



Crick11pro – Dream11 Prediction Along with Chat-Bot

Dr. M. V. Shelke¹, Kasturi Joshi², Sahil Khambad³, Amit Birmal⁴, Shivtej Ghosalkar⁵

¹Assistant Professor, AISSMS Institute of Information Technology, Pune, India.

^{2,3,4,5}AISSMS Institute of Information Technology, Pune, India.

Emails: mayura.shelke@aissmsioit.org¹, kasturijoshi2805@gmail.com², sahilkhambad2003@gmail.com³, amitbirmal@gmail.com⁴, shivtejghosalkar@gmail.com⁵

Article history

Received: 07 May 2024

Accepted: 29 May 2024

Published: 06 June 2024

Keywords:

Chatbot, Decision Making, Fantasy Sports, Gradient Boosting, Predictive Model

Abstract

Dream 11, a popular fantasy sports platform, gives cricket enthusiasts a chance to select a dream team of players and earn rewards. The proposed paper presents a system that would assist users in preparing dream teams for cricket matches. The proposed system introduced a model designed to help the user in selecting ideal teams by applying machine learning techniques. The machine learning model combined cricket data, head-to-head player statistics, and venue-wise data, and then assigned scores to all players from both teams. The system mainly assigned higher weights to the players from the predicted winning team but also considered the performance of key players from the opposing team. The chatbot interface allowed continuous interaction with the user to allow them to gain information about cricket. The system provided an evaluation of the methodology used, and the use of the predictive model, which resulted in displaying player scores for decision-making.

1. Introduction

Cricket is an engaging sport all over the world. Dream11, a fantasy sports platform developed that allowed cricket enthusiasts to create dream teams and win different rewards. The proposed system addressed the audience by introducing a solution that combined prediction skills with a chatbot interface. The necessity for the proposed system arrived from the rising interest in cricket predictions, and the growing demand for data-related decision-making. As of the latest update, no system offered a combination of Dream11 prediction and chatbot functionalities. The idea of combining prediction with a chatbot interface gave a new approach to the domain. The technologies for prediction till now mainly depended on player information which was not sufficient for accurate predictions. To increase the accuracy of prediction

features such as head-to-head player data, venue-wise player performance, pitch conditions, and weather data needed to be included. This helped to provide the users with more accurate player recommendations for fantasy teams. The paper by Nilesh M. Patil et al. (2020) [6] introduced a cricket team winning prediction model, that used the Random Forest Algorithm and Decision Tree classifiers to predict the optimal playing eleven based on the player's previous performance. It integrated batting, bowling, and overall abilities to remove bias in player selection, which offered an understanding of individual and team performance using virtual simulations. Ayush Tripathi et al. (2020) [2] used feature engineering, that mainly focused on the Indian Premier League (IPL). By using machine learning techniques like neural

networks and support vector machines, it assisted in the prediction of match outcomes, and bowler performance evaluation. Saurav Singla et al. (2020) [13] used a Memetic Genetic Algorithm for player selection. It used an integer optimization method based on the player's previous ten matches' performance and suggested enhancements that included head-to-head data and venue-specific performance. Rohit Tamrakar et al. (2021) [10] paper introduced a CHATBOT. This chatbot used Artificial Intelligence and machine learning and provided practical examples and further, helped by proposing how it would be used in CAD applications. Sachin Kumar S et al. (2022) [12] developed Fantasy Sports Platforms, which focused on data science and analytics for decision-making, and combined Greedy and Knapsack algorithms for team selection and utilizing PyCaret and Plotly Python libraries for prediction and visualization. Based on the previous work [14], analysis of venues, team fitness, and other factors such as weather and pitch conditions significantly improved prediction accuracy. This was accompanied by considering various classifiers and machine learning algorithms, with a detailed search for optimal hyperparameters. This research paper covered the proposed implementation plan, by examining various techniques used in Dream11 prediction. It introduced a model that focused on overcoming the limitations that were there in the existing approaches. The proposed system aimed to provide an efficient prediction model that integrated with the chatbot interface. The paper explored the entire project lifecycle, from conceptualization and methodology to implementation and results.

