ML mini project phase 2

by 2019a7ps0244u HIMANSHU RATHI

Submission date: 08-Dec-2022 01:45AM (UTC+0400)
Submission ID: 1974618697
File name: ML_Assignment_Phase_2_2019A7PS0244U_2019A7PS0205U.pdf (579.46K)
Word count: 5192
Character count: 26866

CRICKET MATCH OUTCOME PREDICTION USING MACHINE LEARNING

Himanshu Rathi
Undergraduate in Computer Science
Birla Institute of Technology & Science, Pilani - Dubai
Dubai, UAE
f20190244@dubai.bits-pilani.ac.in

Vignesh Nair
Undergraduate in Computer Science
Birla Institute of Technology & Science, Pilani - Dubai
Dubai, UAE
f20190205@dubai.bits-pilani.ac.in

***Abstract*— In a world where there is an ever- increasing reliance on statistics for distinct reasons, we sought to create this outcome predictor for Cricket. Many people believe in relying on those very statistics repeatedly but how can one go through the tedious task of going through all those statistics for each match and each player? It is true, in a game like cricket the statistics determine a lot and could easily tell you the winner. But it is not so easy doing so in a game that keeps evolving with newer additions to the rules quite often, a game that has a multitude of formats, stadiums, grounds, and pitches that behave different, along with unpredictable weather such as dew or drizzles, humid situations, etc. And not to forget the boundary length, there is clearly a lot of factors that come into play in this sport besides the past and present stats of the players involved themselves. To analyze a match, one would have to go through the statistics of a minimum of eleven players, but as suggested, the model will take care of it all by extracting the datasets from official websites by the ICC and well known and reputed pages like ESPN Cricinfo and so on.**

By using the help and power of machine learning we believe we can make a powerful model with the understanding we received from our literature reviews with regards to the topic at hand. So, using that, we intend to train and evaluate the models that have proven to have a greater accuracy over several factors and classes. Now, there are advantages and disadvantages to what we are create as programmers, as useful as this program, it must be used in caution because there is the possibility of misusing this program to aid someone's betting habits which can lead to disastrous outcomes, and we do not want that. The sole purpose of creating the algorithm is give teams and coaches around the world a tool to use to analyze if their chosen team has the strength and ability to get a win under the circumstances they are going to find themselves in, in this way they can prepare according to the result that the model will generate, to either change up the team or retain them for a match. And therefore, that is how this mini project will find its use in this world of ours..

Keywords*— *Cricket Match prediction, Machine Learning, Random Forest, Decision Tree

I. INTRODUCTION

Sports are a testament to what we as people are capable of, throughout history we have borne witness to great feats of skill, bravery, hard-work, and so much more. In recent times, the introduction of television and the monetisation of various sports have made it more competitive and entertaining for the masses. Therefore, there has been a greater interest in sports ever since events have started being televised, and more recently the sudden rise of fantasy leagues, and other such ventures have garnered more attention towards said sports.

But before exploring how all these factors play a key role in the research we are conducting, it is vital to have a better understanding of the domain we intend to conduct our research in, that being said let us start with the very basics of the famous sport, Cricket.

Cricket is a team sport where each team consists of eleven players each and Two teams compete against each other to win. The composition of the eleven players heavily depends on the format and conditions the team may find themselves in. And in certain situations, the strength of the opposition also plays a role in determining the final eleven that make it to the team.

The rules of the sport are plenty, but the very foundation of cricket lies in setting a score for the opposition to chase / in chasing down a score set by the opposition. A score is made or set by the opposition depending on how the teams' players perform in the given innings. To add to all these factors mentioned already. There is also a Toss that is conducted before a match is conducted, and upon winning the toss, the winner is offered a choice to bat first or second. The team decides to bat first or second depending on the opposition, the statistics of the previous matches on the ground, the pitch they are playing on as well as the season of the year it is.

So, one may notice, there are simply a lot of factors that go into deciding a playing 11 for the team and this is without even considering the vast pool of players available and what they bring to the table. But let us break it down further, starting with the format of the game.

Cricket has three primary formats and a few variations of the game for domesticated leagues around the world. The three primary formats are T20Is (Twenty-Twenty Internationals), ODIs (One Day Internationals), Test Cricket. Then on a domestic level there exists a couple of other formats such as The Hundred in England, T10 leagues and even T5 leagues across the world. T20s are played for 20 overs per innings per team, ODIs on the other hand are 50 overs long per innings but to top that, Test cricket on the other hand lasts a grand total of 5 days and approximately 90 overs per day. Now across the formats, the rules of the game are for the most quite similar with some exceptions that are unique to each format.

Regardless as mentioned earlier there are simply too many factors involved for the human mind to comprehend and so we sought out to make an algorithm that could help alleviate that burden of selecting a suitable playing 11 for the team, an algorithm that could take into consideration all of the factors that would affect the odds of winning. An algorithm that would use existing databases and websites to comb through the statistics for the factors we list out to come up with a comprehensive solution for the situation it is about to deal with.

