League of Legends Strategy-based Team Clustering and Team Performance Prediction

Jason Jin
Department of Mathematics
Emory University
jason.jin@emory.edu

Leisheng Yu
Department of Computer Science
Emory University
leisheng.yu@emory.edu

Siwei Xu
Department of Mathematics
Emory University
siwei.xu@emory.edu

***Abstract*—Esports is a rapidly growing industry. Esports teams play against each other in tournaments and the statistics about different matches and teams are available on various websites. In this paper, we analyze the collected League of Legends team statistics from different websites with some effective data mining techniques and algorithms. In this way, we can make two key contributions: achieving a relatively accurate and nontrivial strategy-based clustering of teams based on selected in-game attributes as well as predicting a team's performance in one tournament or season based on its historical (Pre-season) performance statistics.**

***Index Terms*—Data Mining, Predictive Analysis, Data Visualization, Esports Statistics, League of Legends, Strategy-based Clustering**

I. Introduction

League of Legends is a team-based strategy game developed and published by Riot Games, where two teams, each consists of five players facing off to destroy opponents' base while protecting their own base [1]. Each base is protected by turrets that attack enemy armies, so each team shall aim for destroying the enemy turret to get access to the main base. Unlike traditional sports, there is no time limit for each game in League of Legends, and the game would only terminate when one of the teams successfully destroys its opponents' main base. The game involves selecting game characters, called champions, to perform the battle against opponents' champions, and competing for neutral resources from opponents in order to gather gold for better equipment, which helps in winning the battle as well as destroying turrets.

The game now has mature professional league systems and has regular match seasons annually. Professional players and teams are divided into different regions based on geographical locations. Some major regions are "LCK" from Korea, "LPL" from mainland China, "LEC" from Europe, and "LCS" from North America. Every year each region holds its individual tournament and decides its own championship, and the top teams from each region will be invited to participate in the world championship called "S series" held in every Autumn.

The prosperity of the professional leagues of League of Legends validates the current success as well as attractiveness of this Esports market and explains the large amount of data of different games, tournaments, and teams. The sufficiency of the available data related to League of Legends and the expanding industry have motivated the ideas and focus of this project. The main contributions of this project can be summarized as follows: (1) we distinguish the important attributes that may influence the game win rate, and based on these findings, we predict a team's win rate in a tournament or season based on its historical data (Pre-season game performance); (2) we utilize the useful attributes of teams to put teams into different clusters, and the result of clustering can help to find out whether two teams have similar play-styles (the introduction to the attributes in the employed data set and the reason why using the selected attributes can be effective and convincing in achieving nontrivial strategy-based clustering of teams are presented in later sections). Identifying the factors that significantly affect the game win rate and being able to predict the result of the Esports games can be meaningful because of sports gambling. Effective and accurate predictive analytics should be important for every sports because it can be incorporated in calculating the odds in sports gambling. Moreover, achieving meaningful strategy-based clustering of teams is beneficial in many ways. First, equipped with the knowledge of each team's play-style and the similarity between two teams in terms of their gaming strategies, the commentators can make more useful and concrete comments during the broadcasting of a League of Legends game. Also, being able to realize different strategy clusters of teams and the win rate of every cluster, the developers of League of Legends can adjust the game settings appropriately to avoid the existence of "a single key to victory," which means that the teams in a specific strategy cluster have an unreasonably higher average win rate, making sure that the game is interesting and diverse in terms of gaming strategies.

II. Related Work

Esports is still an emerging field compared to other traditional team sports like soccer, basketball, and volleyball, so the research on League of Legends statistics analysis, which is a specific sub-field of Esports, is not abundant. However, research with similar purposes on some traditional team sports is prosperous and there is study reasoning that traditional team sports and Esports are similar in terms of the competitive settings in which tactics and skills are valued [3]. Moreover, In [4], the authors showed that variations of two popular baseball predictive models can be applied to player performance prediction in Esports. Thus, we planned to explore existing works related to the application of data mining and machine learning on the analysis and outcome prediction in some popular traditional team sports. If data mining and machine learning techniques can show their merit in the field of traditional team sports, the applications of data mining algorithms on Esports will be worthy of exploration.

