FIFA World Cup Win Prediction System using Machine Learning

Ashfiqun Mustari
dept. of CSE
Ahsanullah University of Science and Technology
180204067@aust.edu

Abdullah Al Noman
dept. of CSE
Ahsanullah University of Science and Technology
180204042@aust.edu

Yusha Abdullah
dept. of CSE
Ahsanullah University of Science and Technology
180104012@aust.edu

***Abstract*—Machine learning enables us to open a new data-driven world where prediction, classification, filtering, and detection of any kind of data is possible. Though this is an era of Deep Learning, the usage of machine learning is inevitable in some specific fields. Machine Learning (ML) is a subfield of Artificial intelligence (AI) that allows computers to automatically learn from data and past experiences in order to identify patterns and forecast future events with the least amount of human involvement. As football is one of the most popular games in the world, FIFA World Cup adds another level of excitement to that popularity. The vision of this paper is to introduce the prediction of any FIFA World Cup match based on previous World Cup matches. Our main purpose is to bring more excitement to any FIFA World Cup match such that the result of that specific match is predicted by our Machine Learning (ML) models before that match and we can relate it to the actual game. WorldCupMatches and WorldCupWinners are the two datasets we have used. This study is based on modified ensemble model and six different machine learning models for prediction. The model with the best prediction accuracy is Gradient Boosting and the accuracy that model has attained is 62.40%.**

I. Introduction

Prediction before any kind of games adds more enthusiasm to that game that will be played. It can give a boost to the spirit of any supporter or a player to a game. Accidents, mental breakdown of players, weather change and many other variables can change the flow of any game. But it calms our mind if we can rely on a prediction and enjoy the game a little better.

Machine Learning (ML) algorithms can teach a machine how to learn and produce results based on that learning. Machines can perform repetitive activities with a high frequency and can achieve great accuracy without getting bored. In the previous few decades, rule-driven automation has accounted for the majority of all automation. Every time, these rules operate in the same way. On the other hand, machine learning enables computers to draw knowledge from the past and adjust their decisions accordingly.

Machine Learning Algorithms can be divided into three types: Supervised Learning, Unsupervised Learning and Reinforcement Learning. Supervised Learning requires outside supervision for the machine to learn. The labeled dataset is used to train the supervised learning models. Using Supervised Learning, we can solve classification and regression problems. On the other hand, the algorithm needs to act on that data without any supervision in Unsupervised Learning. The unsupervised models can be trained using the unlabeled dataset, which is neither classed nor categorized. The model searches through the vast amount of data in search of meaningful insights rather than producing a predetermined result. These models are used in order to solve the Association and Clustering problems. Finally, through the process of reinforcement learning, an agent interacts with its surroundings by taking actions and learns from the feedback it receives. The agent receives feedback in the form of rewards; for example, he receives a positive reward for each good action and a negative reward for each bad action. The agent is not under any oversight. Reinforcement Learning can be used for any kind of automation in machineries[2]. We have used Supervised Learning to train our Machine Learning (ML) models. Six different Machine Learning (ML) models such as K-Nearest Neighbor (KNN) classifier, Naive Bayes classifier, Decision Tree classifier, Random Forest classifier, Logistic Regression classifier and Support Vector Machine (SVM) classifier are used in this paper. A customized Ensemble Model has been created using Voting Classifier. In this Ensemble Model, the Voting Classifier was performed on K-Nearest Neighbor and Decision tree to enhance the accuracy.