The factors that the algorithm would have to consider would initially begin by checking the past and present form of the players in the squad by checking their most recent scores, strike rates and world ranking. Once it segregates and ranks the players by the role they perform the next step would be to analyse the opponent they are going to play with to come up with possible matchups for the upcoming game.

The result of the toss is not necessarily in control of the team and therefore the algorithm will have to consider both possibilities and make a decision. As and when the algorithm explores both possibilities, it shall simultaneously progress ahead and take into account the quality of the pitch, the covering on the pitch, the average length of the boundaries and as well as the weather for the day to come.

Taking all of these factors into account, clearly the algorithm would require a suitable database and that is where resources like ESPN Cricinfo, Cric Buzz and so on become useful. With the help of a standard search engine, one can easily retrieve the required information from the above-mentioned websites to assist the model in analysing the players involved.

Once again, the goal of this project is to come up with a powerful algorithm that can help teams around the world figure out an ideal playing eleven for the match by considering the several factors in play.

To better explain this, we will go into detail about a real-life situation where the algorithm would succeed. Looking at the IPL for instance, one would know in advance the matches that were to happen. Assuming Team A has a match scheduled with Team B at the M. Chinnaswamy Stadium, the algorithm would help the team realize what match ups would work against Team B as well as how effective they would be at the M. Chinnaswamy Stadium.

Secondly it would analyse the odds of winning batting first or second based on the previous matches most recently at the current ground. Then create two possible line ups that could be used depending on the toss while taking the weather report of the day into account.

The roles that would consist of the line-up would be based off the average score there as well, for in a situation where it is a high scoring ground the algorithm would give a greater priority to All- rounders in the playing 11 to provide a greater depth to batting and ensure that the bowlers that are selected are economically sound bowler as keeping the economy and making it hard to score are vital in the given In this paper we reviewed different methods incorporated by them and formed our problem statement and objectives. The remainder of paper is organized as following: In depth literature survey of different methods and accumulation of key features of all the papers in tabular format in section 2. Problem Statement and Objective in section 3. Implementation in section 4. Methodology and Architecture Diagram in section 5. Result and Discussion in section 6. Conclusion in section 7. References in section 8.

II. LITERATURE SURVEY

In [1], Stylianos Kampakis, and William Thomas, from the University College London thought that being able to predict the results of county cricket matches in the T20 format would be useful to beat the odds provided by betting clubs. They got their data from ESPN Cricinfo, and collected it in separate databases to get a deeper understanding between team statistics and individual statistics then they took it, and split it to train their model and then test it. To figure the outcome they looked at the following models, NB, LR, RF, Gradient boosted decision tree and the final outcome ranged from 55.6% to 62.4% for accuracy.

In [2], Raza Ul Mustafa, M. Saqib Nawaz, M. Ikram Ullah Lali, Tehseen Zia, Waqar Mehmood proposed a ML model based on crowd opinion on social networks. They collected tweets & comments from twitter and cricket blogs. They employed three different techniques to make their forecasts, depending on the total number of tweets sent out by each side before the game, the opinions of fans towards each club, and Twitter users' predictions of the final score. SVM showed best performance amongst the other classifiers used with an accuracy of 75%.

In [3], Dev Karan Singh, Sarthak Agarwal, Sanjeev Gupta, Manisha Singh, Utkarsh Saxena came up with an algorithm that utilised the official IPL database to predict a match winner. Using the IPL database they pre-processed it, and used a supervised model where they split the data to train & test the model. In this paper they primarily relied on Naive Bayes classification, and Support Vector Machines which yielded a 61% to 75% accuracy to determine the winner of a given match.

In [4], Kalpdrum Passi and Nirav Kumar Pandey over in Canada went a different path than most by seeking to predict the amount runs or wickets a player takes in a given match. As most people, they got data from ESPN Cricinfo and processed it for their models. Both classification problems had different results but ended up with the best accuracy from the Random Forest classifier for both problems as compared to the naïve bayes, multiclass SVM and decision tree classifiers.

In [5], K.A.D.A. Pramoda wrote this dissertation and wanted to use ML techniques to measure the result for different T20 matches and even managed with unbalanced datasets and converted them as well for ease of usage. He made use of a few different models namely MLR, Ridge Regression, Lasso Regression, Neural Network, SVM, Decision Tree, RF, and MLP Classifier and their hybrids to compare and see what had the greater accuracy and settled down with a high accuracy of 93.92% on a hybrid model.

In [6], Daniel Mago Vistro, Faizan Rasheed, Leo Gertrude David also combined ML and Data analytics for creating a ML model for cricket winner prediction. Their model utilised the IPL T20 match dataset provided by ICC's official website. Unlike other papers, this paper used SEMMA methodology for their model. The model used Decision tree, RF, XGBoost yielding 94.87, 80.76, 94.23 percent accuracy respectively.

