A Technique to Predict Indian Premier League Match Winner using Artificial Intelligence

Ajmain Inqiad Alam
BRAC University
2 PUBLICATIONS **9** CITATIONS
SEE PROFILE

Abdullah Umar Nasib
BRAC University
5 PUBLICATIONS **16** CITATIONS
SEE PROFILE

A Technique to Predict Indian Premier League Match Winner using Artificial Intelligence

Abdullah Umar Nasib, Ajmain Inqiad Alam, and Mahfuzur Rahman

Abstract—Many researches have been made to predict the first innings run in an ODI cricket match as prediction is an influential thing for the team management and their economic outcome. However, in premier leagues like IPL, PSL, BIG BASH this little calculation is tremendously useful in helping the franchise and their owners in terms of their business. In this paper, we have predict the winner team of every match in Indian Premier League (IPL) based on the previous performance of players, toss winner, match venue and the city of the venue. We used decision tree model to predict the winner using the available dataset of previous eight seasons of IPL. The used dataset covers all the match by match and ball by ball details starting from 2008 to 2017. The prediction accuracy of this model was 89.844% where using other algorithms, we achieved poor accuracy rate. We came up with the conclusion that significant facts to decide the winner prediction of an IPL match are 'team1', 'toss_decision', 'team2', 'win_by_runs', 'city', 'id', 'toss_winner', 'venue', 'win_by_wickets', 'result', 'dl_applied'. Based on these analyses, our proposed model will determine the winner of every single match.

Index Terms—Indian Premier League, winner prediction, performance analysis, decision tree, classifier analysis;

1 Introduction

CRICKET is a standout amongst the most played and mainstream amusement today on the planet. The game has billion of supporters and number of people has been involved with it [5]. Indian subcontinent including India, Pakistan, Srilanka, Bangladesh, Australian arena, and Africa has a tremendous craze for the game. This outdoor game has been played in open field where ball and bat is involved. The game has formats like ODI (One Day International), Test, T20, 6 a side. Apart from Test matches, in all other format, each team gets to bat once and so does ball. However, in test matches, per team get to bat and ball twice.

The game of Cricket is a play of 11 players team where bat and ball is involved in 22 yard pitch. Batsman tries to hit the ball and score as much as possible where the bowler tries not to give away marks and turning the batsman out following the rules. Each batsman gets to play balls unless gets out. Therefore, if all of them are not out, the team gets to play the full overs which is varies from match format.

T20 is the youngest and most interesting format of cricket. Per team gets to play bat 20 overs where every six legal deliveries count as an over. This format of cricket is less time consuming comparing to ODIs and Test matches.

- *Abdullah Umar Nasib, Dept. of Computer Science & Engineering, BRAC University, Bangladesh, E-mail: umarnasib13@gmail.com*
- *Ajmain Inqiad Alam, Dept. of Computer Science & Engineering, BRAC University, Bangladesh, E-mail: ajmaininqiadalam@gmail.com*
- *Mahfuzur Rahman, Dept. of Computer Science & Engineering, BRAC University, Bangladesh, E-mail: mrasif30@gmail.com*

Manuscript received September 29, 2018; accepted Jan 2, 2019. (Corresponding author: Abdullah Umar Nasib.)

Normally this game ends within 3 and half an hours including the interval time where each inning takes around 1 and half an hours to finish with an over-rate per-hour of 14.1 according to ICC Law 16.2. Being a shorter duration game, people of diverse sectors can enjoy this game. This version of cricket has a tremendous impact on some countries economy today. India, Australia, Pakistan and many more countries organize their premier league T20s and add a huge amount to their government revenue.

Premier leagues like IPL, Big Bash, PSL and CPL impose a serious impact on the spectators of cricket as well as contribute a healthy amount of revenue in the government fund of the organizing countries. On that note, Indian Premier League (IPL) is the most-attended cricket leagues in the world. The brand value of IPL in 2017 was US$5.3 billion where according to BCCI, the contribution to the GDP of Indian economy in 2015 was US$182 million by the season of 2015[8]. The prize money of IPL is US$2.3 million where half of the winning prize money is distributed among the payer by rules and rest of the amount goes to the owner of the team. So, win of a season matter a lot to the managements.