II. Background Study

A. K-Nearest Neighbor (KNN)

KNN determines the distance between a new example and all the other example in the dataset before performing classification or regression tasks on new data [7]. Each new data point is classified by the algorithm based on the existing data points that are comparable to it, and the algorithm stores the complete dataset. KNN solely uses training or ”known” data to make predictions. KNN calculates the distance between data points once the user provides a distance function in order to locate the nearest data points from our training data for any new data point. Using the specified distance, the ”k-neighbors” are the existing data points that are the closest to the new data point. In order to predict the new data label for a classification task, KNN will use the most frequent of all values from the k-neighbors. KNN does nothing more than save the entire dataset and calculate the distance between a new data point and its nearest neighbors without performing any calculations or modeling on top of it. KNN is easy to use and intuitive, in contrast to many other Machine Learning algorithms. It is versatile (you may choose from a variety of distance metrics), changes when new information emerges, and only has one hyperparameter to adjust (the value of ”K”).

B. Naive Bayes

Naive Bayes is a classification method that uses an independence assumption among predictors and is based on Bayes' Theorem [8]. It is used for a wide range of classification applications. A group of classification algorithms built on the Bayes' Theorem are known as naive Bayes classifiers. It is a family of algorithms rather than a single method, and they are all based on the idea that every pair of features being classified is independent of the other. Among them the Gaussian model assumes that characteristics are distributed normally. This indicates that the model thinks that predictor values are samples from the Gaussian distribution if they take continuous values rather than discrete ones.

C. Decision Tree

An approach for supervised learning is the decision tree. For problems concerning classification, it is preferred. It is a tree-structured classifier, where internal nodes stand in for a dataset's features, branches for the decision-making process, and each leaf node for the classification result [5]. The Decision Node and Leaf Node are the two nodes of a decision tree. Whereas Leaf nodes are the results of decisions and do not have any more branches, Decision nodes are used to create decisions and have numerous branches. The given datasets features are used to execute the test or make the decisions. Because the decision tree displays a tree-like structure and may simulate human thought processes while making decisions, the theory behind it is simple to understand.

D. Random Forest

The base of Random Forest is the idea of ensemble learning, which is the process of mixing various classifiers to solve a challenging problem and enhance the model's performance [4]. Instead of relying on a single decision tree, the random forest takes the prediction from each tree and makes its prediction of the final output on the majority votes of predictions. It is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Higher accuracy and overfitting are prevented by the larger number of trees in the forest.

E. Logistic Regression

One of the most often used Machine Learning algorithms, within the category of Supervised Learning, is logistic regression. With a predetermined set of independent factors, it is used to predict the categorical dependent variable [6]. Except in terms of how they are applied, logistic regression and linear regression are very similar. Whereas logistic regression is used to solve classification problems, linear regression is used to solve regression problems. In logistic regression, we fit a ”S” shaped logistic function, which predicts two maximum values, rather than a regression line (0 or 1). It is a significant machine learning algorithm since it can categorize new data using continuous and discrete datasets and make probabilities.

F. Support Vector Machine (SVM)

The SVM algorithm's objective is to establish the best line or decision boundary that can divide n-dimensional space into classes, allowing us to quickly classify new data points in the future [3]. A hyperplane is the term given to this optimal decision boundary. SVM selects the extreme vectors and points that helps in the creation of the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. While using the SVM algorithm, each data point is represented as a point in n-dimensional space (where n is the number of features you have), with each feature's value being the value of a certain coordinate. Then, classification is performed by identifying the hyper-plane that distinguishes the two classes very well.

G. Gradient Boosting

Gradient boosting is one of the most powerful techniques for building predictive models [10]. It is a machine learning technique used in regression and classification tasks.The idea of boosting came out of the idea of whether a weak learner can be modified to become better.It gives a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees. When a decision tree is the weak learner, the resulting algorithm is called gradient-boosted trees; it usually outperforms random forest.A gradient-boosted trees model is built in a stage-wise fashion as in other boosting methods.

H. Voting Classifier

A voting classifier is a machine learning model that gains experience by training on a collection of several models and predicts an output (class) based on the class with the highest probability of being the output [9]. To predict the output class based on the highest majority of votes, it simply averages the results of each classifier that was passed into the voting classifier. The concept is to build a single model that learns from these models and predicts output based on their aggregate majority of voting for each output class, rather than building separate dedicated models and determining the accuracy for each of them.