In [7], Pallavi Tekade, Kunal Markad, Aniket Amage, Bhagwat Natekar sought to figure out the factors that could possibly affect a win and so utilised the data from the first 11 seasons of the IPL to isolate factors that would affect the chances a team had of victory. The data ran through a feature extraction process and the extracted data was split to train & test the model. In this paper they primarily relied on SVM, LR, Decision Tree, Bayes Classifier amongst which the Non Linear SVM had the lowest accuracy at 63.05% while a combination of several models gave it a higher accuracy as high as 83% to 90%

In [8], Shivam Tyagi, Rashmi Kumari, Sarath Chandra Makkena, Swayam Swaroop Mishra, and Vishnu S. Pendyala addressed a novel problem i.e., prediction of

cricket match duration of ODI format. The data was collected from Cricsheet and the paper compares LR, LogR, SVM and SVD classifiers for the proposed model. They also used multiple approaches for the model like team based, player combination based, player profile based team strength etc. Their experimental results show that the player combination approach was better and that the RF algorithm gives better prediction than the SVD model in this approach. The model can further be improved by incorporating natural factors like weather, venue etc.

In [9], Abdul Basit, Muhammad Bux Alvi, Fawwad Hassan Jaskani, MajdahAlvi, Kashif H. Memon, Rehan Ali Shah compared performance of RF, Extra Trees, ID3 and C43 for prediction of ICC T20 World Cup 2020. RF technique was able to get the highest accuracy of 80.86% amongst the others. The model can be improved by adding more natural factors like venue, weather forecast etc.

In [10], Wickramasinghe approached the cricket match winner prediction problem with Naive Bayes approach. The model was able to outperform the previous model for prediction accuracy for the winner of an ODI cricket match. The NB approach was able to yield the highest accuracy of 85.71 percent but the sample size was very low.

In [11], Anurag Sinha combined predictive analytics with machine learning to create a model for cricket match winner prediction. They tested their model on 4 modules with different combinations of features and models resulting in different accuracy. Module 4 yields the highest accuracy with 17 features and 6 ML models.

In [12], Nilesh M. Patil, Bevan H. Sequeira, Neil N. Gonsalves, Abhishek A. Singh deployed a website which incorporates their proposed ML model for finding best team line-ups for winning matches. Their dataset was not mentioned in the paper but they mentioned using automated web scraping tools for finding the required data. The paper compared the model with RF, Decision Tree and algorithms. Their lack of data set and less feature selection resulted in poor performance of the model as it was only able to provide line-ups for a selected few players.

In [13], Aman Sahua, Devang Kaushikb, A. Meena Priyadharsinic used ML to provide analysis for both individual players based on their statistics, the team's performance off the player statistics, and use that to predict the results of different matches. They got their dataset from Kaggle and manually encoded it to make it work for their models which included the Multinomial Logistic Regression, RF, AdaBoost to make it smoother for the model with the greater accuracy and it resulted in an overall accuracy of 84.21% for the best model.

In [14], Mazhar Javed Awan, Syed Arbaz Haider Gilani, Hamza Ramzan, Haitham Nobanee, Awais Yasin, Azlan Mohd Zain, and Rabia Javed used the big data approach to tackle the problem of match prediction. The model utilized data from Cricsheet and Linear Regression (LR). LR was able to produce better predictions in Spark framework as compared to Scikit learn with an accuracy of 95% and 88% respectively.

In [15], Shristi Priya, Ankit Kumar Gupta, Atman Dwivedi, Aryan Prabhakar believed that once upon a time even trying to predict a match winner for cricket was difficult but believed with the emergence of more machine learning models and greater availability of statistical information. So they set out to figure what model would have the hight accuracy in predicting the winning team. They used a Kaggle dataset, processed it and applied the RF, SVM, LR, k-NN, Decision Tree and NB models to conclude that Random Forest Classifier and Decision Tree Classifiers had the best accuracy with 74% and 73% respectively.

Reference,	Objectives	Problem Statement	Methodology	Dataset	Algorithm	Advantage	Disadvantage	Performance Measure
1 2016	ML for English County T20	Cricket Match Winner Prediction	Data Collection in separate databases, Data Pre-processing, Train & test splitting	Data from Cricinfo	NB, LR, RF, Gradient boosted decision tree	Model outperformed the bookmakers	Very old dataset used for model evaluation	Accuracy
2 2017	Use crowd opinion on social networks for prediction	Cricket Match Outcome Prediction	Tweets Collection, Feature representation, Classifier Training, Prediction Hypothesis, Evaluation	Tweets from twitter and some comments from popular Cricket websites	SVM, NB, LR	Able to outperform bookmaker's prediction	Accuracy was less than 75%	Accuracy