In this paper, a method has been demonstrated to predict the winner of a match right after the toss has taken place [1]. In our approach, we have not solve any problem rather implementing an effective system that can predict the match winner in T20 cricket match providing previous data where as proposed model does the job with an accuracy of over 89.884 percentage. We have used a rich dataset containing ball by ball data and match by match data to predict the winner [2]. Before every match we just need to know the humidity, city the match taking place, which team won the toss, decision they have taken after winning the toss. Hence, having the information we analyze our dataset with the proposed method and the method will let me show the predicting winner of the match. In the dataset, we have got all the details of the ten previous seasons starting from 2008 to 2017.

In this paper, we have followed the following format: At the immediate section after this, we have briefly discussed on the previous researches in the very field to predict the winner. Where in section III, we focused on the proposed model and used algorithms with complete workflow. Hence, under IV which is Training and Testing the model has been talked about where section V is to focus on the comparison and analysis of our proposed model with existing researches. Following that, the conclusion section is been shown in IV no section and the paper concludes with the references we have used in the paper and research work.

2 Related Works

Only a very few researches has been made predicting the winner of a match in cricket. However, some of the works are well known to predict the total score of an innings. One of them is Winning and Score Predicting (WASP), which is a PhD research project by B. Scott and H. Seamus of University of Canterbury [6]. The work evaluates about good a general batting team will do against a general bowling alley group under provided circumstances and the current amusement condition. In the 1st innings it guess the extra runs that a team can score with the given number of balls and remaining wickets. In next innings it predicts the triumphant likelihood with the given number of balls and wickets remaining, runs scored at the unchanged conditions. The assessments have been produced using a dynamic programming, another work from Thapar University, India by S. Tejinder, S. Vishal and B. Parteek to predict the winner and the score using data mining entitled Score and Winning Prediction in Cricket through Data Mining is a mentionable research in the very field. In that paper they proposed a has two strategies, where initially predicts the 1st innings score not just based on present run rate yet additionally counts the number of wickets fallen, scenario of the match and batting team [7]. The 2nd technique forecast the result of the match in the second innings observing indistinguishable qualities from of the previous strategy alongside the objective set for the batting team. Linear Regression Classifier and Nave Bayes Classifier has been used to implement the mentioned two methodologies for first and second innings respectively. Apart from these, few more researches has been made on this field.

3 PROPOSED MODEL

This section presents the detail descriptions of proposed model which contains several phases. Figure 1 represents the whole working procedure of our proposed model.

In the proposed technique we have divided the whole system in two separate part called train and test where the first one to train the code and the second one to check the accuracy of the designed model. Firstly, we had to choose the suitable algorithm for our technique and we choose decision tree method. A decision tree is one of the most successful techniques for supervised classification learning.

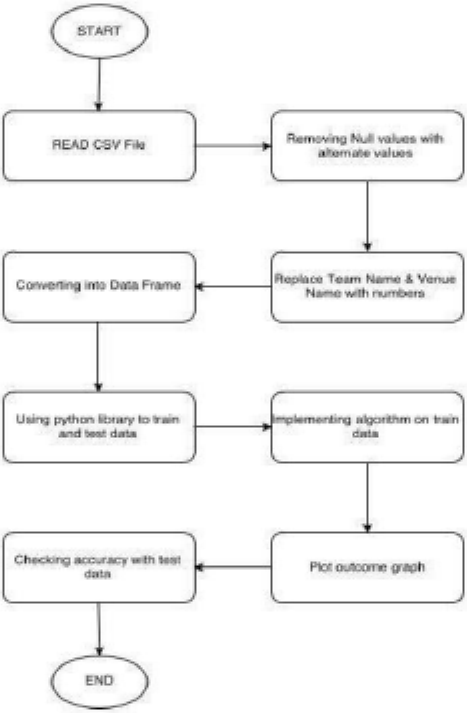
START
READ CSV File
Removing Null values with alternate values
Replace Team Name & Venue Name with numbers
Converting into Data Frame
Using python library to train and test data
Implementing algorithm on train data
Plot outcome graph
Checking accuracy with test data
END