In [5], it was shown that machine learning algorithms like Simple Logistics Classifier, Artificial Neural Networks, SVM, and Naïve Bayes can be effective in outcome prediction for basketball games. Furthermore, because of the attractiveness of game result prediction for team sports, there are several comprehensive reviews of the applications of data mining and machine learning techniques for outcome prediction for traditional team sports [6], [7]. These works demonstrate the effectiveness and fruitfulness of employing data mining algorithms in analyzing traditional sports statistics. Hence, we were further motivated to explore the applications of data mining techniques for Esports statistics analysis. However, there are indeed some significant differences between League of Legends competitions and traditional team sports games. For example, in traditional sports, the game outcome is mainly decided by performance of the players on the pitch; in contrast, in League of Legends, variables such as champion selection should also be considered.

Although existing works related to League of Legends data mining are extremely limited, we still succeeded in finding several previous studies that have made nontrivial contributions to League of Legends data mining and are related to our focus. In [8], the author elucidated how to get the data for Esports statistics analysis and made an overview on how to apply basic data mining and data visualization techniques to analysis relevant to League of Legends, which serves as a comprehensive tutorial or introduction. Also, Analytical Hierarchy Process (AHP) has been shown to generate a satisfactory result when being applied in game outcome prediction in Esports [9]. Moreover, in [10], the authors used unsupervised learning to achieve clustering of player behaviors. In [11], the author demonstrated the effectiveness of employing logistic regression and decision tree in predicative analysis on Esports games based on in-game attributes as well as the champion selection. Although our project has been motivated by these existing works, none of them could solve the problems that this project is focusing on. Some of them could do clustering of players, but existing works that could do strategy-based clustering of teams were not found; many of them contributed to the prediction of the outcome of a single game, but studies that can predict a team's win rate in a coming season based on the historical (Pre-season) game performances could not be discovered.

III. Key Questions

Our goals include answering the questions listed below. These questions are the bedrock of our focus. Through exploring these questions with different kinds of approaches, we can make better analysis and predictions on the performance of a League of Legends team with a specific value for each attribute.

A. Are there any significant correlations between attributes?

There may exist a certain correlation between a pair of attributes. For example, there may be a positive correlation between first dragon participation and dragon rate. We plan to explore the existence of nontrivial correlations for every pair of attributes through applying data mining algorithms and data visualization.

B. How can a relatively accurate and effective strategy-based clustering of teams be achieved?

There are various attributes that can be analyzed to assess and evaluate a team's overall performance in a season such as "first blood rate," meaning the likelihood of taking down one of the opponents' champions earlier than the opponents do since the start of the game; "CSPM" indicates the average minions taking down per minute, which is related to the income per minute as taking down minions gives players money. These attributes are easy to find and can deliver additional information about the team's gaming styles and strategies. For example, the most explicit attribute "game duration" sums the game lengths of a team's all matches. By calculating the average game duration for each match, we will be able to divide teams into finishing the game in a very short time and finishing the game in a relatively long time duration. Ideally, the teams with an aggressive game style that "actively organize attacks in the early game time" tend to have a low number on the game duration attribute because they finish the game with their early attacks. In contrast those teams that prefer a more low-risk game style tend to finish games late in order to gather enough map resources before launching major attacks.

There are a number of other attributes that can be used to tell different team strategies in different directions, such as the "first dragon rate," which can in some extent distinguish teams that prefer to focus on the bottom part of the map from others, since the dragon as a map resource is fixed to spawn in the bottom part of the game map. Correspondingly, another map resource called "rift herald" is fixed to spawn in the upper part of the game map, and thus, those teams with a high "first rift herald control rate" can be determined to have a strategy focusing on the upper part of the map. Via mining the team statistics, it is possible to find similarities and differences among the teams with respect to specific attributes, thereby it becomes possible to cluster or classify them into groups, which is helpful in understanding the meta of League of Legends professional games as well as making predictions on counter relationships, such as "teams with aggressive styles have higher win rate when playing against teams that prioritize dragons." Thus, we are motivated to achieve a relatively accurate and nontrivial strategy-based clustering of teams.

C. How to predict a team's win rate in the summer tournament based on its game performance in the spring?

The teams that have a higher win rate may have some common characteristics. The keys to success probably have a general pattern. Hence, we plan to predict a team's win rate in the summer tournament based on the data of its' spring matches (which happened before) that include values for different attributes by applying some data mining techniques. After achieving this goal, it will also be applicable to categorize the teams into different tiers based on their predicted performances (winning percentage).