3 2018	Supervised Learning for prediction of IPL match winner	Cricket Match Winner Prediction	Data Collection, Data Pre-processing, Train & test splitting	Data from IPL's official website	NB, SVM, RF	More accuracy when combined	Not compared with other algorithms	Accuracy
4 2018	Using Multiclass SVM for increasing prediction accuracy in Cricket Prediction	Model for player selection in Cricket	Data Collection in separate databases, Data Pre-processing, Train & test splitting	Data from Cricinfo	NB, Decision Tree, RF, SVM	Better Performance	Inconsistent with some features	Precision, Recall, F1 Score, AUROC, RMSE
5 2018	Hybrid ML model for Cricket winner prediction for T20 Format	Cricket Match Winner Prediction	Pre-processing, Learning, Evaluation, Prediction	Data from Cricsheet	MLR, Ridge Regression, Lasso Regression, Neural Network, SVM, Decision Tree, RF, and MLP Classifier	Multi Stage Prediction Available	Unbalanced dataset used	Accuracy
6 2019	Use Data Analytics for Cricket match winner prediction	Cricket Match Winner Prediction	SEMMA (Sample, Explore, Modify, Model, and Assess)	IPL T20 match winner dataset	NB, Decision Tree, SVM, RF,XGBoost	Better accuracy than the previous models	Unlike other models RF was not able to generate higher accuracy than the other algorithms	Accuracy
7 2020	Supervised Learning for prediction of IPL match winner	Cricket Match Winner Prediction	Data Collection, Data Pre-processing, Train & test splitting	Data from Cricinfo	SVM, LR, Decision Tree, Bayes Classifier	Better accuracy than the previous model	Slow Computation	Accuracy
8 2020	Predict Match duration using multiple ML algorithms	Cricket Match Duration Prediction	Data collection, pre-processing, Training and Prediction	Data of IPL from Cricsheet	LR, LogR, SVM,SVD	Novel Approach	Small dataset as most of the features were dropped	RMSE
9 2020	T20 Cricket World Cup 2020 Winner Prediction	Cricket Match Winner Prediction	Data collection, data splitting, Training and Prediction	Data from ESPN Cricinfo	RF,ID3,C4.5, Extra Trees	Better Performance	Skipped many real world features like venue and weather forecast	Custom Accuracy, RMSE
10	Naive Bayes approach	Cricket Match Winner Prediction of ODI	Naive Bayes	Data of ODI matches between 2018-	LogR, NB	Better accuracy than the previous model for	Very small sample size	Accuracy

2020		format		2020 from ICC		ODI match format		
11 2020	Use ML for IPL 2020 winter prediction	Cricket Match Winner Prediction	Data collection, Feature extraction, training, evaluation	Data on IPL from ICC's T20 top 100 players and from Cricbuzz website	SGDRegressor , KNN- Regressor, LR	Better Accuracy	Model can only be applied to IPL format	Accuracy
12 2020	Create a website incorporating a ML based best team line-ups generator	Best team line-ups for Cricket	Data collection, Feature extraction, training, evaluation	Not mentioned but they are using web scraping for the data	RF, Decision Tree, Extra Tree	Website deployed for the model making navigation easier	Very less features and a small dataset was used	Accuracy
13 2021	Predictive Analysis of Cricket	Model for player selection in Cricket	Data Collection in separate databases, Data Pre-processing, Model creation, Decision for cricket council	Kaggle, a db from cricket crowd	Multinomial Logistic Regression, RF, AdaBoost	Novel Approach	Use of AdaBoost decreased the performance	Accuracy
14 2021	Use big data approach for prediction	Cricket Match Winner Prediction	Data collection, Pre-processing, Training, Model Selection based on evaluation	Data of ODI from Cricsheet	Linear Regression in Spark ML & Scikit Learn Framework	This approach can be applied to other sports	Model not fully automated. User has to give multiple manual inputs	Accuracy, RMSE, MAE
15 2022	Analysis and predict Cricket Match winner using multiple ML algorithms	Cricket Match Winner Prediction	Data Collection in separate databases, Data Pre-processing, Model creation	Data from Kaggle	RF, SVM, LR, k-NN, Decision Tree and NB	Multiple Algorithm tested	Poor data cleaning	Accuracy

III. PROBLEM STATEMENT

The problem statement for this project will be like the problem statement found in most of the papers reviewed in the literature survey section. "Cricket Match Winner Prediction using Machine Learning."

The objectives are:

- Make a model that can accurately predict the win/loss probability of a team
- Understand the diverse ways of data collection and pre-processing as the data for cricket is vastly available in multiple websites.
- Analyse the performance of the used model and compare the findings with those from other models.

IV. IMPLEMENTATION

After carefully searching for the fields we would need to make this predictor work as accurate as possible. We went on to look for the data we required in various cricket based websites like ESPN Cricinfo, CricBuzz, Official sites of the ICC, BCCI, IPL and so much more.