Fig. 1: Proposed Working Architecture

It is also called a classification tree. In this tree each internal node is labelled by an input feature. The arcs of a node labeled with a feature are marked by each of the possible numbers of the provided information. Every leaf node of the tree is denoted by a class or an approximate prediction of the possible values. In the related papers, the researchers have tried to predict using regression method whereas we have proposed this model using classification method. We have taken our dataset shown in Figure 2 from kaggle.com, a well-known online resource for dataset. Our experimental dataset consists of two files. One file has the ball by ball details of previous 10 seasons IPL matches. The second one consists of the detail data of match by match information. To train our system, we have written dataset readable python code. Before using the dataset, the null values has checked and replaced by identical numerical numbers. Otherwise, it would decrease the accuracy rate of the prediction when testing. After removing the null values, the data has modified and the team names were also replaced by few more identical numbers as followings: Mumbai Indians: 1 ,'Kolkata Knight Riders': 2 ,'Royal Challengers Bangalore': 3, 'Deccan Chargers': 4, 'Chennai Super Kings': 5, 'Rajasthan Royals': 6, 'Delhi Daredevils': 7, 'Gujarat Lions': 8, 'Kings XI Punjab': 9, 'Sunrises Hyderabad': 10, 'Rising Pune Supergiant': 11, 'Kochi Tuskers Kerala': 12, 'Pune Warriors': 13. Whereas the venue names are also replaced in the following identical numerical numbers to simplify the process: 'Barabati Stadium': 1, 'Brabourne Stadium':2, 'De Beers Diamond Oval':3, 'Buffalo Park':4, 'Dr DY Patil Sports Academy':5, 'Visakhapatnam ACA-VDCA Cricket Stadium':6 and so on.

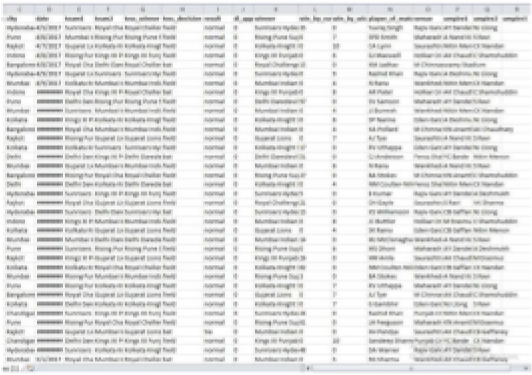

Fig. 2: Dataset contains match by match details

City names has also replaced with numbers: 'Abu Dhabi':1, 'Ahmedabad':2, 'Bangalore':3, 'Bloemfontein':4, 'Cape Town':5, 'Centurion':6 etc. We also have replaced toss_decision columns value: bat:1, field:2. The values of result are also replaced with numbers: 'no result':1, 'normal':2 'tie':3. The features we have used for this model are city, date, team1, team2, toss_winner, toss_decision, result, dl_applied, venue and the target attribute winner. The data has been converted in data frame in this segment to access in details using our python program. As already shown in Figure 1, the dataset was divided into two portion called train and test, we did the separation after the data frame was prepared. From the dataset, data of first 9 seasons, means from 2008 to 2016, was used to train the system [12]. We implemented decision tree learning algorithm on the separated dataset. Decision tree learning algorithm works by dividing the training set continuously in order to gain sub-functions that are equally correct to the provided class [9]. Every single leaf of the tree is directly connected to the specific dataset T which is splatted via an associated test on a characteristic.

3.1 Mathematical Formulation

Provided training vectors $x_i \in R^n$, i=1,..., l and a vector ,a classification tree repeatedly divides the space in such a way that the examples with the same labels are categorized together. Assuming the data at m node be represented by Q. For individual participant split θ = (j, tm) containing a character J and threshold t m , divide the data into $Q_{left}(\theta)$ and $Q_{right}(\theta)$ sublevels. The noise at m is calculated using a impurity set H(), the choice of which relies on the problem solving

$$G(Q, \theta) = \frac{n_{left}}{N_m} H(Q_{left}(\theta)) + \frac{n_{right}}{N_m} H(Q_{right}(\theta))$$

Increasing the information gain by selection of the parameters,

$$\theta^* = \text{argmin}_\theta\, G(Q, \theta)$$

Repeat for sub-functions $Q_{left}(\theta^*)$ and $Q_right(\theta^*)$ until the highest acceptable depth is obtained, $N_m < min_{samples}$ or $N_m = 1$.

