ASEAN
Journal of Scientific and Technological Reports
Online ISSN:2773-8752

Research article

A Study on Using Machine Learning to Predict Winner in Multiplayer Online Battle Arena (MOBA) Game

Nattapat Tangniyom[1] and Pruet Boonma[2*]

[1] Data Science Program, Faculty of Engineering, Chiang Mai University, Chiang Mai, 50200, Thailand
[2] Data Science Program, Faculty of Engineering, Chiang Mai University, Chiang Mai, 50200, Thailand
* Correspondence: pruet.b@cmu.ac.th

Citation:
Tangniyom, N.; Boonma, P. A Study using ensemble learning to predict winner in multiplayer online battle Arena (MOBA) game. *ASEAN J. Sci. Tech. Report.* **2024**, *27*(5), e252289. https://doi.org/10.55164/ajstr.vx27i5.252289.

Article history:
Received: January 4, 2024
Revised: September 5, 2024
Accepted: September 6, 2024
Available online: September 7, 2024

Publisher's Note:
This article has been published and distributed under the terms of Thaksin University.

Abstract: Realm of Valor (RoV) is a famous multiplayer online battle arena (MOBA) game. An average of 25 million games are played daily in Thailand alone. The game also competes in international events, with millions of U.S. dollars in the prize pool. However, the game is very complex and requires a player to have high experience to win. In particular, the hero selection process that each user has to perform at the beginning of each game because the set of selected heroes can affect the game outcome, but there are many heroes to be selected. This paper compares machine learning techniques to predict the winner's side based on the player's and opponent's selection of heroes and the relationship among the selected heroes. Three traditional machine learning techniques, namely, k-Nearest Neighbor, Logistic Regression, and Decision Tree, are compared against ensemble learning with their optimized parameters. The algorithms are evaluated using k-fold cross-validation, and the accuracy of each algorithm is measured. The results show that winning predictions can be improved by considering the relationship among selected heroes. Also, ensemble learning can compete with traditional learning.

Keywords: Winner prediction; E-Sport; Machine learning; Ensemble learning

1. Introduction

The game industry has expanded rapidly in recent years, especially the MOBA (Multiplayer online battle arena) genre, where multiple players, divided into two teams, compete with each other on an online game battlefield. The expansion leads to professional game competition events, i.e., e-sports, which have been arranged nationally and internationally. In some countries, one of the most popular e-sport games is Realm of Valor (RoV), also known as Arena of Valor (AoV). This game is a variation of Honor of Kings (HoK), a Chinese online game with some internationalized characters. However, they share the same gameplay and mechanism.

RoV was first released to the general public in 2016 in many Asian countries, such as Indonesia, Vietnam, Taiwan, and Thailand. In Thailand alone, within five years after release, there are 45,961,817,910 gameplays, an average of 25 million games played daily. Moreover, many international RoV e-sport events have been organized. For instance, AIC 2021 (Arena of Valor International Championship 2021) has been organized in many countries, e.g., Vietnam, Taiwan, and Thailand, with a prize pool of $1,000,000. In 2022, AIC increased the prize pool to $2,000,000, a 100% increase from the previous year.

In RoV, a player selects a set of five "heroes" to be used throughout a game from a collection of 109 characters, which are added occasionally by the developers. Then, they form a team of five positions: Support, Carry, Mage, Off-line, and Jungle. Each position has its specific function, and the selected hero will affect that function. Therefore, selecting a hero at the beginning of the game is crucial for the game outcome. However, because (1) there are many characters to select from, (2) the rule inhibits players from selecting the same hero in the subsequent game, and (3) the rule that all players ban the hero selection of the opponent; selecting a set of heroes is a complicated process and require a massive amount of information, for instance, the previous selection of opponent, the outcome, the selectable heroes, among the others. Thus, this selection process generally requires the player's initiative and is unreliable.

In this paper, the selection of heroes is used to predict the winning outcome of a game based on records. This paper considers synergy, i.e., a combination of heroes in the same team, and counter, i.e., a combination of heroes from different teams, in the training process to improve prediction accuracy. The game records used in this research are from official international e-sport events such as AIC 2022 and the 31st Southeast Asian Games e-sport event. The training dataset includes the list of heroes from each player, the order of usage, and the result. This research investigates two training strategies; first, individual machine learning algorithms, i.e., k-Nearest Neighbor (KNN), Logistic Regression (LR), and Decision Tree (DT) are studied. They are selected based on their popularity in current literature. Second, this paper proposes to use ensemble learning to improve prediction accuracy. The model is evaluated for accuracy against the actual results. The structure of the paper is as follows: the second section discusses related works. The studied algorithms and datasets are presented in the third section. The following section shows the evaluation results, and the final section concludes and discusses future works.

2. Related Works

A technical report by Kinkade, Yul, and Lim studied the features set that improves win prediction accuracy [1]. The report uses logistic regression to consider *Synergy* and *Counter* in the past game records. This approach can improve the win prediction accuracy based on hero selection alone, with the highest prediction accuracy of 73%.

Semenov et al. proposed using player skills, e.g., normal, high, and very high skills, to partition past game records [2]. Then, the matching algorithm in Dota2, MMR, matches two players with similar skills to balance the game. To predict the winning side, Naïve Bayes, logistic regression, factorization machines, and gradient boosting of decision trees are used to predict the winning players. The study results show that the prediction of winning results of normal-skill players yields higher accuracy than that of higher-skill players. Moreover, factorization machines show the highest prediction accuracy among all studied algorithms.