IV. Proposed Approaches

We collected data of each team's tournament performances between 2017 and 2021, including their winning rate, number of games played, average game duration, kills per game, and etc. Most of the attributes are of numerical type, and there are very few missing values.

To address the first question, we calculate the correlation between features with the following formula:

$$corr(X,Y) = \frac{\sum_{k=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \sigma_y}$$

where $\bar{x}$, and $\bar{y}$ is the mean of the attribute X and Y respectively, while σ_x and σ_y being the standard deviations. If we rank the correlation score between the winning rate and every other attribute, we get a list of factors that contributes to the win and loss of a team in a descending order of importance. To get a more intuitive idea of relationships between features, we can also do visualization with bi-variate scatter plot where each coordinate represents a pair of attributes. This step also prepares us for the next few questions by detecting data redundancies, and eliminating features that are highly correlated.

To address the second problem, we have removed redundant features beforehand, but we still need to manually delete attributes that are not collected during the game, such as number of games played, and the winning rate. We standardize all points with z score, and then compare the result of directly performing K-means clustering on the dataset, and performing PCA or t-SNE to reduce the dimensionality before using the clustering algorithm. This is because we have a data set with relatively small amount of samples, but large amounts of attributes. To decide the number of clusters (K), we apply the silhouette evaluation, which is a measure of how similar a data object is to its own cluster compared to other clusters. When the silhouette coefficient value of a data object approaches 1, the cluster containing this data object is compact and this data object is far away from other clusters, which is an ideal case. Therefore, we want the average silhouette value to be as close to 1 as possible. From Fig. 1, we can see that $K = 2$ is the preferable case, because of its highest silhouette value. Finally, we select a set of attributes that can possibly represent the strategies of a team by logic, and then do scatter plot of pairs of these attributes while coloring points by labels derived from K-means. We observe these plots and conclude a different strategy for each cluster. Furthermore, we can test whether there is a difference between the average winning percentages of two clusters to show if some combinations of strategies are better.

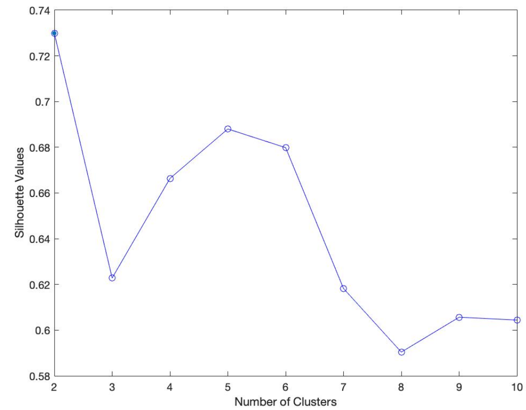

Fig. 1. Silhouette Values versus Number of Clusters

To predict the performance of a team in the summer tournament, we fix our target variable to be whether the winning rate of the team exceeds 0.5, and X variables to be attributes that describe a team's performance in the spring season. Then we build a classifier on the training data set which is all the teams from Season 7 to 9, and evaluate the model with our testing data set, Season 10. In the end, we apply our model to Season 11, and get predictions of winning rate of teams in the summer tournament, which will happen soon. We can then categorize teams into different tiers based on the predictions of their win rates.

V. Experiment results

Before showing the result to the first question, a brief description is given on pre-processing the data set. Select all team's performance statistics in 2021 and make it our input. There are 274 teams and 29 different features that are showcased and explained in Table 1. First, we check for the existence of any missing values, and there are four teams that do not have either the first tower rate or the first blood rate, so we delete these four rows from the data frame. Next, we convert all features except for team name, region, and season to float numbers such as the win rate, which was stored as a string before, and the game duration (33:55:00) with the following formula:

$$minute + second/60$$

Lastly, we manually remove non-numerical features, including name, region, and features that can be easily derived from each other, such as "DRA" and "VWPM." After the entire process, we use 270 teams and 24 features to study the correlation between features and their respective contributions to the win or loss of a game.