III. Related Works

A research paper [11] proposes new methods for predicting winner of world cup. In this research they have focused more on implementing random forest. In this research they collected data of all matches from world cup 2002 to world cup 2014. In their research they have implemented 1. Random forests 2. Regression 3. Poisson model.

The final result gives a table which contains the probability of each team qualifying rounds of world cup and winning final match. Another research [12] is based on a regularised Poisson regression model. They have used the dataset of World Cup matches from 1994 to 2010. Their model concentrates on the number of goals a team scores against a specific opponent. In this research they have implemented Poisson ranking and also Lasso estmator.

IV. Dataset

The datasets we have used for our experiment is our own created datasets. Our datasets are structured datasets which are tabular datasets. The first dataset contains the data of world cup matches from year 1930. We have collected these data from internet mostly from Wikipedia. There are 916 data in total. Initially there are 10 features meaning 10 columns in our dataset which are Year, Stage, Home team name, Home team goals, Away team goals, Away team name etc. The second dataset contains the winner of each world cup. We saved our collected data in a excel sheet and then converted into a CSV file.

V. Proposed Methodology

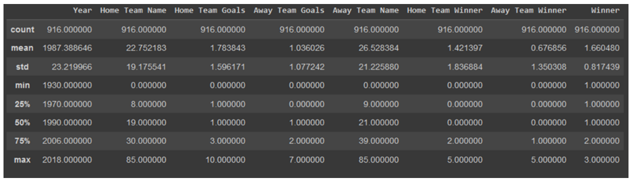
The goal of our experiment is to predict the probability of winning of a team. To achieve this we have divided our work into two parts. 1. Data processing and feature extraction. 2. Implementing Machine Learning Models.

A. Data processing and feature extraction

Our data is a structured data which is in a tabular format. There are 916 rows and 10 columns.10 columns means 10 features. We don't need all these features so we extracted important columns from our dataset. First we have dropped all NULL values from our dataset as it is redundant.

For extracting features we have dropped the columns which we don't need. After dropping these columns we have Year , Home team goals, Away team goals and Away team name, home team winner,away team winner and winner. The last column is the label of our dataset where label 2 means away team winner and 1 means home team winner. Our prediction mostly depends on Home team goals, Away team goals, Winner features. Here is the description of our data after feature extraction and preprocessing.

Here is the histogram of our dataset-



	Year	Home Team Name	Home Team Goals	Away Team Goals	Away Team Name	Home Team Winner	Away Team Winner	Winner
count	916.000000	916.000000	916.000000	916.000000	916.000000	916.000000	916.000000	916.000000
mean	1987.388646	22.752183	1.783843	1.036026	26.528384	1.421397	0.676856	1.660480
std	23.219966	19.175541	1.596171	1.077242	21.225880	1.836884	1.350308	0.817439
min	1930.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000
25%	1970.000000	8.000000	1.000000	0.000000	9.000000	0.000000	0.000000	1.000000
50%	1990.000000	19.000000	1.000000	1.000000	21.000000	0.000000	0.000000	1.000000
75%	2006.000000	30.000000	3.000000	2.000000	39.000000	2.000000	1.000000	2.000000
max	2018.000000	85.000000	10.000000	7.000000	85.000000	5.000000	5.000000	3.000000