2. Methodology

2.1 Architecture

Figure 1 shows the System Architecture, which consists of a user interface that allows the users to connect with the system. Users could gain access to information related to cricket as well as fantasy cricket prediction and engage with the unified chatbot. The backend service included a prediction model that exercised algorithms of machine learning for the prediction of the Dream 11 team. The Database Layer stored historical cricket data, head-to-head player statistics, and player performance at each venue, which provided the information needed by the prediction model and chatbot response.

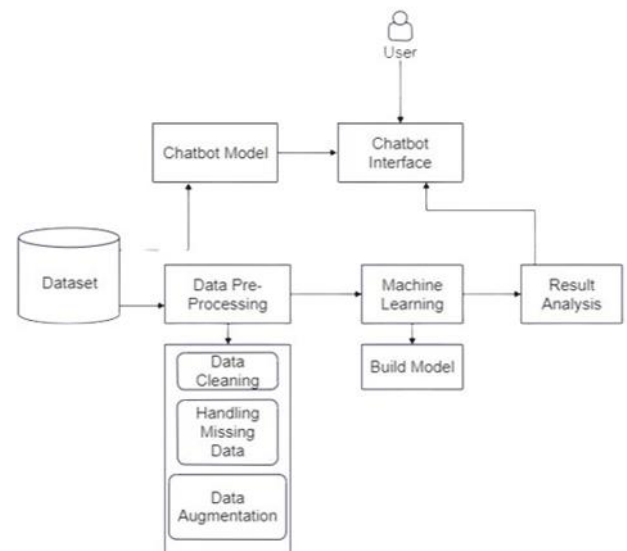


Figure 1 Architecture of The System

2.2 Dataset Description

For the proposed system, three main datasets were used to increase the precision and effectiveness. The main dataset used included cricket data which contained a record of all IPL matches that were played up to date [15]. This dataset helped to find the winner, toss outcome, and key performers of each match. The next dataset contained a head-to-head analysis [16] of each player against each other. This consisted of how each batsman played against each bowler. The last dataset included the player's performance at different venues [17]. This included batsman performance as well as bowler performance.

2.3 Dataset Processing

2.3.1 Head-to-Head Data

A formula was used to calculate a performance index for each player against every bowler. The formula was expressed as follows:

$$\begin{aligned}
 Performance_{Index} &= (Weights['Runs'] \\
 &\times data['Runs'] \\
 &+ Weights['Balls'] \\
 &\times data['Balls'] \\
 &+ Weights['Wickets'] \\
 &\times data['Wickets'] \\
 &+ Weights['StrikeRate'] \\
 &\times data['StrikeRate'])
 \end{aligned}$$

Following the calculation of the performance index, normalization was performed to scale the values between the range [0, 1]:

Normalized Perf Index

$$= \frac{\text{Performance_index} - \text{min_perf_index}}{\text{max_perf_index} - \text{min_perf_index}} \times 20$$

Later, the normalized performance index was mapped to distinct scores within the range of 0 to 10:

$$\text{Batsman_Score} = \text{round}(\text{Normalized_Perf_Index})$$

This approach ensured a meaningful and standardized evaluation of each player's batting performance, which helped the accuracy and relevance of the predictive algorithms integrated into the cricket prediction system.

2.3.2 Venue-Wise Player Performance Data

At each venue, the batsmen and bowlers perform differently. Probability Score (PS) was used to calculate the performance of players. For batting performance, it combined batting average and strike rate at each venue where the match was played. The PS for a batsman B_i at venue V was calculated using the following formula:

$$\text{PS}_{B_i} = (2 \times \text{Avg}^*_{B_i, V}) + (3 \times \text{SR}^*_{B_i, V})$$

Where $\text{Avg}^*_{B_i, V}$ and $\text{SR}^*_{B_i, V}$ represented the batting average and strike rate of batsman B_i at venue V , respectively. Similar to batting performance, bowling performance was also assessed. The factors that were taken into consideration were the number of wickets taken and the economy rate of the bowlers at each venue. The PS for a bowler B_j at venue V was computed as:

$$\text{PS}_{B_j} = (3 \times \text{Wkts}^*_{B_j, V}) + (2 \times (1 - \text{ER}^*_{B_j, V}))$$

Here, $\text{Wkts}^*_{B_j, V}$ and $\text{ER}^*_{B_j, V}$ denoted the normalized wickets taken and the economy rate of bowler B_j at venue V , respectively.