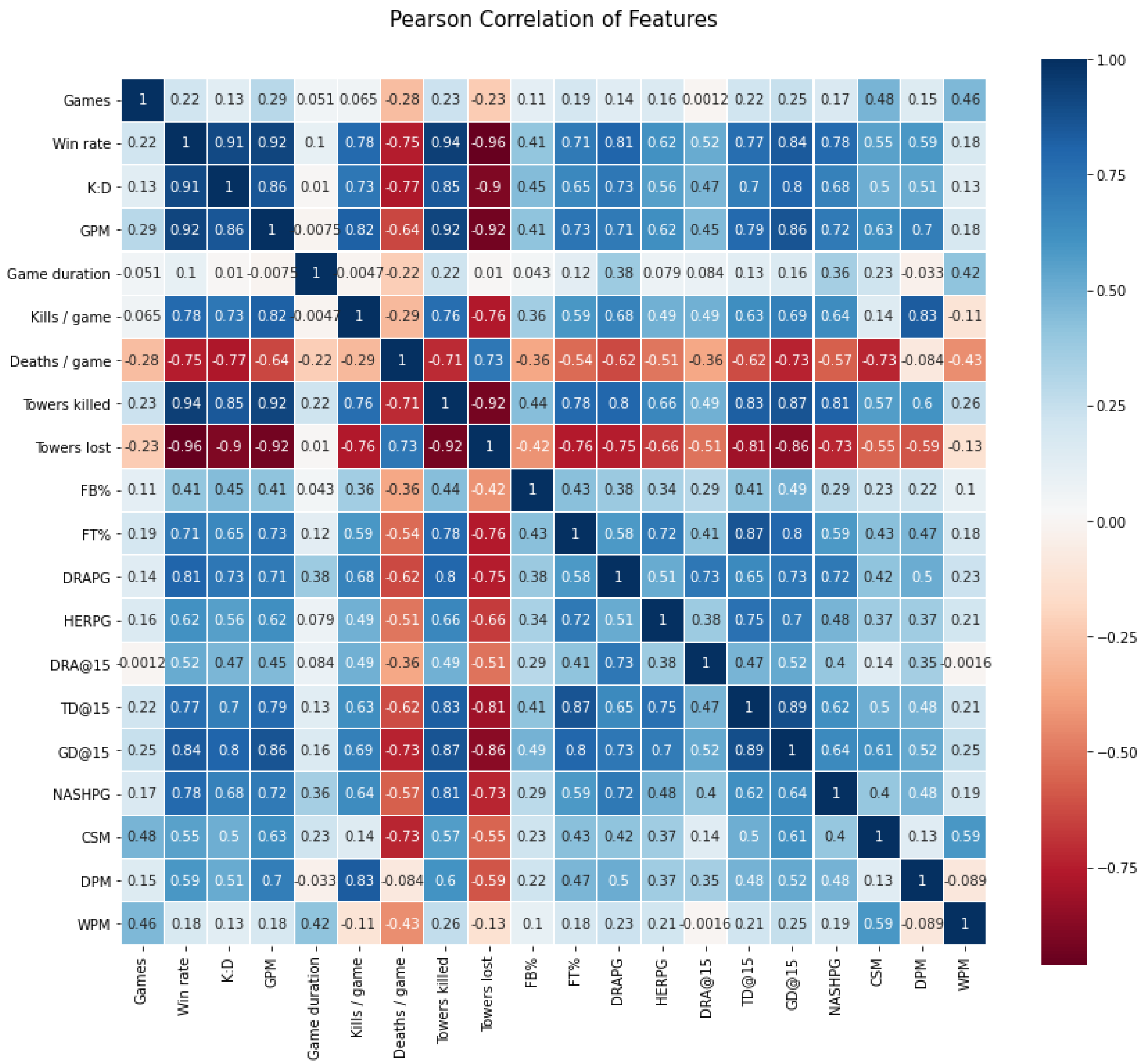

Fig. 2. Correlation Table between Features

A. Key Question 1

In Fig. 2, we present the correlation score between every pair of features. And we can see that there is a relatively strong relationship either positive or negative between most of the features. This is within our expectation because if a team wins a game, it is reasonable for this team to have a high kill versus death rate, tower kill rate, a large GPM (gold per minute), and more dragons killed. Among these features, game duration, FB (first blood rate), and WPM (wards per minute) are less correlated with others. Additionally, we see a few strong correlations that were rather unexpected as they are usually not directly related to each other.