3.2 Classification Criteria

If a desired value is a classification result calculating based on values starting from 0, 1,..., K-1, for leaves, that represents an area Rm with Nm times lookup [9], assume

$$p_{mk} = 1/N_m \sum_{x_i \in R_m} I(y_i = k) \tag{1}$$

Is the ratio of class k times lookup in leaf m Mutual observations of impurity are Gini [10]

$$H(X_m) = \sum_k p_{mk}(1 - p_{mk}) \tag{2}$$

Cross-Entropy [11]

$$H(X_m) = -\sum_k p_{mk} \log(p_{mk}) \tag{3}$$

And Wrong-classification [11]

$$H(X_m) = 1 - \max(p_{mk}) \tag{4}$$

This is how the decision tree learning algorithm mathematically performs in the library of Python Programming. For conducting our research, first of all we have extracted the features from dataset and calculated the entropy of the whole dataset. Then, we have calculated the information gain for each of the features and found out the best information gain. Depending on that gain, we have generated a tree and deleted the best information gain feature from the features array. After that we have repeated the steps till all the features have been checked.

Support vector machine (SVM) supports both dense and sparse sample vectors as input. SVM is capable of perform multi classification on a dataset. Provided vectors of training datasets, i=1... n, in multiple classes, and a vector, Support Vector Classification solves the below basic complexity [12]:

$$\begin{aligned} &\min_{w,b,\zeta} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \zeta_i \\ &\qquad \text{subject to } y_i(w^T \phi(x_i) + b) \geq 1 - \zeta_i, \\ &\qquad\qquad \zeta_i \geq 0, i = 1, ..., n \end{aligned} \tag{5}$$

Its another part is [12]

$$\begin{aligned} &\min_{\alpha} \frac{1}{2} \alpha^T Q \alpha - e^T \alpha \\ &\qquad \text{subject to } y^T \alpha = 0 \\ &\qquad\qquad 0 \leq \alpha_i \leq C, i = 1, ..., n \end{aligned} \tag{6}$$

Here e represents the set of 1s, upper limit is $C > 0$, Q is a non-negative semi definite matrix of n by n size. Here training sets are manually framed into a upper (probably immeasurable) dimensional space by the set [13]. For our research purpose, we have calculated threshold of the dataset and split the dataset based on that threshold. After that we have calculated the accuracy.

In random forests every tree within the ensemble is made from a sample drawn with replacement from the training set. Additionally, once rending a node throughout the development of the tree, the split that's chosen isn't any longer the most effective split among all options [14]. Instead, the split that's picked is that the best split among a random set of the options. As a results of this randomness, the bias of the forest sometimes slightly will increase however, thanks to averaging, its variance additionally decreases, sometimes quite compensating for the rise in bias, thence yielding an overall higher model [14]. A multilayer perceptron classifier trains iteratively since at each time step the partial derivatives of the loss function with respect to the model parameters are computed to update the parameters [15].

4 EXPERIMENTAL SETUP AND RESULT ANALYSIS

After completing the preprocessing and the training part we did the experimental setup and result analysis. In this very part, we followed the following stages which are testing the system and following that, representing the result graphically and finally analyzing the achieved result. All of them are being discussed below:

4.1 Testing the System

We completed the testing segment of our model in this part. With the rest of the data of 2017 IPL matches, we ran our project and tested our proposed model.

Based on the pre- set parameters, our proposed model took decision on the above mentioned algorithm and gave us the output of the winner of every single match. We have experimented taking two teams as opponents and provided by the information of the parameters, the system, have shown us the output in graphical representation format.

4.2 Result Representation

The decision tree learning algorithm gives us the output in form of winner of a match in graphical representation as displayed in Figure 3.

In Figure 3 below, in y axis: teams name converted into numbers as mentioned earlier 'Mumbai Indians':1,'Kolkata Knight Riders': 2 ,'Royal Challengers Bangalore': 3, 'Deccan Chargers': 4, 'Chennai Super Kings': 5, 'Rajasthan Royals': 6, 'Delhi Daredevils': 7, 'Gujarat Lions': 8, 'Kings XI Punjab': 9, 'Sunrises Hyderabad': 10, 'Rising Pune Supergiant': 11, 'Kochi Tuskers Kerala': 12, 'Pune Warriors': 13. Hence, if prediction shows winner team 1 this means Mumbai Indians will win the match. However x axis represents match number.

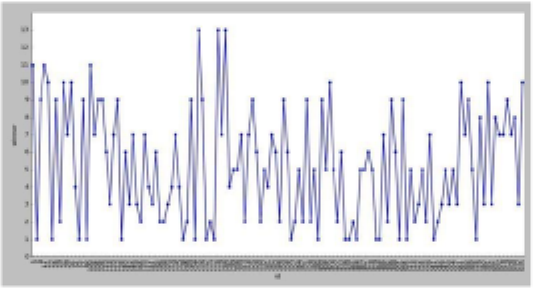

Fig. 3: Graphical representation of prediction using Decision Tree Classifier

In Figure 3, id 3 means, match no 3. This is how our program gives the output of the predicted winner of every single IPL match.