2.4 Score Calculation Using Machine Learning

Based on the scores calculated and the machine learning model, the proposed system then used a technique for calculating the scores of the players. Initially, using the machine learning algorithm, winning and losing teams were determined and initial scores were assigned to all the players of the teams. Higher scores were assigned to players belonging to the winning team. The system then analyzed the performance of key players of both teams from the last two matches and updated the scores. Further, the scores calculated from venue-wise data and head-to-head data were added to adjust the scores. The scores were then combined and normalized and a final score was calculated for each player.

2.5 Chatbot Model

With the help of Botpress [5], a chatbot was developed. This allowed users to interact with the bot and find cricket-related information. The information provided included match statistics, player information, historical data, or any other information needed. By concentrating on cricket, the users are ensured that they receive the information they want. Using Botpress, users are guaranteed a streamlined experience, easy accessibility, and engagement.

3. Implementation

3.1 Support Vector Machine

The Results should include the rationale or design of the experiments as well as the results of the experiments. Results can be presented in figures, tables (1), and text.

3.2 AdaBoost Classifier

The AdaBoost Classifier [11] trained weak classifiers on match data iteratively and then adjusted the focus on difficult ones. It was used to predict the match winners based on features like home team, away team, and venue as it combined weak learners into strong learners to improve accuracy.

3.3 Logistic Regression

Logistic Regression (LR) [3-8] tried to fit a logistic curve to the input data, which consisted of features such as home team, away team, and venue. This helped to predict winners by predicting the probability of the winning team with accuracy.

3.4 Random Forest Classifier

The Random Forest Classifier [1-2] combined multiple decision trees to predict the winning team. The final prediction of this classifier is calculated by combining predictions of individual trees to improve accuracy, remove overfitting, and make reliable choices for match winner.

3.5 Gradient Boosting Classifier

The Gradient Boosting Classifier [7], built a sequence of decision trees using features such as home team away team, and toss outcome during training to predict the winning team. This classifier helped in achieving accuracy which handled complex interactions between features.

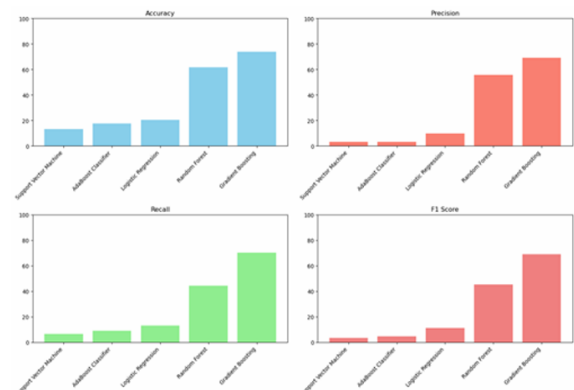


Figure 2 Evaluation of Different Models

As shown in figure 2. the graph shows the evaluation metrics for different machine learning models implemented.