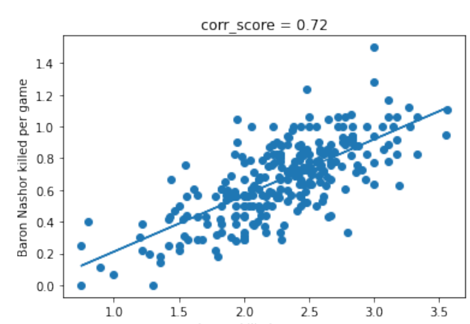

Fig. 3. Dragons killed v.s. Nashor barons killed

TABLE I
DESCRIPTION OF FEATURES

Category	Features	Description
Before Game Info	Name	Team Name
	Season	year of the game
	Region	region of the team
	Games	number of games played
Game Result	Win rate	games won/games played
In-game features	Kills/game	kills per game
	Deaths/game	deaths per game
	GPM	gold per minute
	GDM	gold differential per minute
	Game duration	game duration
	K:D	$\frac{Kills/game}{Deaths/game}$
	Towers killed	Towers defeated
	Towers lost	Towers lost
	FB	first blood rate
	FT	first tower rate
	DRAPG	dragons killed per game
	HEPRG	Herald killed per game
	DRA@15	Average dragon at 15min
	TD@15	Tower differential at 15min
	GD@15	Gold differential at 15min
	NASHPG	Baron Nashor killed per game
	CSM	creeps per minute
	DPM	damages to champions per minute
	WPM	minions per minute

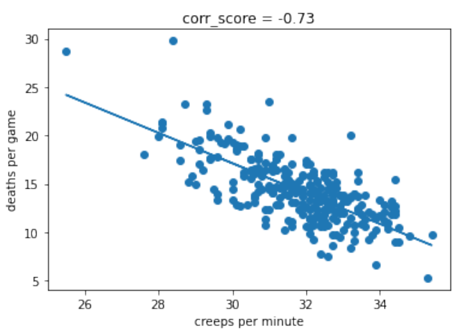

Fig. 4. Deaths per game v.s. creeps per minute

From the correlation between dragons killed per game against Baron Nashor in Fig. 3, which is the most powerful neutral monster in the game and provides strong enforcement to the team who kills it, we can see a fairly strong positive relationship. However, the dragons, as mentioned before, are fixed to spawn at the bottom part of the map while the Baron Nashor spawns at the upper part. Therefore, it may seem unintuitive in the first glance. In fact, due to the fact that dragons also provide valuable resources and strengths to the teams, when a certain team has already taken down a number of dragons, it has the choice to bait the opponent team to fight with them by grouping around the position of the Baron. The opponent team would have to make a decision of either giving up the Baron to their rivals or fighting with disadvantage (since the rival possesses more dragons and is likely to be stronger in the group fight), and vice versa, the team taken down the baron has advantage of getting the next dragon as well.

	feature	importance
5	Towers killed	0.940383
1	GPM	0.920246
0	K:D	0.907622
13	GD@15	0.844105
9	DRAPG	0.805098
3	Kills / game	0.781229
14	NASHPG	0.778464
12	TD@15	0.766181
8	FT%	0.712494
10	HERPG	0.616186
16	DPM	0.590622
15	CSM	0.553520
11	DRA@15	0.516517
7	FB%	0.408548
17	WPM	0.175887
2	Game duration	0.103085
4	Deaths / game	-0.748988
6	Towers lost	-0.960399

Fig. 5. Feature importance

The correlation between creeps (minions) per minute against deaths per game in Fig. 4 is also indirect but easy to infer. In a specific match, when a team dies a lot, it is likely that they are in disadvantage against the opponent. When that is the case, the team will have to be extra cautious because the opponent team can easily kill the players who are trying to earn income by killing more minions (minions are in the lanes which are shared and exposed to both teams). As a result, the disadvantage team will not be able to farm as much as they want but will have to give up some minions to ensure safety, which can result in such a negative relationship.

We consider the win rate as our target variable, so now we use the correlation score between win rate and each feature to

