's objective, the methods implemented to achieve this objective, and early experimentation has been a product of equal contributions by each group member. The creation of this proposal document also saw much collaborative effort, but we chose to divide primary responsibility for each section among the members, as follows:

Brian J Park: Introduction (Basics of League of Legends), Methods, Conclusion, Reference
Min Gu Kim: Related Work, Feasibility, Reference
Youngseok Hahm: Introduction, Related Work, Reference

II. Figures

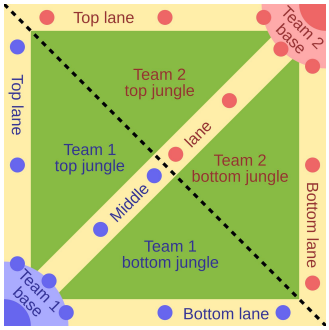

Figure 1: A visual representation of the map in LoL.

III. Git Repository

The following is a link to a the private GitHub repository on which all work relating to this project will be uploaded: https://github.com/bpark131/runif