Chen et al. proposed using the Monte-Carlo tree search (MCTS) and artificial neural network (ANN) to predict winners based on past games [3]. In the paper, the data consists of human and artificial intelligence (A.I.) best-of-N games with the condition that each hero can be used only once. The evaluation result shows that MCTS that use value networks with long-term value can outperform the MCTS without long-term value, using a random approach, the highest winning approach, and the baseline approach to predict a higher winning rate. On the other hand, artificial neural networks show better performance on winning prediction.

Hodge et al. utilize a gradient boosting machine (GBM), logistic regression (LR), and random forest (R.F.) to predict the winning party of Dota 2 [4]. The games used in this research include both public games and professional tournament games. In the dataset from professional tournament games, the first five minutes of the game screen are recorded as time series for live prediction. The evaluation results show that the algorithms can predict the winner with 74.59% accuracy in professional games and 77.51% in hybrid games (public + tournament). Then, the research implements live predictions, which predict the winner while the game is still in play; the accuracy of this approach is 85% after the game is played for the first five minutes.

Tian et al. proposed a Hero featured Network (HFN) that uses past games from Honor of Kings and considers the *Synergy* and *Restraint* features of the game [5]. HFN considers three important properties, gold, kills, and fortress, to predict the winning side. The research focuses on these two features and when to use them to predict the game's winning. By considering when to use these features, HFN can outperform LSTM, TSSTN, Transformer, LR, and SVM.S.

Song et al. proposed training logistic regression with past games of Dota2 [6]. This research focuses on combo heroes with an effectiveness of at least 50 combos. Then, the research uses Stepwise Regression for heroes with small past game data. K-fold cross-validation was used to evaluate the model. The result shows that at least 60 features are required to have low prediction error.

Yang et al. studied real-time win prediction of Honor of Kings with a two-stage spatial-temporal network (TSSTN) model [7]. The model considers six features: gold, kill, tower, wild resource, soldier, and heroes. The prediction accuracy shows that TSSTN can outperform the heuristic approach but performs slightly worse than Fully Connected Network (FCN) and LSTM.

Yang et al. used LTSM and Transformer to predict the winning side, who can kill the boss (Tyrant), the hero who will kill next, and the hero who will be killed next [8]. The results show that the prediction accuracy keeps increasing when the game progresses; however, the prediction accuracy reduces at the end of the game. The results also show that, in general, Transformer can outperform LSTM.

McGuire et al. proposed using an artificial neural network (ANN) with three, five, and seven fully connected layers to predict the DotA2 winning side [9]. The ANN is configured with a feed-forward configuration with a non-linear sigmoid function in each layer. Two features are considered: hero selection and hero selection/time. Comparing three configurations, five fully connected layers show the best performance at 0.73, while using just hero selection shows better performance than hero selection/time.

Conley and Perry proposed using logistic regression (LR) and K-nearest neighbors (KNN) to predict the winning side by using past games from DotA2 [10]. This work considers both *Synergy* and *Counter* features. The prediction accuracy of Logistic Regression is about 69.8% when trained with 18,000 games, while K-Nearest Neighbors can archive 70% when trained with 50,000 games.

Kalyanaraman proposed using logistic regression, genetic algorithm, and augmented regression to predict the winning side of a DotA2 match from past games [11]. This work also considers both *synergy* and *countering* features for hero selection. The paper observes that the accuracy of augmented regression is improved when augmented with a genetic algorithm. Comparing the three algorithms, Logistic Regression shows the best performance with 75.2% accuracy, while Genetic Algorithm and Augmented Regression can archive 74.1% and 68.4%, respectively. However, augmented regression has the highest recall performance, at 90.9%.

Do et al. proposed using player-champion experience to predict the winning side in League of Legends (LoL) [12]. The champion in this game is similar to the hero in RoV. The paper employs Support Vector Machine, k-Nearest Neighbors, Random Forrest, Deep Neural Network, and Gradient Boosting as prediction models, and the results show that Gradient Boosting has the highest accuracy at 75.4%, followed closely by Deep Neural Network at 75.1%. However, Gradient Boosting has the highest standard error compared with the others, indicating the algorithm is unstable. Hence, a Deep Neural Network is the best choice because it has high accuracy and is stable.

Sena and Emanuel investigated winning prediction in Mobile Legends by considering average gold, average level, total kills, first blood, first turtle, and first lord, besides the hero selections [13]. The results show that the paper can achieve 82% and 80% accuracy using Artificial Neural Network and Random Forrest, respectively. The data set in this work comes from 600 competition events.

Costa et al. proposed to use banned champions, picked champions, player statistics, picked champions + players statistics, and banned champions + picked champions + player statistics as features in winning prediction [14]. The research uses a Decision Tree, Naïve Bayes, k-Nearest Neighbor, Support Vector Machine, Random Forest, and Linear Regression. The results show that Random Forrest and Linear Regression can achieve 97% accuracy when using player statistics.

Tuzcu et al. investigated the impact of feature selections on the winning prediction accuracy of League of Legends [15]. The research employed a Gini score-based algorithm to select the top ten features with Random Forrest, Decision Tree, Naïve Bayes, Logistic Regression, Gradient Boosting, LightGBM, and AdaBoost. The result shows that by selecting the top ten features, the algorithm's accuracy can be improved;

for example, Logistic Regression is improved from 89% to 98%, while Gradient Boosting is improved from 96% to 98%.

Outside of e-sports, machine learning is also used to predict winners in traditional sports. Ishi et al. studied two machine learning algorithms, Logistic Regression and Support Vector Machine, to predict the winning side of the One Day International Cricket match [16]. The two algorithms are studied as individual and as ensemble approaches. The features used in this study are the force to hit the ball, the scoring pattern, and the team's overall strength. The results show that both Logistic Regression and Support Vector Machine, as individual algorithms, can archive up to 96.3%. However, when combining Logistic Regression and Support Vector Machine into an ensemble, the accuracy can increase to 96.07%.