Then we found source named Cric Sheet where in we got data from different international matches played that also met the requirements we needed. So the dataset was

collected from cricsheet.org and was pre-processed before it can be used for training and testing the model. The dataset we selected from the website had a plethora of csv files of match data and deliveries that amounted up to 1701 T20 world cup matches of mens cricket. The match_id_info.csv contains data of each team playing with their team members, match venue, date, event, toss details, winner details(by runs/wickets)etc. Figure x shows the sample of one such file.

version	2.2.0		
info	balls_per_over		6
info	team	Tanzania	
info	team	Eswatini	
info	gender	male	
info	season	2022/23	
info	date		06-12-22
info	event	ICC Men's T20 World Cup Sub Regional Africa Qualifier	
info	match_number		17
info	venue	Gahanga International Cricket Stadium. Rwanda	
info	city	Kigali City	
info	toss_winner	Eswatini	
info	toss_decision	field	
info	player_of_match	AR Patwa	
info	umpire	S Kintu	
info	umpire	Stephen Harris	
info	reserve_umpire	C Phiri	
info	match_referee	SA Fritz	
info	winner	Tanzania	
info	winner_runs		66
info	player	Tanzania	AR Patwa
info	player	Tanzania	II Selemani
info	player	Tanzania	AP Rajeevan
info	player	Tanzania	S Thakor
info	player	Tanzania	K Nassoro
info	player	Tanzania	MO Kitunda
info	player	Tanzania	Akhil Anil
info	player	Tanzania	HA Chohan
info	player	Tanzania	SA Jumbe
info	player	Tanzania	AM Kimote
info	player	Tanzania	YM Nkanya

Figure 1 Sample data file 1343784_info.csv

This data was cleaned and formatted to make sure that only valid data is present and as we had .csv for each match all those csv filed were combined to create a single .csv containing relevant details of all 1701 matches.
The algorithm in Figure x was used to dump the cleaned and formatted data into their respective .csv files and excel was used to merge all the csv.

The match_id.csv files contained all the data related to each ball in the match. It contains the details of runs, wickets, batting team, bowling team, venue, season, batsman, bowler, non-striker, etc. This data was also cleaned, formatted and merged to create a delivery.csv file containing details of each ball for all 1701 matches.

```
res = []
# Iterate directory
for file in os.listdir('matches'):
    # check only text files
    if file.endswith('.csv'):
        res.append(file)
```

```
path = r'matches'
files=[os.path.join(path,x) for x in res]
len(files)
```

```
1701
```

```
newpath = r'folder'
for x in files:
    m1 = pd.read_csv(x, on_bad_lines='skip')
    m2 = m1.T
    m2.columns = m2.iloc[0]
    m = m2[1:]
    m.to_csv(os.path.join(newpath,x))
```

Figure 2 Algorithm for preprocessing match_id_info.csv

Figure 3 and 4 shows the sample of delivery.csv file

Source.Na	match_id	season	start_date	venue	innings	over	balls	batting_te	bowling_te	striker	non_strik
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	1	1	Australia	Sri Lanka	AJ Finch	M Klinger
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	1	2	Australia	Sri Lanka	AJ Finch	M Klinger
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	1	3	Australia	Sri Lanka	AJ Finch	M Klinger
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	1	4	Australia	Sri Lanka	M Klinger	AJ Finch
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	1	5	Australia	Sri Lanka	M Klinger	AJ Finch
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	1	6	Australia	Sri Lanka	M Klinger	AJ Finch
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	2	1	Australia	Sri Lanka	M Klinger	AJ Finch
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	2	2	Australia	Sri Lanka	M Klinger	AJ Finch
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	2	3	Australia	Sri Lanka	AJ Finch	M Klinger
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	2	4	Australia	Sri Lanka	AJ Finch	M Klinger
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	2	5	Australia	Sri Lanka	AJ Finch	M Klinger
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	2	6	Australia	Sri Lanka	AJ Finch	M Klinger
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	3	1	Australia	Sri Lanka	M Klinger	AJ Finch
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	3	2	Australia	Sri Lanka	AJ Finch	M Klinger
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	3	3	Australia	Sri Lanka	M Klinger	AJ Finch
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	3	4	Australia	Sri Lanka	M Klinger	AJ Finch
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	3	5	Australia	Sri Lanka	M Klinger	AJ Finch
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	3	6	Australia	Sri Lanka	M Klinger	AJ Finch
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	4	1	Australia	Sri Lanka	AJ Finch	M Klinger
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	4	2	Australia	Sri Lanka	AJ Finch	M Klinger
1001349.c	1001349	2016/17	17-02-17	Melbourne	1	4	3	Australia	Sri Lanka	M Klinger	AJ Finch