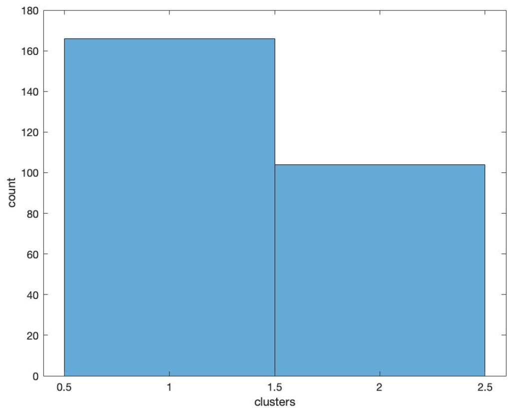

Fig. 6. Option One

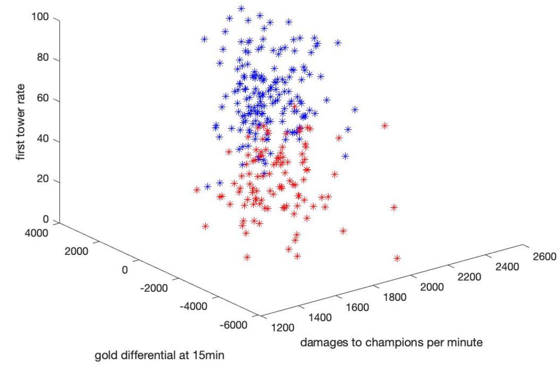

Fig. 7. Clustering visualization in a 3-dimensional space

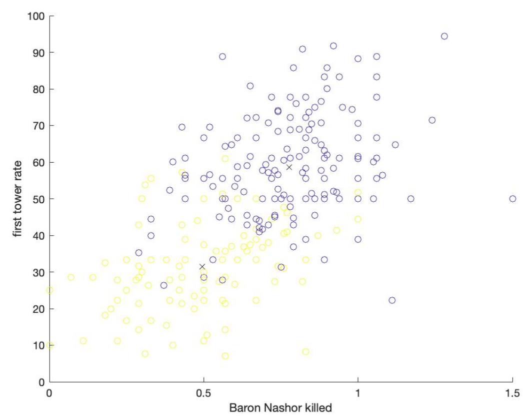

Fig. 8. Clustering visualization in a 2-dimensional space

represent the importance level of each attribute. From Fig. 5, we get the following results:

1) Tower killed, K:D, and GPM contribute the most to the winning of a game, which is under expectation.

2) Game duration does not have much an impact on the win or loss of a game.

3) Killing dragons seems to help with winning the game more than killing heralds does.

4) Taking the first tower is much more relevant to the win or loss than taking the first blood.

B. *Key Question 2*

We performed K-means and clustered all teams in the data set into two clusters. Since we utilized about twenty attributes in clustering the teams and we could not visualize the clustering result in a twenty-dimensional space, we provided two options for simplified visualization.

Option one is using the histogram, shown in Fig. 6. The only information we could get from this histogram is that there are more teams in cluster 1.

Option two is to visualize the clustering result in a three-dimensional, or two-dimensional, space. A combination of three representative in-game features is chosen as an example, shown in Fig. 7, and a combination of two representative in-game features is chosen as a second example, shown in Fig. 8. From Fig. 7, we can see that two clusters are separated by different colors, and the blue teams get higher scores on all three attributes: first tower rate, gold differential at 15min, and damages to champions per minute. This makes sense because, if a team has more gold in a game, they can purchase better equipment for their champions, which leads to greater damages to opponents' champions and towers. In Fig. 8, the yellow teams have lower scores on both first tower rate and Baron Nashor killed. This pattern can also be explained by a similar reason: if a team has more Baron Nashor killed, they can possess more resources, which can aid the team in defeating opponents' towers.

After the visualization, we would like to compare the average win rates of the two clusters, to see which cluster is better. The result shows a large discrepancy: the average win rate of teams in cluster 1 is about 59.8849% and that of teams in cluster 2 is about 29.2067%. Since we are doing a strategy-based team clustering, we can infer that the overall strategy of teams in cluster 1 may be more effective than that of teams in cluster 2. But because of the large difference and the previous visualizations, it is reasonable to suspect that such a difference in average win rates is fundamentally caused by players but not strategies: if a team has better players, they can afford a more advanced and demanding strategy, which aims at better defense and better attack simultaneously.

C. *Key Question 3*

Now we show the answer to the third problem, which is to predict the winning percentage of a team in the summer tournament based on its performance in the spring season. We set the target variable to be 1 if the win rate of the team in summer is larger than 0.5, and 0 if it is smaller than 0.5. Then, we want to choose independent variables, which are also known as features. From the result of the first key question, we know that there are 24 features after getting rid of those redundant ones, but there exists a strong relationship between many of them. When we build a classifier, we prefer not to have attributes that are highly correlated since it increases training time, making the model more complicated. So for every pair of features such that the correlation score is larger than 0.7, or smaller than -0.7, we delete the one that is less important in Fig. 5. Eventually, We reduce the number of features to 10, and they are collected in the spring tournament: “Games,” “Win rate,” “Game duration,” “FB,” “FT,” “DRA@15,” “NASHPG,” ’CSM,” “DPM,” “WPM.”