Fig. 1. Data Description

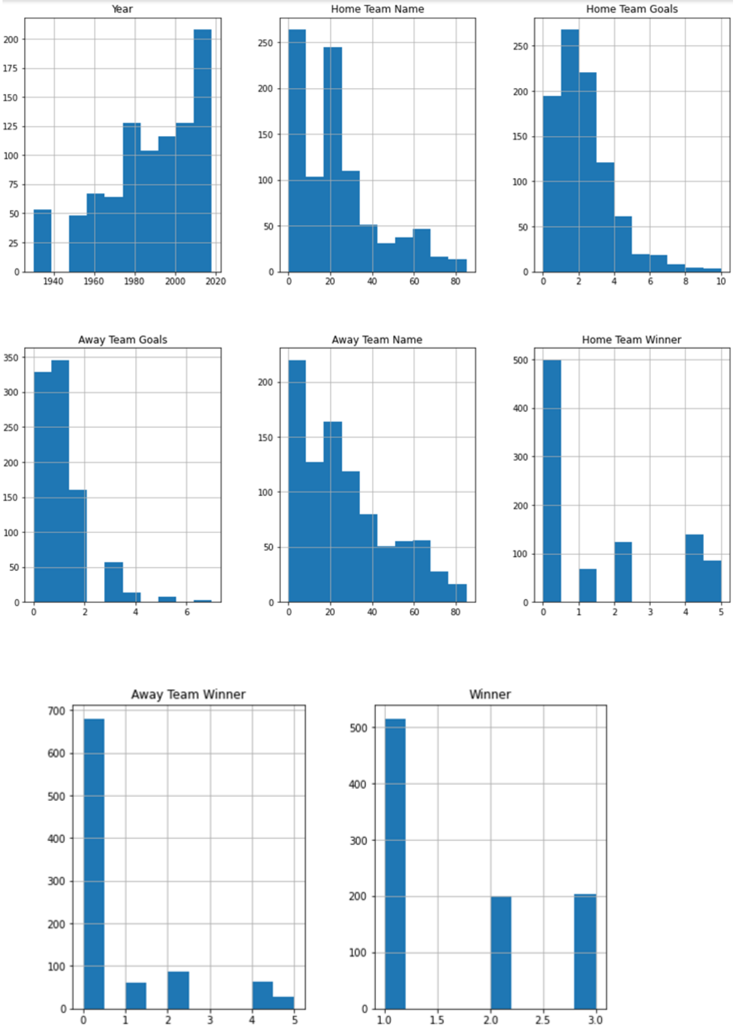


Fig. 2. Data Description

B. Splitting Data

We have augmented then and shuffled our dataset. Then split our dataset in the ratio of 80:20. Meaning if we have 100 data then 80 data will be used for training and rest 20 will be used for testing. After splitting we have: Number of training Data: 1465 Number of testing Data: 367

C. Implementing Machine Learning Models

In our experiment we have implemented six machine learning Models. These are:

1. KNN
2. Naïve Bayes

3. Decision Tree
4. Random Forest
5. Logistic Regression
6. Support Vector Machine SVM
7. Gradient Boosting
8. Ensemble Model

Our experiment requires classification. So we have used classification models. Finally in an attempt to increase accuracy we have implemented Ensemble Model.

VI. Experimental Result

To evaluate our machine learning models, we have used precision, recall, f1-score, support and accuracy as evaluation metrics.

Precision: Precision is one indicator of a machine learning model's performance – the quality of a positive prediction made by the model. Precision refers to the number of true positives divided by the total number of positive predictions (i.e., the number of true positives plus the number of false positives).

Recall: Recall, also known as the true positive rate (TPR), is the percentage of data samples that a machine learning model correctly identifies as belonging to a class of interest—the "positive class"—out of the total samples for that class.

F1-Score: F1 score is a machine learning evaluation metric that measures a model's accuracy. It combines the precision and recall scores of a model.

Support: Support is the number of actual occurrences of the class in the specified dataset. Imbalanced support in the training data may indicate structural weaknesses in the reported scores of the classifier and could indicate the need for stratified sampling or rebalancing.

Accuracy: Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition: Accuracy = Number of correct predictions Total number of predictions.