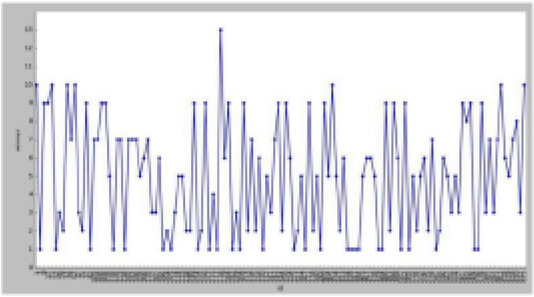

Fig. 4: Graphical representation of prediction using Random Forest Classifier

Random forest classifier algorithm gives name of every match winner which is shown in Figure 4. The match id and the team names represent the axis of this graph.

In this Figure 4, in y axis, we have team names converted into number as previously discussed and in x axis, we have match id which indicated the match no.

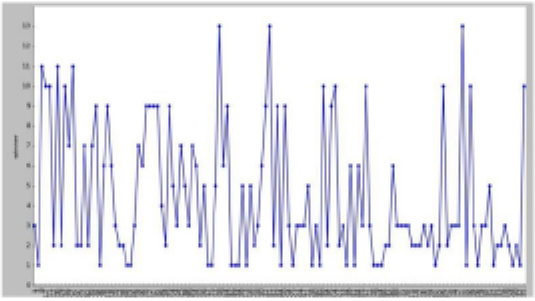

Fig. 5: Graphical representation of prediction using MLP Classifier

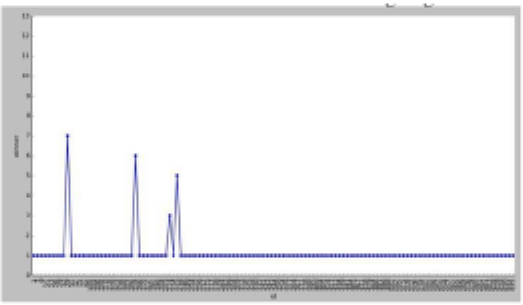

Fig. 6: Graphical representation of prediction using SVM Classifier

In Figure 5, team names in form of numbers has shown in y axis and match no is shown in x axis. This Figure 5 tells that on that match which team is going to win.

In Figure 6, team names in form of numbers has shown in y axis and match no is shown in x axis. This Figure 6 tells on that match which team is going to win.

TABLE 1: Algorithm with Accuracy

Serial	Algorithm	Accuracy
01	SVM classifier	17.969%
02	Decision Tree Classifier	89.844%
03	Random Forest Classifier	65.625%
04	MLP Classifier	20.312%

4.3 Result Analysis

The proposed model gives us the accuracy of the match winner of 89.884% using decision tree learning algorithm. However, we also have run few more approach with different algorithms but didnt get an expected or better than this accuracy rate. The other algorithms we experienced apart from decision tree learning are SVM classifier, Random Forest Classifier and MLP Classifier. The experimented algorithms are being noted below:

We could not choose Random forest because the bias of the forest is increased due to the randomness. For SVM and MLP classifier, data noise is too high to calculate the result accurately. Therefore we came to the conclusion that for our proposed model, decision tree approach is the most suitable and applicable method to be chosen.

5 Conclusions

The main purpose of this paper is to demonstrate our proposed model to predict the winner of an Indian Premier League match analyzing the previous ten season match dataset. In this model, decision tree classify method has been used to analysis the data and after the derivation, predict the winner of every single match. This model has some pre-set parameters to decision making criteria fulfilling. Not likely the existing models that can predict the scores of an innings or guess the result of an ODI match, our model can predict the winner of Indian Premier League T20 match winner with an accuracy of 89.844 percent. Since, with the available dataset of IPL; this analysis prediction was made, its more likely to be possible to predict the winner of all format of games in any tournament on the condition of having a rich dataset [12]. As the proposed model works based on the previous record, the proposed model is not completely capable of predicting a teams' performance if it's new. However, considering some other factors like venue, weather, day/night etc. unchanged, partially this method will be able to predict so.