Figure 3 Sample of delivery.csv File Part 1

e bowler	runs_off_t	extras	wides	noballs	byes	legbyes	penalty	total_runs	wicket_typ	player_dis	ot
SL Malinga	0	0	0	0	0	0	0	0			
SL Malinga	0	0	0	0	0	0	0	0			
SL Malinga	1	0	0	0	0	0	0	1			
SL Malinga	2	0	0	0	0	0	0	2			
SL Malinga	0	0	0	0	0	0	0	0			
SL Malinga	3	0	0	0	0	0	0	3			
KMDN Kuli	0	0	0	0	0	0	0	0			
KMDN Kuli	1	0	0	0	0	0	0	1			
KMDN Kuli	0	0	0	0	0	0	0	0			
KMDN Kuli	0	0	0	0	0	0	0	0			
KMDN Kuli	4	0	0	0	0	0	0	4			
KMDN Kuli	2	0	0	0	0	0	0	2			
JRMVB Sar	1	0	0	0	0	0	0	1			
JRMVB Sar	1	0	0	0	0	0	0	1			
JRMVB Sar	0	0	0	0	0	0	0	0			
JRMVB Sar	0	0	0	0	0	0	0	0			
JRMVB Sar	4	0	0	0	0	0	0	4			
JRMVB Sar	0	0	0	0	0	0	0	0			
KMDN Kuli	0	0	0	0	0	0	0	0			
KMDN Kuli	1	0	0	0	0	0	0	1			
KMDN Kuli	1	0	0	0	0	0	0	1			

Figure 4 Sample of delivery.csv File Part 2

After collection and pre-processing of data. Pandas was used for analysing the data for all available features that can be used and it was also used to create new compound featured using the pre-existing ones. After creation of feature and data analysis, certain features were selected that showed significance in influencing the result of prediction and performance of the model.

The available data was then split into training and testing datasets using the train_test_split function of sklearn. We used one hot encoding to deal with the categorical features present in the final dataframe. Three prebuilt model in sklearn library were trained using the data with default parameters and the result were analysed to check which one of them gives the best accuracy.

V. METHODOLOGY WITH PROPOSED ARCHITECTURE DIAGRAM

To build and create a prediction model that is accurate in its prediction and efficient in the manner it functions we sought out to use a variety of machine learning algorithms to figure out the best possible option moving forward. Figure x shows the steps taken for building the model and figure x show the architecture diagram of the model built.

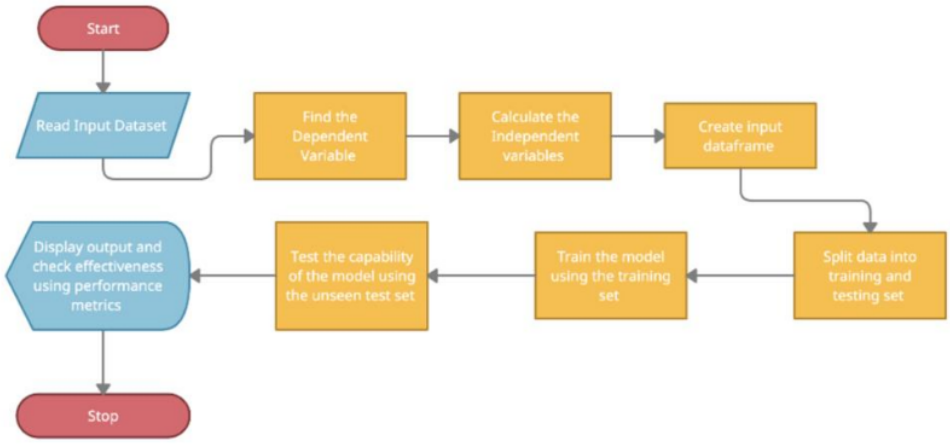

Figure 5 Methodology

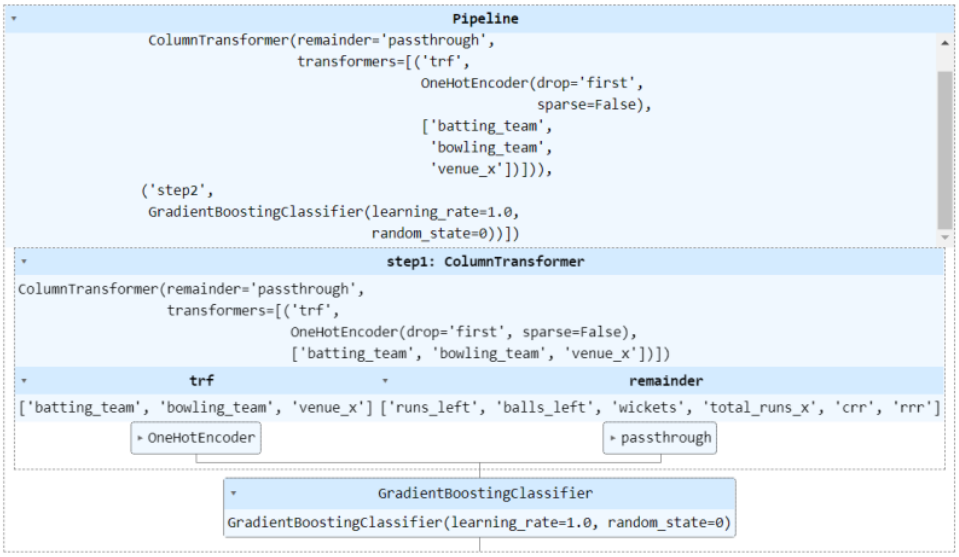

Figure 6 Architecture of Model

VI. ALGORITHMS

Gradient Boosting Classifier

The classifier believes that the amalgamation of the existing models along with the next best possible model can yield a lesser error rate in prediction and hence is useful to.