Table 1. Related Works Summary

Related Works	Game	Dataset Criteria			Feature Criteria		
		Pre-Game	**Tournament**	**Global Ban Pick**	**Hero Selection**	**Synergy**	**Counter**
[1, 10]	Dota 2	Y	N	N	Y	Y	Y
[2]	Dota 2	Y	N	N	Y	N	N
[3]	HoK	Y	N	Y	Y	N	N
[4]	Dota 2	Y	Y	N	Y	N	N
[5]	HoK	N	N	N	Y	Y	Y
[6, 9, 11]	Dota 2	Y	N	N	N	N	N
[7]	HoK	N	N	N	N	N	N
[8]	HoK	Y	N	N	N	N	N
[12]	LoL	Y	N	N	Y	N	N
[13]	LoL	Y	Y	N	Y	N	N
[14]	LoL	Y	N	Y	Y	Y	N
[15]	LoL	Y	N	N	Y	N	N
This paper	**RoV**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**

Table 1 shows a summarization of related works. Two criteria were considered: dataset criteria and feature criteria. The dataset criteria consider three sets of parameters: pre-game parameters, tournament parameters, and global ban pick parameters. The feature criteria include *Hero Selection*, *Synergy*, and *Counter*. These are the features that are presented in the game. Dota 2 and League of Legends (LoL) are popular MOBA games similar to RoV. Compared with the works in the literature, this paper proposes considering *Hero Selection*, *Synergy*, and *Counter* features. Also, this paper predicts the game result in the pre-game period and considers the global band pick rule. Finally, this paper uses data from international tournament events where professional, top-of-the-class players play.

3. Materials and Methods

This research considers two machine learning approaches. First, a set of individual algorithms, K-Nearest Neighbor (KNN), Logistic Regression (LR), and Decision Tree (DT), will be trained with the best parameters as a baseline. Then, ensemble learning is proposed with voting and stacking decision-making. The proposed algorithm is validated using 5-fold cross-validation, and the f-score is measured to show the accuracy of the proposed algorithm.

3.1 Individual Learning Models

In this research, three individual machine learning algorithms are selected to predict the winning side. They are selected based on popularity in literature; thus, they can present the current state-of-the-art research in this area. The high-level concepts of each algorithm are as follows.

K-Nearest Neighbors (KNN) is a non-parametric supervised learning method that can be used for classification and regression. In classification, the output is the maximum count of a class of nearest neighbors. In regression, the output is the property value, which is the average of the values of k nearest neighbors.

Logistic Regression (LR) models the probability of an event by having the log odds for the event to be a linear combination of independent variables. Logistic regression estimates the probability of an event occurring based on the linear combination of independent variables of the training dataset.

Decision Tree (DT) is a predictive model to conclude a set of observations. Expressly, in classification, leaves of decision trees represent class labels, and branches represent conjunctions of features that lead to a particular leaf.

3.2 Ensemble Learning Models

Ensemble learning uses multiple machine-learning techniques to solve the same problem. Then, the outputs of the techniques are combined to create a final result. In particular, an ensemble function combines multiple outputs to form a better output than an individual one. Many ensemble functions exist, such as Bayes' optimal classifier, boosting, bucket of models, and bagging. This research utilizes two ensembles, namely, stacking and voting.

Stacking or stacked generalization trains the model by combining the prediction of several other learning algorithms. The process starts by training all the other learning algorithms, called based estimators, with the available data. The prediction performance of all algorithms is recorded. Then, a final estimator, in this paper, either KNN, LR, or DT, is trained to make a final prediction using all the predictions of the base estimators as additional inputs.

Voting combines prediction results from many machine learning algorithms with the same data. There is no assumption on the accuracy or performance of the algorithms. When the problem is classified, the result will be the class with the majority among all algorithms. On the other hand, when the problem is prediction, the result will be the average of all results. Furthermore, the voting process can be either soft or hard voting. The class with the highest vote count will be the final result in hard voting. Soft voting summarizes the average probability of each class and then declares the winner as having the highest weighted probability.

3.3 Datasets

This research uses past game data from professional RoV tournaments from the AIC 2022. The tournament employs the global band pick rule and consists of 1,017 games. Following are the rules of the tournament that are applied to the dataset:

Global ban pick: In a best-of-three or best-of-five game, the global band pick rule disallows players to choose the same heroes from the previous round. However, this rule will be applied to the first six games on a best-of-seven game. However, in the seventh game, any heroes can be used.

Right to pick the playing side: The team on the left-hand side has the right to pick the playing side in the first game. In the following games, the loser can pick the playing side.

Ban pick order: in the first ban period, the blue team will pick two heroes to be banned from the Red team, and vice versa. Then, the blue team will pick the first hero. Next, the red team will pick their first two heroes. This selection of two heroes will be switched for the blue and red teams. Finally, the red team will pick their last hero. Then, the second ban period will be started, repeating the same process.

Table 2. Example Data Used in This Research.

ROfflane	RJungle	RMage	RCarry	RSupport	BSupport	BCarry	BMage	BJungle	BOfflane	Result
Tachi	Keera	Liliana	Tel'Annas	Roxie	Xeniel	Laville	Lorion	Kriknak	Veres	0
Airi	Skud	Dirak	Slimz	Payna	Arum	Elsu	Krixi	Tulen	Yena	0
Qi	Errol	Yue	Laville	Lumburr	Baldum	TheJoker	Krixi	Aoi	Florentino	1

Table 2 shows example data from three RoV games. Each row represents data from a game, while each column shows a feature. The first five columns are selected heroes of the Red Team, with the prefix R. The following five columns are heroes from the Blue Team, with the prefix B. Each hero has their position, e.g., offlane or jungle. The names of selected heroes are in each row's first column to the tenth column. The final column is the end game result; 0 is the Red Team win, while 1 is the Blue Team win. The field type of the 1^{st} – 10^{th} columns is a string, while the last is an integer. However, the data must be pre-processed before applying to the model, as shown in the next section.