TABLE I
Experiment-1: K-Nearest Neighbor

SL	Precision	Recall	F1-score
0	0.60	0.55	0.57
1	0.58	0.63	0.61
Accuracy			0.59
Macro Avg	0.59	0.59	0.59
Weighted Avg	0.59	0.59	0.59

TABLE II
Experiment-2: Naive Bayes

SL	Precision	Recall	F1-score
0	0.60	0.61	0.60
1	0.60	0.60	0.60
Accuracy			0.60
Macro Avg	0.60	0.60	0.60
Weighted Avg	0.60	0.60	0.60

TABLE III
Experiment-3: Decision Tree

SL	Precision	Recall	F1-score
0	0.59	0.63	0.61
1	0.60	0.55	0.58
Accuracy			0.59
Macro Avg	0.59	0.59	0.59
Weighted Avg	0.59	0.59	0.59

TABLE IV
Experiment-4: Random Forest

SL	Precision	Recall	F1-score
0	0.61	0.58	0.60
1	0.60	0.63	0.62
Accuracy			0.61
Macro Avg	0.61	0.61	0.61
Weighted Avg	0.61	0.61	0.61

TABLE V
Experiment-5: Logistic Regression

SL	Precision	Recall	F1-score
0	0.61	0.61	0.61
1	0.61	0.62	0.62
Accuracy			0.61
Macro Avg	0.61	0.61	0.61
Weighted Avg	0.61	0.61	0.61

TABLE VI
Experiment-6: Support vector Machine (SVM)

SL	Precision	Recall	F1-score
0	0.61	0.60	0.60
1	0.60	0.61	0.61
Accuracy			0.60
Macro Avg	0.60	0.60	0.60
Weighted Avg	0.60	0.60	0.60

TABLE VII
Experiment-7: Gradient Boosting

SL	Precision	Recall	F1-score
0	0.62	0.62	0.62
1	0.62	0.62	0.62
Accuracy			0.62
Macro Avg	0.62	0.62	0.62
Weighted Avg	0.62	0.62	0.62

TABLE VIII
Experiment-8: Ensemble of KNN and Decision Tree

SL	Precision	Recall	F1-score
0	0.58	0.74	0.65
1	0.64	0.46	0.54
Accuracy			0.60
Macro Avg	0.61	0.60	0.59
Weighted Avg	0.61	0.60	0.59

VII. Result Analysis

A. Accuracy

Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

Accuracy = Number of correct predictions Total number of predictions.

TABLE IX
ACCURACY COMPARISON

Classifier	Accuracy
K-Nearest Neighbor	58.86%
Naive Bayes	60.22%
Decision Tree	59.40%
Logistic Regression	61.31%
Random Forest	60.76%
Support Vector Machine	60.49%
Gradient Boosting	62.40%
Ensemble Model	60.22%

In our experiment, K-Nearest Neighbor(KNN) and Decision Tree performed accuracy below 60%. So we ensembled the models together to bring out a better performance. We have incorporated voting classified to ensemble the two models and the ensemble model has proven to perform better with an accuracy of 60.22%.

B. Receiver Operating Characteristics (ROC) Curve

Performance evaluation is a key role in machine learning. Thus we can rely on an AUC-ROC Curve when it comes to a classification task. The AUC (Area Under The Curve) ROC (Receiver Operating Characteristics) curve is used to evaluate or illustrate the performance of the multi-class classification [1]. It is one of the most crucial evaluation criteria for assessing the effectiveness of any classification model. It is also called AUROC (Area Under the Receiver Operating Characteristics). The curve is plotted between two parameters, which are True Positive Rate or TPR (Sensitivity) and False Positive Rate or FPR (1-Specificity)

Here is ROC curve of the classifiers that we have worked on- From the ROC curve implementation of our models, we have observed that the Gradient Boosting Classifier has the best AUCROC score (0.624). It has proven to be the best classifier for our dataset.