Logistic Regression

Usually used to predict a very straightforward outcomes, something as simple as a yes or no answer etc. We analyze indepent variables to figure out the outcome which is binary in nature.

Random Forest Classifier

A sub category of decision trees, wherein the various data is categorizes into further branches. The Random Forest Classifier just fits the new data under or into on of the nearest trees there.

KNeighbors Classifier

As the name suggests this classifier gets trained by the datasets its fed and then for future examples it uses the information it has to find the k closest neighbors.

AdaBoost Classifier

AdaBoost is a classic solution used in strengthening weak solutions and situations into stronger more accurate classes.

VII. Result and Discussion

As seen in the section above, the various algorithms used have yielded different scores which are as follows.

AdaBoostClassifier

	precision	recall	f1-score	support
0	0.83	0.85	0.84	19631
1	0.82	0.80	0.81	16814
accuracy			0.82	36445
macro avg	0.82	0.82	0.82	36445
weighted avg	0.82	0.82	0.82	36445

confusion_matrix
[[16611 3020]
[3381 13433]]

LogisticRegression

	precision	recall	f1-score	support
0	0.89	0.89	0.89	19631
1	0.88	0.88	0.88	16814
accuracy			0.89	36445
macro avg	0.89	0.88	0.88	36445
weighted avg	0.89	0.89	0.89	36445

confusion_matrix
[[17565 2066]
[2100 14714]]

RandomForestClassifier

	precision	recall	f1-score	support
0	0.82	0.86	0.84	19631
1	0.82	0.78	0.80	16814
accuracy			0.82	36445
macro avg	0.82	0.82	0.82	36445
weighted avg	0.82	0.82	0.82	36445

confusion_matrix
[[16788 2843]
[3669 13145]]

KNeighborsClassifier

	precision	recall	f1-score	support
0	0.94	0.95	0.95	19631
1	0.94	0.93	0.94	16814
accuracy			0.94	36445
macro avg	0.94	0.94	0.94	36445
weighted avg	0.94	0.94	0.94	36445

confusion_matrix
[[18675 956]
[1123 15691]]

GradientBoostingClassifier

	precision	recall	f1-score	support
0	0.96	0.96	0.96	19631
1	0.95	0.95	0.95	16814
accuracy			0.96	36445
macro avg	0.96	0.96	0.96	36445
weighted avg	0.96	0.96	0.96	36445

confusion_matrix
[[18876 755]
[818 15996]]

VIII. Conclusion

This opportunity presented to us to apply the concepts we have learned in the classroom has been very enlightening, as the real-life application of said concepts have given us a greater understanding of machine learning.

We set out to apply our learnings in predicting the outcome of cricket matches. Our purpose of doing so was to assist coaches, teams and players around the world with better understanding their game and hopefully assist them in improving so much that they can convert games into more wins. Our prediction model made use of the dataset we got from online sources and utilized information about the players, toss, venue, dismissals and much more to train our models. We went an extra step to make sure our users got the best model possible in terms of accuracy, so we tried 5 different models and as inferred from the results, one can note that out of the five algorithms we have used, they have returned the corresponding accuracies when run (Gradient Boosting Classifier - 96%),(Logistic Regression - 89%),(Random Forest Classifier - 82%), (KNeighbors Classifier - 94%),(AdaBoost Classifier - 84%) With accuracies as great as 96% and 94%, we are confident that our model can be state of the art technology that can be used by great cricketing nations to optimize their games even more.

REFERENCES

[1] Suvobrata Sarkar, An Improved Rough Set Data Model for Stock Market Prediction,2014, 2nd International Conference on Business and Information Management (ICBIM).

[2] Reza Ramezanian, Arsalan Peymanfar, Seyed Babak Ebrahimi, An integrated framework of genetic network programming and multi-layer perceptron neural network for prediction of daily stock return: An application in Tehran stock exchange market,Applied Soft Computing,Volume 82,2019,105551,ISSN 1568-4946

[3] Ryo Akita,Akira Yoshihara,Takashi Matsubara,Kuniaki Uehara,Deep Learning for Stock Prediction using Numerical and Textual Information,ICIS 2016, June 26-29, 2016, Okayama, Japan

[4] Ehsan Hoseinzade, Saman Haratizadeh, CNNpred: CNN-based stock market prediction using a diverse set of variables, Expert Systems with Applications, Volume 129, 2019, Pages 273-285, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2019.03.029.

[5] Omer Berat Sezer, Ahmet Murat Ozbayoglu, Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach, Applied Soft Computing, Volume 70, 2018, Pages 525-538, ISSN 1568-4946, https://doi.org/10.1016/j.asoc.2018.04.024.