3.3 Feature Set

Because the number of heroes in each game is limited to only 5 per team from the 109 heroes, many heroes will not be selected. To represent the *Hero Selection*, one-hot encoding is used to represent the red team and blue team hero selection, as shown in Equations 1 and 2, respectively. In the equations, X_i represents the i^{th} feature. Hence, for the hero's selection, there will be 218 features.

$$X_{0+i} = \begin{cases} 1, if\ hero\ i\ is\ on\ red\ team \\ 0, otherwise \end{cases} \quad (1)$$

$$X_{109+i} = \begin{cases} 1, if\ hero\ i\ is\ on\ blue\ team \\ 0, otherwise \end{cases} \quad (1)$$

Then, the combination of heroes in the same team will be considered to represent the *Synergy* property among heroes. Let S_{ij} be the winning ratio when hero *i* and hero *j* are on the same team. Equations 3 and 4 show how to measure the synergy property of the red team, S_R, and the blue team, S_B, respectively. Then, the synergy feature is measured from the difference in the total synergy of the red and blue teams, as shown in Equation 5 as the 219^{th} feature.

$$S_R = \sum_{i\in R} \sum_{j\in R, i\neq j} S_{ij} \quad (2)$$

$$S_B = \sum_{i\in B} \sum_{j\in B, i\neq j} S_{ij} \quad (3)$$

$$X_{218} = S_R - S_B \quad (4)$$

On the other hand, *Counter* property considers how a hero on one team impacts a hero on the other. Let C_{ij} be a winning rate when hero *i* is used against hero *j* of the other team. Then, the counter value of the red team is calculated as in Equation 6, where *R* is the set of heroes of the red team and *B* is the set of heroes of the blue team. Next, the counter value of the blue team can be calculated from $C_B = 1 - C_R$. Finally, the counter property of the red team can be assigned as the 220^{th} feature, i.e., $\boldsymbol{X_{219} = C_R}$. There is no need to include C_B feature.

$$C_R = \sum_{i\in R} \sum_{j\in B} C_{ij} \quad (5)$$

There will be 220 features used in this paper to train machine learning models.

3.4 Methodology

This research investigates eight machine learning prediction models: three traditional algorithms and five ensemble configurations. The models are evaluated using the following process.

1. The AIC2022 dataset is encoded using one-hot encoding, as mentioned in section 3.3, to create feature set.

2. Each model's optimized parameter is indicated, as discussed in section 4.2-4.7. For ensemble learning with voting, there is no need to find an optimized parameter.

3. The encoded data in step 1 is used to evaluate individual models using k-fold-validation, as mentioned in section 4.1

4. The evaluation results are compared as shown in section 4.8 and 4.9

5. Unseen data from the 31st Southeast Asian Games e-sport event is tested against the models to evaluate the generality of the models.

4. Results and Discussion

This paper evaluates three individual machine learning algorithms, namely, K-Nearest Neighbor (KNN), Logistic Regression (LR), and Decision Tree (DT), against five ensemble learning methods using either voting or stacking. The algorithms will be tested on several scenarios with the combination of the following properties: *Hero Selection*, *Synergy,* and *Counter*. The dataset contains 1,017 games collected during AIC 2022. The algorithms will be tested with accuracy to predict the winning side. For each algorithm, the optimized parameter is identified first and then used in the comparison.

4.1 Validation Method

To validate the prediction outcome from the machine learning algorithms, this paper uses k-Fold Cross-Validation with k is five. K-Fold Cross Validation creates a set of data subsets, five subsets in this paper, then uses one as testing data while the rest is used for training. Then, in the next round, another subset is selected as testing data, while the rest is used for training. This process is repeated until all subsets are selected as testing data once. The output of all subset tests will be averaged and used as the result of the test. The k-fold cross-validation is used in this paper because of the limited dataset size.

4.2 Optimized Parameter for k-Nearest Neighbors

To properly use k-Nearest Neighbors, a value of k that is suitable for the dataset and application must be identified.

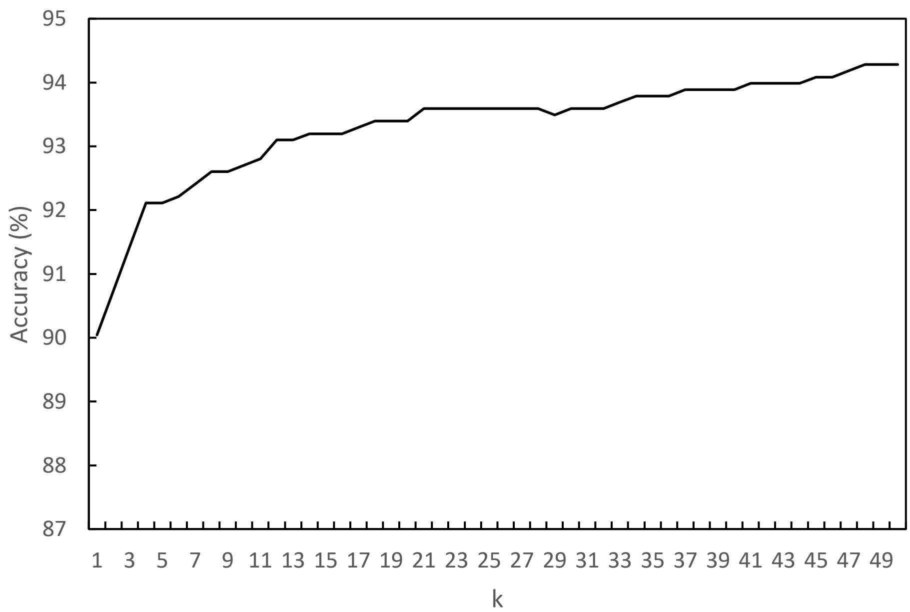

Figure 1. Impact of k Value on Prediction Accuracy

Figure 1 shows the impact of the k value on k-Nearest Neighbors accuracy. The feature set used in this result combines *Hero Selection, Synergy,* and *Counter.* The k value is evaluated in the range of 1 to 50. K-fold cross-validation with five folds is used to measure the algorithm's accuracy. The result shows that the best value of k is 49, which will be used in this paper.