Next, we define the training and testing data set. Generally, we do random split among all the points we have, but in this case, time order matters. For example, we cannot use data from Season 9 to train the classifier, and use it to predict winning rate for Season 8, since Season 8 happens before Season 9. Therefore, we take Season 7 to 9 as our training data set, and Season 10 as our testing data set.

Before building any classifier, we use a simple rule to predict the target variable (y): if a team’s win rate in the previous spring is above 0.5, our prediction for y is 1, otherwise, it is 0. When we compare our prediction to the real y, we get the following result:

ROC AUC : 0.7069

	precision	recall	f1-score	support
0	0.76	0.71	0.73	99
1	0.65	0.71	0.68	75
accuracy			0.71	174
macro avg	0.70	0.71	0.70	174
weighted avg	0.71	0.71	0.71	174

To evaluate the model, we get the roc auc score, precision and recall on both 0 and 1, as well as the f1-score. This is also how we will evaluate all the classifiers. We claim that a classifier is only valid if it has a better roc auc score than what we got from applying the rule–0.7069. There are a lot of options for the classifier. Here, a concrete discussion on each classifier we tried out is given below, including a brief theoretical explanation, adaptations made for implementation (packages in sklearn are used for all classifiers), and the model evaluation.

1) Decision Tree: data continuously splits according to a certain parameter, till we reach the leaf node which returns a class label. From the evaluation table, we can see that the score is really low, which means the model is not doing well.

ROC AUC : 0.5679

	precision	recall	f1-score	support
0	0.63	0.58	0.60	99
1	0.50	0.56	0.53	75
accuracy			0.57	174
macro avg	0.57	0.57	0.57	174
weighted avg	0.58	0.57	0.57	174

2) Random Forest: decision tree did not work well, so we try an ensemble classifier – random forest, which is composed of many decision trees. And it returns the final label by doing weighted average on all the tress. To prevent the algorithm from over-fitting, we add limits to the trees. Specifically, the max depth is 5, and the minimum number of samples required to split a node is 10. From the table, we see that the random forest classifier did a lot better than decision tree, but still not as good as the model generated by the simple rule.

ROC AUC : 0.6721

	precision	recall	f1-score	support
0	0.71	0.76	0.73	99
1	0.65	0.59	0.62	75
accuracy			0.68	174
macro avg	0.68	0.67	0.67	174
weighted avg	0.68	0.68	0.68	174

3) Logistic Regression: it is a statistical model that uses a logistic function $(\frac{1}{1+e^{-t}})$ to model a binary dependent variable, and in the training process, we kept changing the parameters $(a_0x_0 + a_1x_1 + ... + a_nx_n)$ to minimize the cost function. We need to standardize our data set with the following formula: $z = \frac{x-\mu}{\sigma}$. Then, we could feed it into the algorithm. From the table, we see that this model is valid, since 0.7152 is slightly bigger than our reference number. The f1-score for both 0 and 1 cases are decent.

ROC AUC : 0.7152

	precision	recall	f1-score	support
0	0.78	0.70	0.73	99
1	0.65	0.73	0.69	75
accuracy			0.71	174
macro avg	0.71	0.72	0.71	174
weighted avg	0.72	0.71	0.71	174

4) Support vector machine: this algorithm finds a plane that separates samples from two classes with maximum margin. We choose a linear kernel since in our data set, features are mostly linearly correlated with the target variable. Again, we need to standardize the data set before the training starts. From the table, we can see that SVM outperforms all the algorithms that have been described before. It is also a stable method since there is no randomness throughout the entire training process.