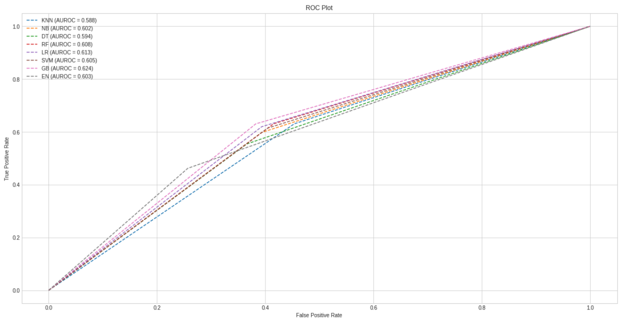


Fig. 3. ROC curve

VIII. FUTURE WORK

We have done our study on a limited features and limited number of matches from FIFA world cup. As more world cups will happen, more data can be incorporated in the dataset. Besides, more features such as the weather condition of the matches, fitness of the players, refree selection, and many more can be explored to train the model on different aspects and make better predictions.

IX. CONCLUSION

In our study, we have built a system to predict the win probability of FIFA world cup matches with the help of Machine Learning. We have studied 5 different machine learning classifiers which are K Nearest neighbor, Naive Bayes, Decision Tree, Logistic Regression, and Support Vector Machine. We have also studied experimented with two ensemble classifiers which are Random Forest and Gradient Boosting. Among all the classifiers, Gradient Boosting classifier has the best accuracy which is 62.40%. On the other hand, K-Nearest Neighbor and Decision tree had accuracy less than 60%. So we ensembled the two models to improve the accuracy, which then improved to be 60.22%. We have experimented with limited features now. In the future, more features can be explored to improve the performance of the prediction system.

X. REFERENCES

[1] Hoo, Z. H., Candlish, J., & Teare, D. (2017). What is an ROC curve?. Emergency Medicine Journal, 34(6), 357-359.

[2] Wiering, M. A., & Van Otterlo, M. (2012). Reinforcement learning. Adaptation, learning, and optimization, 12(3), 729.

[3] Awad, M., Khanna, R., Awad, M., & Khanna, R. (2015). Support vector machines for classification. Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers, 39-66.

[4] Paul, A., Mukherjee, D. P., Das, P., Gangopadhyay, A., Chintha, A. R., & Kundu, S. (2018). Improved random forest for classification. IEEE Transactions on Image Processing, 27(8), 4012-4024.

[5] Song, Y. Y., & Ying, L. U. (2015). Decision tree methods: applications for classification and prediction. Shanghai archives of psychiatry, 27(2), 130.

[6] Feng, J., Xu, H., Mannor, S., & Yan, S. (2014). Robust logistic regression and classification. Advances in neural information processing systems, 27.

[7] Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003). KNN model-based approach in classification. In On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings (pp. 986-996). Springer Berlin Heidelberg.

[8] Jiang, L., Wang, D., Cai, Z., & Yan, X. (2007). Survey of improving naive bayes for classification. In Advanced Data Mining and Applications: Third International Conference, ADMA 2007 Harbin, China, August 6-8, 2007. Proceedings 3 (pp. 134-145). Springer Berlin Heidelberg.

[9] Zhang, Y., Zhang, H., Cai, J., & Yang, B. (2014, May). A weighted voting classifier based on differential evolution. In Abstract and applied analysis (Vol. 2014). Hindawi.

[10] Sun, R., Wang, G., Zhang, W., Hsu, L. T., & Ochieng, W. Y. (2020). A gradient boosting decision tree based GPS signal reception classification algorithm. Applied Soft Computing, 86, 105942.

[11] Groll, A., Ley, C., Schauberger, G., Van Eetvelde, H. (2018). Prediction of the fifa world cup 2018-a random forest approach with an emphasis on estimated team ability parameters. arXiv preprint arXiv:1806.03208.

[12] Groll, A., Schauberger, G., Tutz, G. (2015). Prediction of major international soccer tournaments based on team-specific regularized Poisson regression: An application to the FIFA World Cup 2014. Journal of Quantitative Analysis in Sports, 11(2), 97-115.