4.3 Optimized Parameter for Logistic Regression

In order to properly use Logistic Regression, a value of C, i.e., regularization strength, that is suitable for the dataset and application needs to be identified.

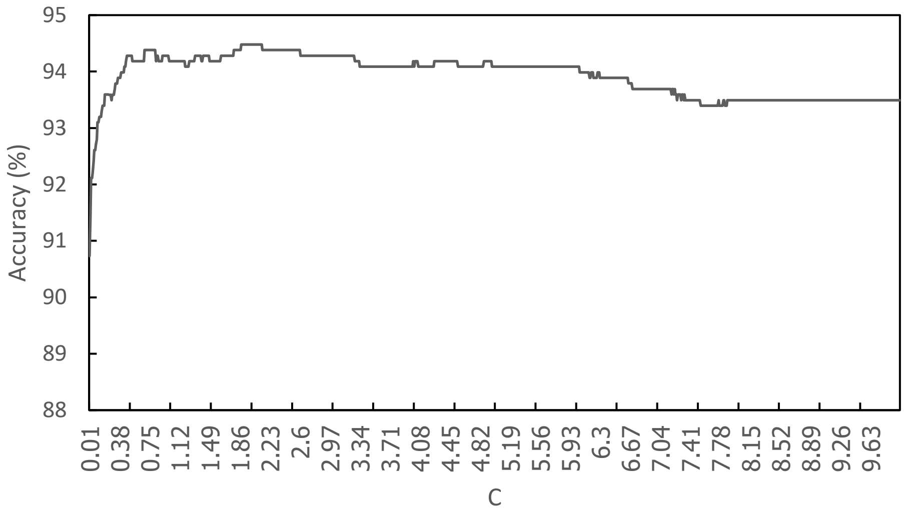

Figure 2. Impact of C Value on Prediction Accuracy

Figure 2 shows the impact of the C value on Logistic Regression accuracy. The feature set used in this result combines *Hero Selection, Synergy,* and *Counter.* The value C is evaluated in the range of 0.01 to 10.0. K-fold cross-validation with five folds is used to measure the algorithm's accuracy. The result shows that the best value of C is 1.89, which will be used in this paper.

4.4 Optimized Parameter for Decision Tree

In order to properly use a decision tree, a tree max depth value suitable for the dataset and application needs to be identified.

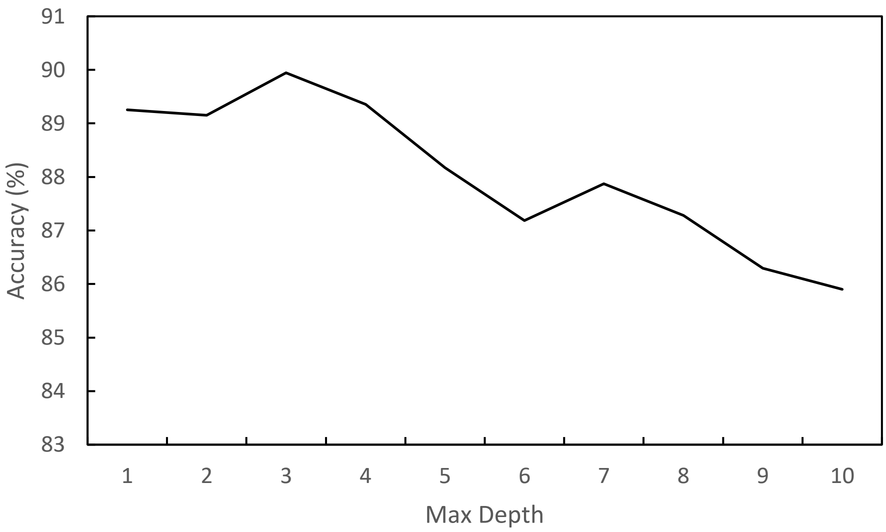

Figure 3. Impact of Max Depth on Prediction Accuracy

Figure 3 shows the impact of max depth value on Decision Tree accuracy. The feature set used in this result combines *Hero Selection, Synergy,* and *Counter.* The depth value is evaluated in the range of 1 to 10. K-fold cross-validation with five folds is used to measure the algorithm's accuracy. The result shows that the best value of max depth is 3, which will be used in this paper.

4.5 Optimized Parameter for Stack Ensemble Learning with KNN

In order to properly use k-Nearest Neighbors as the final estimator in ensemble learning with stack, a value of k that is suitable for the data set and application needs to be identified.

