

Project Title: League of Legends Match Results Predictor and Team Recommender

Project Members: Hubert Ye and Caitlin-Dawn Sangcap

Team Member's Role and Contributions:

We will both help compile the dataset and gather the information of what will be used, because the API provided by Riot Games has a limit of requests when used by free accounts.

We will each write our own algorithm that we will use to analyze the dataset, because we want to see which algorithm is the better predictor. One for Logistic Regression, and the other will program k-Nearest-Neighbors.

We will also each write our own own algorithm for the predictor. One for Genetic Algorithm and another for Collaborative Filtering.

CSCI 353 Related Topics

We will both contribute to doing dimensionality reduction and bias/variance trade offs.

We are both also contributing to using Logistic Regression, k-Nearest-Neighbors, Genetic Algorithm and Collaborative Filtering.

Project Description:

For the final project, we propose to create a game winner predictor. The predictor will be used on the online multiplayer game, League of Legends. The game League of Legends has a variety of characters to choose from when playing the game called Champions. The basis of prediction will be based on these Champions of the teams playing and the player's mastery level for that champion.

It is our goal to see if character selection has a bigger effect on the outcome of the game than is normally seen. There are a plethora of factors that could be counted

into seeing who could win a game but before the game even starts, the Champion Selection and ban list come first. The purpose of our project is to see how much of the champion selection comes into play in deciding which team will be the victor. In addition to predicting the winner, we are attempting to improve the champion composition of the losing team in an attempt to improve the losing team's chances of winning.

Dataset:

We will gather the usernames of professional League of Legends esports players that are registered for tournaments this year. Then we will use the API provided by Riot Games, in a different program that we will write to get their player information and what happened in the games they participated in, and export it into a .csv file. This .csv file will contain the information of the Champions used on both teams, which team won or lost, and other information that the API may give us about the game.

By using the previous matches played by the professional players as the training data, we want to predict the outcome of more recent matchups. We will only use data from specific dates because League of Legends often gets patch updates that upgrade and downgrade specific champions to balance out the game design, and we want to avoid such variables.

Project Timeline

November 3rd: Finish compiling the list of professional LoL esports players we are basing the data on and also finish writing the code to get the dataset of all the games played by the list of players above using the Riot Games API

November 10th: Finish compiling and cleaning the dataset to remove unnecessary input variables for each model and begin implementation.

November 17th: Finish implementing Logistic Regression and kNN

December 8th: Finish implementing Genetic Algorithm and Collaborative Filtering

December 15th: Final Presentation

Demo of the final exam date:

We plan to demo the algorithms and display which one both thinks which team will win based on the characters the team has chosen against their opponents along with the actual winner. We will have a training and testing data set already preloaded and we will stream that as our presentation.

How we plan to evaluate:

We will compare our Logistic Regression to the result in Lin's "Legends Outcome Prediction", to see if we can get closer to Conley's and Perry's results in "How Does He Saw Me ? A Recommendation Engine for Picking Heroes in Dota 2." To see if we can improve his 55.2% accuracy to Conley's and Perry's 67%.

We also plan to use the program from "An Approach for Team Composition in League of Legends using Genetic Algorithm" to see the chances of winning with the different team compositions based on the three preprogramed strategies (hard engage, team fight and poke), and then compare it to another algorithm like Collaborative Filtering..

References:

Conley, K., & Perry, D. (2013). "How Does He Saw Me ? A Recommendation Engine for Picking Heroes in Dota 2."

[PerryConley-HowDoesHeSawMeARecommendationEngineForPickingHeroesInDota2.pdf](#)

We plan to use this as the baseline for the project because in this paper, they had created a predictor and recommender for another game of the same genre, Defense of the Ancients 2, DotA 2. They used Logistic Regression and k-Nearest-Neighbors to predict which team would win. Then using those models, they used a greedy algorithm to recommend a better team composition but never specified which algorithm that was.

L. Costa, A. Souza and F. Souza, "An Approach for Team Composition in League of Legends using Genetic Algorithm," in *2019 18th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*, Rio de Janeiro, Brazil, 2019 pp. 52-61.

Github repo: <https://github.com/tekpixo/moba.aid>

This paper expanded on the paper above but rather than predicting the winner of the game, this paper took the approach of improving the team composition based on champion attributes and team strategy. This approach gave more emphasis to a champions' role, armor, health point, attack damage and attributes of that nature to satisfy the conditions of a specific strategy. This approach would continuously improve the team selection over the specified number of generations, populations and mutation rate to provide the best team composition.

Lin, L. (2016) "League of Legends Outcome Prediction"

<http://cs229.stanford.edu/proj2016/report/Lin-LeagueOfLegendsMatchOutcomePrediction-report.pdf>

This is a more modern paper, and uses League of Legends to predict the winning team using Gradient Boosted Trees and Logistic Regression together. However, unlike the first paper we are referencing, it does not reference any other algorithms that it is comparing to other than itself, so we will compare it to k-Nearest-Neighbors for our predictor, to see if it is still a better predictor compared to in the past for DotA 2.