4. Result and Analysis

Different machine learning algorithms were used to train the model and provide a set of outcomes. Evaluation metrics including Accuracy, Precision, Recall, and F1-score were used to measure the performance of the machine learning algorithms [9]. The formulas for the evaluation metrics are as follows:

$$\text{Accuracy} = \frac{(TP + TN)}{TP + TN + FP + FN}$$

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

The following data shows the results that were obtained by employing the different machine-learning algorithms for model training:

Table 1 Result Analysis

Algorithm	Accuracy	Precision	Recall	F1 Score
Support Vector Machine	13.33	3.13	6.63	3.51
AdaBoost Classifier	17.77	3.20	9.12	4.51
Logistic Regression	20.55	9.83	13.11	11.12
Random Forest	61.67	55.77	44.32	45.16
Gradient Boosting	73.89	69.18	70.26	69.17

Based on the results observed, the Gradient Boosting Classifier was found to give the most accurate prediction for the proposed model with an accuracy of 73.89%. By including the Gradient Boosting Classifier in the model, based on the input given by the user the winning team was predicted. Based on the predictions, further score calculations of players were done and then displayed back to the users.

Conclusion

The proposed system showcased advancement in Dream11 prediction models. The system integrated the Dream11 prediction model with a chatbot interface that overcame the limitations that were there in the existing platforms by increasing the accuracy of prediction. With the help of a user-friendly interface, it enabled users to obtain the

fantasy team. The fusion of the machine learning model and chatbot in the system helped to enhance user experience and engagement. With the new features added, the probability of obtaining an accurate team increased and helped the users gain rewards on the fantasy application.

References

- [1]. A. B. Shaik and S. Srinivasan, "A brief survey on random forest ensembles in classification model," in International Conference on Innovative Computing and Communications, pp. 253–260, Springer, 2019.
- [2]. Ayush Tripathi, Rashidul Islam, Vatsal Khandor, Vijayabharathi Murugan. "Prediction of IPL matches using Machine Learning while tackling ambiguity in results," 2020
- [3]. Diaz-Quijano, F. A. (2012). A simple method for estimating relative risk using logistic regression.
- [4]. Evgeniou, T., & Pontil, M. (2001). Workshop on support vector machines: Theory and applications. Center for Biological and Computational Learning, and Artificial Intelligence Laboratory, MIT, E25-201, Cambridge, MA 02139, USA.
- [5]. J. U., & Deepamala, N. (2020). Chatbot for Business Environment Using Botpress Conversational AI Platform. IRJET
- [6]. Nilesh M. Patil, Bevan H. Sequeira, Neil N. Gonsalves, Abhishek A. Singh. "Cricket team prediction using machine learning techniques.," 2020
- [7]. Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. fortiss GmbH, Munich, Germany and Department of Informatics, Technical University Munich, Garching, Munich, Germany
- [8]. Peng, C.-Y. J., Lee, K. L., & Ingersoll, G. M. (2002). An Introduction to Logistic Regression Analysis and Reporting.
- [9]. Pramanik, M. A., Rahman, M. Z., Suzan, M. M. H., Kalaiarasi, A., & Biswas, A. A. (2022). Performance Analysis of Classification Algorithms for Outcome Prediction of T20 Cricket Tournament Matches. In Proceedings of the 2022 International Conference on Computer Communication and Informatics (ICCCI) (pp. Page Range). IEEE. DOI:

10.1109/ICCCI54379.2022.9740867

- [10].Rohit Tamrakar, Niraj Wani.” Design and Development of CHATBOT: A Review (2021)”
- [11].Ruihu Wang (2012) AdaBoost for Feature Selection, Classification and Its Relation with SVM*, A Review Ruihu Wang Department of Science and Technology Chongqing University of Arts and Sciences Yongchuan, Chongqing 402160, CHINA
- [12].Sachin Kumar S, Prithvi H V, C. Nandini. “Data Science approach to predict the winning fantasy cricket team - Dream 11 fantasy sports”, 2022
- [13].Saurav Singla, Swapna Samir Shukla. “Integer Optimisation for Dream 11 Cricket Team Selection”, 2020
- [14].Srikantaiah, K. C., Khetan, A., Kumar, B., Tolani, D., & Patel, H. (2021). Prediction of IPL Match Outcome Using Machine Learning Techniques.
- [15].<https://www.iplt20.com/matches/results>
- [16].<https://t20-head-to-head.com/ipl-head-to-head-player-stats/>
- [17].<https://t20-head-to-head.com/player-statistics-by-venue/>