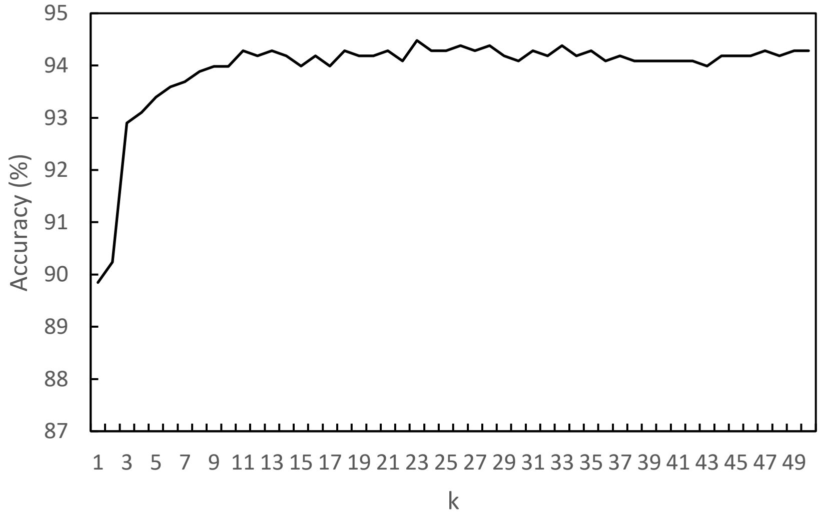

Figure 4. Impact of k Value on Prediction Accuracy

Figure 4 shows the impact of the k value on the accuracy of ensemble learning with stack and k-nearest Neighbors as the final estimator. The feature set used in this result combines *Hero Selection*, *Synergy*, and *Counter.* The k value is evaluated in the range of 1 to 50. The k-fold cross-validation with five folds measures the algorithm's accuracy. The result shows that the best value of k is 23, which will be used in this paper.

4.6 Optimized Parameter for Stack Ensemble Learning with LR

In order to properly use logistic regression as a final estimator in ensemble learning with stack, a value of C, i.e., regularization strength, that is suitable for the data set and application, needs to be identified.

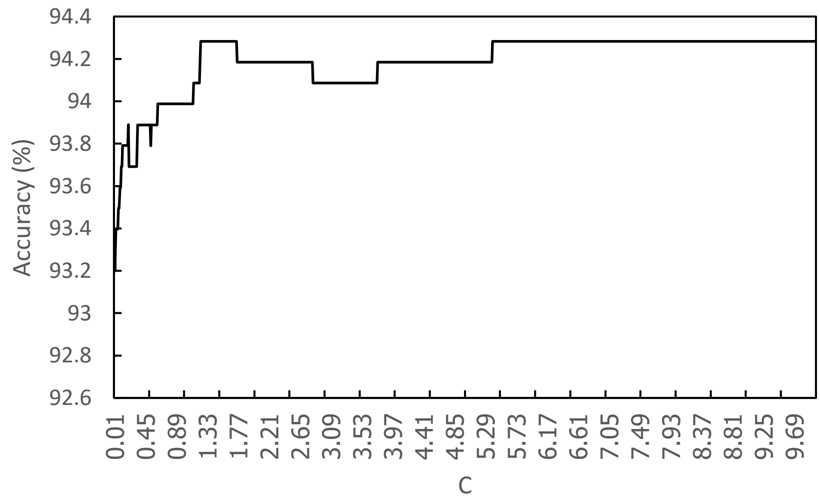

Figure 5. Impact of C Value on Prediction Accuracy

Figure 5 shows the impact of the C value on the accuracy of ensemble learning with stack and Logistic Regression as the final estimator. The feature set used in this result combines *Hero Selection, Synergy,* and *Counter.* The value C is evaluated in the range of 0.01 to 10.0. K-fold cross-validation with five folds measures the algorithm's accuracy. The result shows that the best value of C is 1.24, which will be used in this paper.

4.7 Optimized Parameter for Stack Ensemble Learning with DT

In order to properly use the Decision Tree as the final estimator in ensemble learning with stack, a value of tree max depth suitable for the data set and application needs to be identified.

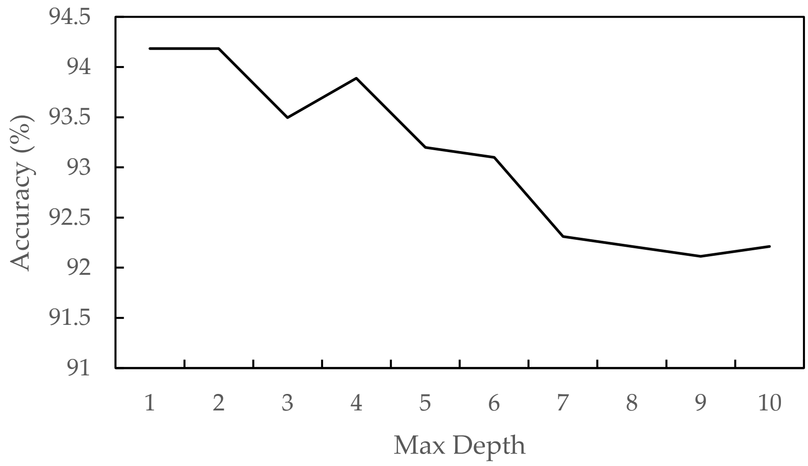

Figure 6. Impact of Max Depth on Prediction Accuracy

Figure 6 shows the impact of max depth value on the accuracy of ensemble learning with stack and Decision Tree. The feature set used in this result combines *Hero selection, Synergy,* and *Counter.* The depth value is evaluated in the range of 1 to 10. The k-fold cross-validation with five folds measures the algorithm's accuracy. Interestingly, the result shows that the best max depth values are 1 and 2. When the value of max depth increases, the accuracy is decreased. Thus, the value of max depth will be used in this paper.

The optimized parameter from Section 4.2 – 4.7 will be used in the subsequent evaluation.

Table 3. Comparing Prediction Accuracy with K-fold Validation.