Furthermore, there are other parameters like ranking of the players, performance of bowlers against left or right arm batsman, performance of batman against the off or leg break bowlers which is not available till today. In terms of international cricket tournaments like ICC World Cup, Champions Trophy, we also can implement this approach as we have a further feature available for dataset, the ranking of the team and the player ranking with the rating of each player. With that dataset this prediction rate accuracy would be more prestige and trustworthy. Similar to cricket, we can even try this proposed approach in other competitive sports tournament like Football matches winner prediction with available dataset.

References

[1] Dzone.com. (2018). Predicting the Outcome of Cricket Matches Using AI - DZone AI. [online] Available at: https://dzone.com/articles/ipl-cricket-analytics-and-predictive-model [Accessed 26 Mar. 2018].

[2] Spin, D. (2018). How I Used Machine Learning To Predict Soccer Games For 24 Months Straight. [online] Doctor Spin. Available at: https://doctorspin.me/digital-strategy/machine-learning/ [Accessed 26 Mar. 2018].

[3] Medium. (2018). Betting: Football Chat now with AI predictor Football Score Chat Medium. [online] Available at: https://medium.com/@ScoreChat/betting-football-chat-now- with-ai-predictor-f1f4b922d4d0 [Accessed 26 Mar.2018].

[4] Nyquist, R. and Pettersson, D. (2017). Football match prediction using deep learning. Available at: http://studentarbeten.chalmers.se/publication/250411-football-match-prediction-using-deep-learning [Accessed 26 Mar. 2018]

[5] Khabir Uddin Mughal. Top 10 Most Popular Sports In The World. http://sporteology.com/top-10-popular-sports-world/ [Accessed 2 Feb 2018.].

[6] I. Bhandari, E. Colet, and J. Parker. Advanced Scout:Data mining and knowledge discovery in NBA data. Data Mining and Knowledge Discovery, 1(1):121125,1997.

[7] S. Tejinder, S. Vishal, B. Parteek. Score and Winning Prediction in Cricket through Data Mining. 2015 International Conference on Soft Computing Techniques and Implementations- (ICSCTI) Department of ECE, FET, MRIU, Faridabad, India, Oct 8-10, 2015.

[8] Thenewsminute.com. (2018). [online] Available at: https://www.thenewsminute.com/article/kerala-beat-west-bengal-dramatic-shootout-win-santosh-trophy-6th-time-78842 [Accessed 1 Apr. 2018].

[9] L. Breiman, J. Friedman, R. Olshen, and C. Stone. Classification and Regression Trees. Wadsworth, Belmont, CA, 1984.

[10] J.R. Quinlan. C4. 5: programs for machine learning. Morgan Kaufmann, 1993

[11] T. Hastie, R. Tibshirani and J. Friedman. Elements of Statistical Learning, Springer, 2009.

[12] I. Guyon, B. Boser, and V. Vapnik, Automatic Capacity Tuning of Very Large VC-dimension Classifiers, Advances in Neural Information Processing Systems, pp. 147155, 1993.

[13] C. Cortes and V. Vapnik, Support-vector networks, Machine Learning, vol. 20, no. 3, pp. 273297, 1995.

[14] L. Breiman, Random Forests, Machine Learning, 45(1), 5-32, 2001.

[15] G. E. Hinton, Connectionist learning procedures, Artificial Intelligence, vol. 40, no. 1-3, pp. 185234, 1989.

[16] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Learning representations by back-propagating errors, Nature, vol. 323, no. 6088, pp. 533536, 1986.

[17] H. Saikia, D. Bhattacharjee, H. H. Lemmer, Predicting the performance of bowlers in ipl: an application of artificial neural network, International Journal of Performance Analysis in Sport 12 (1) (2012) 75-89

[18] R. Lamsal, A. Choudhary, Predicting outcome of indian premier league (ipl) matches using classification based machine learning algorithm, arXiv preprint arXiv:1809.09813.

[19] R. U. Mustafa, M. S. Nawaz, M. I. U. Lali, T. Zia, W. Mehmood, Predicting the cricket match outcome using crowd opinions on social networks: a comparative study of machine learning methods, Malaysian Journal of Computer Science 30 (1) (2017) 63-76.

[20] A. Naik, Winning prediction analysis in one-day-international (odi) cricket using machine learning techniques, International Journal Of Emerging Technology and Computer Science 3 (2) (2018) 94-100