ROC AUC : 0.7455

	precision	recall	f1-score	support
0	0.79	0.76	0.77	99
1	0.70	0.73	0.71	75
accuracy			0.75	174
macro avg	0.74	0.75	0.74	174
weighted avg	0.75	0.75	0.75	174

After comparing different options, we finalized support vector machine as our classifier model. Now, we apply it to Season 11. If we enable the classifier to predict the probability of a sample having 1 for its target variable, we get the chance

	number of teams
(0.0, 0.1]	6
(0.1, 0.2]	18
(0.2, 0.3]	40
(0.3, 0.4]	32
(0.4, 0.5]	47
(0.5, 0.6]	45
(0.6, 0.7]	38
(0.7, 0.8]	34
(0.8, 0.9]	7
(0.9, 1.0]	1

Fig. 9. Count of teams in every tier

of a team winning more than 50 percent of the games in the summer tournament. The bigger this chance is, the better the team is, so we can categorize teams into different tiers by this value. Specifically, we cut the interval 0 to 1 into 10 bins: 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1. The teams will fall into the corresponding tiers given our prediction on their chance of winning more than 50 percent of the games. We count the teams that fall into each tier to get an idea of the distribution of teams' strengths in the upcoming tournament. From Fig. 9, we see that there are very few teams that fall into the two extremes. Most of them are within 0.3 to 0.6.

VI. Discussions

Our study explored what in-game features can contribute to the winning of a game the most, which can help the LoL teams in the future better emphasize the crucial factors during the game. Moreover, the strategy clustering we achieved in this paper can help the game experts better understand the meta of League of Legends and assist both the media and commentators better introduce the teams. The method we proposed in predicting the future winning percentage of a team based on historical statistics can definitely prove its meaningfulness in sports gambling.

Although we have found nontrivial results, our research in this field is still limited. For example, we did not consider the factor of champion selection in our data mining process because the data set we used did not contain related information; also, there still exists some blurring in team clustering: is the strategy-based team clustering we did fundamentally strength-based clustering? These could be targets of our future investigation.

VII. Conclusions

In this paper, we tackled three key questions related to data mining in League of Legends statistics. The first key question is whether there are any significant correlations between attributes of League of Legends teams. We used a correlation table to show the correlation between every pair of attributes. We specifically highlighted the strong positive relationship between dragons killed per game and Baron Nashor killed per game as well as the negative relationship between deaths per game and creeps per minute. Also, by listing the correlation between the win rate and each in-game feature, we found nontrivial patterns such as that towers defeated, K:D, and GPM contribute the most to the winning of a game. The second key question asks about achieving an effective strategy-based clustering of teams. We applied K-means and clustered the teams into two clusters. Also, we provided different ways to visualize the clustering result and a large discrepancy in the average win rates of two clusters were discovered. Last but not least, the third key question aims at predicting a team's win rate in the summer tournament based on its previous performance statistics in spring. We compared different existing popular classifier models and discovered that the support vector machine had the best performance and could make relatively accurate predictions. Then we used SVM to make predictions on teams' performances in Season 11 and categorize the teams into different tiers based on their predicted winning percentages.

References

[1] https://na.leagueoflegends.com/en-us/how-to-play/

[2] https://gol.gg/teams/list/season-S11/split-Spring/tournament-ALL/

[3] Michaluk, T. (2012). Changes in the meaning of physicality in modern sport -from disabled sport to e-sport. Physiotherapy / Fizjoterapia. 20(1), 64-70.

[4] Shim, K. J., Sharan, R., Srivastava, J. (2010). Player performance prediction in massively multiplayer online role-playing games (MMORPGs). Pacific-Asia Conference on Knowledge Discovery and Data Mining. 71-80.

[5] Cao, C. (2012). Sports data mining technology used in basketball outcome prediction. Masters Dissertation. Technological University Dublin.

[6] Haghighat, M., Rastegari, H., Nourafza, N. (2013). A review of data mining techniques for result prediction in sports.Advances in Computer Science: an International Journal,2(5), 7-12.

[7] Rory Bunker, Teo Susnjak. (2019). The application of machine learning techniques for predicting results in team sport: a review.

[8] L.B. Hugo. (2010). Data Science as support of esports performance and strategies (I).

[9] Aryanata, G., Rahadi, P., Sudarmojo, Y. (2017). Prediction of DOTA 2 Match Result by Using Analytical Hierarchy Process Method. International Journal Of Engineering And Emerging Technology, 2(1), 22-25. Retrieved from https://ojs.unud.ac.id/index.php/ijeet/article/view/34440

[10] Ong, H. Y., Deolalikar, S., Peng, M. (2015). Player behavior and optimal team composition for online multiplayer games.

[11] W. Tian. (2018). Predictive analysis on eSports games - a case study on League of Legends (LOL) eSports tournaments.