	Method	**Hero Selection**	**Synergy**	**Counter**	**Synergy + Counter**	**Hero Selection + Synergy**	**Hero Selection + Counter**	**Hero Selection + Synergy + Counter**
Individual	**KNN**	0.525457	0.872995	0.839987	0.**883625**	0.857321	0.844456	0.878027
	LR	0.515958	0.876351	0.845019	0.892579	0.874669	0.878587	**0.924457***
	DT	0.523787	0.875239	0.842783	**0.881942**	0.872447	0.843340	**0.881942**
Ensemble	**Vote (Soft)**	0.531059	0.876353	0.843907	0.887537	0.879713	0.876354	**0.919438**
	Vote (Hard)	0.539458	0.875232	0.842226	0.887542	0.870760	0.857892	**0.899289**
	Stacking (KNN)	0.498029	0.868521	0.836632	0.884734	0.873552	0.869633	**0.916631**
	Stacking (LR)	0.522660	0.877472	0.842228	0.888659	0.878032	0.877468	**0.922787**
	Stacking (DT)	0.514822	0.876913	0.839987	0.885859	0.870749	0.874672	**0.923898**

4.8 Prediction Accuracy with K-fold Validation

After indicating all proper parameters for each algorithm, the algorithms are evaluated against each other, as shown in Table 3. The accuracy is measured using k-fold cross-validation with five folds. The prediction accuracy is measured as the average of the prediction on each testing subset. The value with a bold number is the best in each algorithm, while the value with underline is the best when considering a combination of properties, i.e., *Hero Selection, Synergy,* and *Counter.* The value with an asterisk symbol (*) is the best overall.

The result shows that by considering the combination of *Hero Selection, Synergy,* and *Counter* to train the machine learning model, the winner prediction accuracy can be improved in most cases, as shown in the bold number in each row. Considering traditional algorithms, from the result of section 4.2, the value of k for KNN is very high, i.e., 49, which indicates that many clusters are compared with the data set size. Thus, the prediction accuracy of KNN will be sub-optimal, i.e., too specific. On the other hand, the result of section 4.4 shows that the depth of DT is very shallow, which will lead to sub-optimal prediction, i.e., too generic. Therefore, the result of LR is the best among the three traditional algorithms, as shown by the underscored number in each column. Furthermore, ensemble learning shows promising performance in many cases but cannot outperform individual algorithms, i.e., LR, as the best algorithm. However, it can achieve the second-best prediction accuracy with stacking (DT) configuration and utilize all three features very close, i.e., less than 0.000559, which is different from the best.

The result shows that the prediction can improve by using the two additional features to the heroes' selection, and ensemble learning can improve prediction accuracy compared with the individual algorithms.

Table 4. Comparing Prediction Accuracy with F-Score

	Method	Hero Selection	Synergy	Counter	Synergy + Counter	Hero Selection + Synergy	Hero Selection + Counter	Hero Selection + Synergy + Counter
Individual	KNN	0.520192	0.872196	0.836569	**0.882000**	0.855409	0.841983	0.877146
	LR	0.502259	0.875116	<u>0.843147</u>	<u>0.891407</u>	0.872802	<u>0.877524</u>	**<u>0.923809*</u>**
	DT	0.537429	0.875998	0.838133	**0.880860**	0.873353	0.836794	**0.880860**
Ensemble	Vote (Soft)	0.522808	0.876238	0.839832	0.887369	<u>0.877740</u>	0.875259	**0.918888**
	Vote (Hard)	<u>0.553788</u>	0.874884	0.838370	0.886313	0.870047	0.853910	**0.898619**
	Stacking (KNN)	0.475083	0.865899	0.833573	0.882672	0.869444	0.869265	**0.915175**
	Stacking (LR)	0.472897	<u>0.877402</u>	0.838911	0.887390	0.875785	0.875863	**0.921995**
	Stacking (DT)	0.374027	0.876617	0.836780	0.887983	0.874842	0.869067	**<u>0.923089*</u>**

4.9 Prediction Accuracy with F-Score

F-score measures the prediction accuracy from precision and recall of the test. This paper measures a balanced F-score (i.e., F1 score), as shown in Equation 7.

$$F_1 = 2\,\frac{precision\ x\ recall}{precision + recall} \quad (6)$$

Table 4 shows the results of the F-score of each model used in this paper. The results are very similar to the previous section, confirming that using *Hero Selection*, *Synergy*, and *Counter* features in the training model helps achieve the best prediction accuracy in most cases, indicated by the bold number in each row. Also, linear regression shows the best balance on precision and recall; however, stacking ensemble learning with DT performs similarly. As mentioned in the previous section, LR is more suitable for this application than KNN and DT. Moreover, ensemble learning generally can outperform KNN and DT, and ensemble learning with DT can compete with LR

The result shows that the prediction can improve by using the two additional features to the heroes' selection, and ensemble learning can improve prediction accuracy compared with the individual algorithms.

Table 5. Comparing Prediction Accuracy with Unseen Data

	Method	Hero Selection	Synergy	Counter	Synergy + Counter	Hero Selection + Synergy	Hero Selection + Counter	Hero Selection + Synergy + Counter
Individual	KNN	0.555556	0.611111	0.638889	<u>**0.694444**</u>	<u>0.555556</u>	0.583333	<u>0.666667</u>
	LR	0.555556	0.611111	<u>**0.722222***</u>	0.638889	0.527778	0.583333	0.555556
	DT	0.500000	0.611111	0.638889	<u>**0.694444**</u>	0.583333	0.611111	**0.694444**
Ensemble	Vote (Soft)	0.500000	0.611111	**0.638889**	**0.638889**	<u>0.555556</u>	0.583333	**0.638889**
	Vote (Hard)	0.555556	0.611111	0.638889	**0.666667**	0.583333	0.583333	<u>**0.666667**</u>
	Stacking (KNN)	<u>0.638889</u>	<u>0.638889</u>	**0.666667**	0.638889	0.583333	<u>0.555556</u>	0.527778
	Stacking (LR)	0.555556	<u>**0.638889**</u>	**0.638889**	**0.638889**	<u>0.555556</u>	0.583333	0.583333
	Stacking (DT)	0.527778	<u>**0.638889**</u>	**0.638889**	**0.638889**	0.500000	0.583333	0.527778