[6] Jiawei Long, Zhaopeng Chen, Weibing He, Taiyu Wu, Jiangtao Ren, An integrated framework of deep learning and knowledge graph for prediction of stock price trend: An application in Chinese stock exchange market, Applied Soft Computing, Volume 91, 2020, 106205, ISSN 1568-4946, https://doi.org/10.1016/j.asoc.2020.106205.

[7] B. B. Nair, N. M. Dharini and V. P. Mohandas, "A Stock Market Trend Prediction System Using a Hybrid Decision Tree-Neuro-Fuzzy System," 2010 International Conference on Advances in Recent Technologies in Communication and Computing, 2010, pp. 381-385, doi: 10.1109/ARTCom.2010.75.

[8] Lei Wang,Qiang Wang,Stock Market prediction using artificial neural networks based on HLP, 2011 Third International Conference on Intelligent Human-Machine Systems and Cybernetics.

[9] Shanoli Samui Pal, Samarjit Kar, Time series forecasting for stock market prediction through data discretization by fuzzistics and rule generation by rough set theory, Mathematics and Computers in Simulation, Volume 162, 2019, Pages 18-30, ISSN 0378- 4754, https://doi.org/10.1016/j.matcom.2019.01.001.

[10] Kimoto, T., Kazuo Asakawa, Morio Yoda and Masakazu Takeoka. "Stock market prediction system with modular neural networks." 1990 IJCNN International Joint Conference on Neural Networks (1990): 1-6 vol.1.

[11] Rohit Verma,PROF. Pkumar Choure,Upendra Singh,Neural Networks through Stock Market Data Prediction,International Conference on Electronics, Communication and Aerospace Technology ICECA 2017

[12] S. Kadam, S. Jain, "Stock Market Prediction Using Machine Learning", (IJIRT), ISSN:2349-6002, Volume-8 Issue-2, July 2022

[13] S. S. Maini and K. Govinda, "Stock market prediction using data mining techniques," 2017 International Conference on Intelligent Sustainable Systems (ICISS), 2017, pp. 654-661, doi: 10.1109/ISS1.2017.8389253.

[14] Sunil Kumar Khatri,Ayush Srivastava,Using Sentimental Analysis in Prediction of Stock Market Investment,2016 5th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), Sep. 7-9, 2016.

[15] Sergio Garcia-Vega,Xiao-Jun Zeng, John Keane,Stock returns prediction using kernel adaptive filtering within a stock market interdependence approach,Expert Systems with Applications,Volume 160, 1 December 2020, 113668

[16] K. Chen, Y. Zhou and F. Dai, "A LSTM-based method for stock returns prediction: A case study of China stock market," 2015 IEEE International Conference on Big Data (Big Data), 2015, pp. 2823-2824, doi: 10.1109/BigData.2015.7364089.

[17] Dharmaraja S, Vineet K, Abhishek M. Indian stock market prediction using artificial neural networks on tick data. Selvamuthu et al. Financial Innovation (2019) 5-16.

[18] Raut S, Shinde I, D. Malathi. International Journal of Pure and Applied Mathematics. Volume 115 No. 8 2017, 71-77. ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (online version) url: http://www.ijpam.eu Special Issue.

[19] M. Nabipour, P. Nayyeri, H. Jabani, A.Mosavi, E. Salwana, Shahab S. (2020) Deep Learning for Stock Market Prediction.

[20] Kranthi R. Stock Market Prediction Using Machine Learning. International Research Journal of Engineering and Technology (IRJET) Volume: 05 Issue: 10 | Oct 2018 e-ISSN: 2395-0056 p-ISSN: 2395-0

ML mini project phase 2

ORIGINALITY REPORT

6%	4%	3%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.irjmets.com Internet Source	1%
2	www.mdpi.com Internet Source	1%
3	assets.researchsquare.com Internet Source	1%
4	Submitted to BITS, Pilani-Dubai Student Paper	1%
5	"Table of Contents", 2020 IEEE 23rd International Multitopic Conference (INMIC), 2020 Publication	<1%
6	Submitted to UNITEC Institute of Technology Student Paper	<1%
7	Submitted to University of Westminster Student Paper	<1%
8	scholarworks.sjsu.edu Internet Source	<1%
	Submitted to Brisbane Catholic Education	

9	Student Paper	<1%
10	www.mitmoradabad.edu.in Internet Source	<1%
11	"Third International Conference on Image Processing and Capsule Networks", Springer Science and Business Media LLC, 2022 Publication	<1%
12	Fatimah Alzubaidi, Peyman Mostaghimi, Guangyao Si, Pawel Swietojanski, Ryan T. Armstrong. "Automated Rock Quality Designation Using Convolutional Neural Networks", Rock Mechanics and Rock Engineering, 2022 Publication	<1%
13	dokumen.pub Internet Source	<1%
14	www.proceedings.com Internet Source	<1%
15	www.ijert.org Internet Source	<1%
16	"Advances in Artificial Intelligence and Data Engineering", Springer Science and Business Media LLC, 2021 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		