4.10 Prediction Accuracy with Unseen Data

The models are used to predict unseen data to investigate the generality of the prediction models. The models were trained with the AIC datasets; then, they were tested against the data set from the 31st Southeast Asian Games e-sport event. The testing dataset includes 36 competition games. As shown in Table 5, the result shows that linear regression that considers the *Counter* property shows the highest prediction accuracy, shown in bold number, with 0.722222 accuracy. The result indicates that individual algorithms, such as LR, can be more generalized than ensemble learning. However, the dataset from the 31st Southeast Asian Games is minimal and has only 36 competition games; therefore, this will be left for further study in future works.

5. Conclusions

Realm of Valor (RoV) is a popular multiplayer online battle arena game. This paper proposes to use machine learning to predict the winner's side based on the selection of heroes used in the battle by the player and opponent. Four machine learning techniques, namely, k-nearest neighbor, logistic regression, decision tree, and ensemble learning, are compared with their optimized parameters. The evaluation result shows that by considering three features, namely *Hero Selection*, *Synergy*, and *Counter*, which are features available only in RoV, the winning prediction accuracy can be improved. Also, ensemble learning can outperform individual algorithms as the best prediction algorithm. Future work includes further investigation of the generality of the prediction models and additional features that can be used to improve the models.

6. Acknowledgements

Author Contributions: Conceptualization, N.T. and P.B..; methodology, N.T. and P.B..; software, N.T.; validation, N.T. and P.B..; formal analysis, N.T. and P.B..; data curation, N.T.; writing—original draft preparation, N.T.; writing—review and editing, P.B..;. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] Kinkade, N.; Jolla, L.; Lim, K. Dota 2 win prediction. Technical Report, University California San Diego, La Jolla, CA, USA, **2015**

[2] Semenov, A.; Romov, P.; Korolev, S.; Yashkov, D.; Neklyudov, K.. Performance of machine learning algorithms in predicting game outcome from drafts in Dota 2, In Proceeding of 5th International Conference Analysis of Images, Social Networks and Texts (AIST 2016), **2016**, pp. 26-37.

[3] Chen, S.; Zhu, M.; Ye, D.; Zhang, W.; Fu, Q.; Yang, W. Which heroes to pick? Learning to draft in MOBA games with neural networks and Tree Search, *IEEE T GAMES*, **2021**, *13*(4), 410–421.

[4] Hodge, V. J.; Devlin, S.; Sephton, N.; Block, F.; Cowling, P. I.; Drachen, A. Win prediction in multiplayer esports: Live professional match prediction, *IEEE T GAMES*, **2021**, *13*(4), 368–379.

[5] Tian, P.; Lan, W.; Zhang, X. Hero featured learning algorithm for winning rate prediction of honor of kings, In Proceeding of 24th IEEE Conference on Games (CoG), **2022**, pp. 322-329.

[6] Song, K.; Zhang, T.; Ma, C. Predicting the winning side of DotA2, Technical Report, Stanford University, Stanford, CA, USA, **2015**.

[7] Yang, Z.; Pan, Z.; Wang, Y.; Cai, D.; Shi, S.; Huang, S.-L.; Bi, W.; Liu, X. Interpretable real-time win prediction for honor of kings—a popular mobile MOBA Esport, *IEEE T GAMES*, **2022**, Volume *14*(4), 589–597.

[8] Yang, Z.; Wang, Y.; Li, P.; Lin, S.; Shi, S.; Huang, S.-L.; Bi, W. Predicting events in MOBA games: Prediction, attribution, and evaluation, *IEEE T GAMES*, **2023**, *15*(2), 193-201.

[9] McGuire, S.; Epifano, J. Dota 2 Game Prediction, Technical Report, Rowan University, Glassboro, NJ, USA, **2018**.

[10] Conley, K.; Perry, D. How Does He Saw Me? A Recommendation Engine for Picking Heroes in Dota 2, Technical Report, Stanford University, Stanford, CA, USA, **2013**.

[11] Kalyanaraman, K. To win or not to win? A prediction model to determine the outcome of a DotA2 match, Technical Report, University of California San Diego, La Jolla, CA, USA, 2015.

[12] Do, T. D.; Wang, S. I.; Yu, D. S.; McMilian, M. G.; McMahan, R., P. Using Machine Learning to Predict Game Outcomes Based on Player-Champion Experience in League of Legends, In Proceeding of the 16th International Conference on the Foundations of Digital Games, **2023**, pp. 1 – 5.

[13] Sena, I. G. W.; Emanuel, A. W. R. Mobile Legend Game Prediction Using Machine Learning Regression Method, *Jurnal Teknologi dan Sistem Informasi*, **2023**, *9*(2).

[14] Costa, L. M.; Mantovani, R. G.; Souza, F. C. M.; Xexéo, G. Feature Analysis to League of Legends Victory Prediction on the Picks and Bans Phase, In Proceeding of 23rd IEEE Conference on Games (CoG), **2021**, pp. 1-5.

[15] Tuzcu, A.; Ay, E. G.; Uçar, A. U.; Kılınç, D. A Machine Learning Based Predictive Analysis Use Case For eSports Games, *Artificial Intelligence Theory and Application*, **2023**, *3*(1), 25-35.

[16] Ishi, M.; Patil, D. J.; Patil, D. N.; Patil, D. V. Winner prediction in one-day international cricket matches using Machine Learning Framework: An ensemble approach, *Indian J of Computer Science and Engineering*, **2022**, *13*(3), 628–641.