Score Card Prediction – Cricket

1st Manan Pujara

*Department of Computer Enginnering*

*Marwadi University*

Gujrat,india

mananpujara666@gmail.com

***Cricket score prediction presents a tricky problem because diverse things affect it, such as team play, the pitch next to the weather. Old methods often don't grab these changing parts well. This work examines how machine learning could make score prediction better. It does this through study of past match information plus vital game data. Through the use of smart formulas, the model intends to give forecasts based more on information, which helps decision-making for teams, experts along with fans. The method suggested makes predictions better. It gives useful details about how the game works too.***

***Keywords-Cricket score prediction, machine learning, Random Forest, XGBoost, sports analytics, data-driven forecasting, predictive modeling, big data analysis, game dynamics, feature engineering***

# I. INTRODUCTION

Precise cricket score prediction is very important for better game breakdown, strategy development along with fan interest. Old prediction methods frequently don't work because cricket is dynamic. Pitch states player skill, in addition to weather all affect it. Machine learning gives a stronger data way. It checks huge datasets, spots secret trends as well as makes predictions better. With machine learning methods, we improve the forecasting of match results, team points as well as how well each player does.The research uses past match data, weather data as well as in-game measures. This develops a strong, smart system for score forecast. Because of this the main aim is to make cricket decisions easier. It supplies exact live predictions that help players, coaches, experts along with fans.

II. LITERATURE REVIEW/SURVEY

## A. Comprehensive Prediction Model for T20 and Test Match Outcomes Using Machine Learning

|  |
| --- |
| XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE |

Cricket is a popular sport worldwide, played with a bat and balls. This paper used classification and regression to predict the T20 and Test matches results and scores because cricket fans and analysts always want to predict which team will win the match and how much score a team will make. We have used a supervised machine learning technique to predict the score and result for T20 and Test matches. Cricket match prediction dataset comprising of 13 features and 7827 instances utilized for training the model. For the prediction of winning and losing, we have used Naïve Bayes, Gradient Boosted Trees, Logistic Regression, Deep learning, Decision Tree, Random Forest, and Generalized Linear Model. For Score Prediction, we have used Decision Trees,Random Forests, Gradient Boosted Trees, Generalized Linear Models, and Deep Learning. We've shown the most accurate model after evaluating the accuracy percentages of the several classifiers listed above. Our introduced model has gained an

accuracy rate of 96.15% using the Gradient Boosted Trees classifier for Match Score Prediction and 71.72% accuracy by using the Generalized Linear Model for match result prediction. The Rapid Miner tool has been used to perform machine learning techniques to train models. This paper discusses the effectiveness and use of machine learning methods to develop highly accurate models for Cricket match prediction dataset comprising of 13 features and 7827 instances utilized for training the model.

## B. One Day International Cricket Match Score Prediction Using Machine Learning Approaches

Cricket is the game that binds Bangladeshis to-gether. The datasets were split into different parts based on the time of the cricket matches to experiment with the results. We have used Random Forest Regression and Linear Regression to experiment with the results. We have worked on different sets of independent variables while working with different datasets but runs and wickets were constant features. The linear Regression technique gave 0.58 of r-squared metric and its MAE: 32.857046003557706, MSE:

1793.4360657965133, RMSE: 42.34897951304746 values indicate that the regression model is making predictions that are reasonably close to the actual ob-served values. However, there is still some room for improvement. The rsquared metric of 0.787 was derived from a relatively brief dataset, MAE: 20.48880467759807, MSE: 690.1167559064677, RMSE: 26.270073389818837 metrics indicate that the regression model is making predictions that are very close to the actual observed values. Random Forest Regression's r-squared metric resulted in 0.807 but the learning curve pointed out an overfit model. Thus, linear regression contained the best result. We have applied the regularisation technique to achieve a better result. The dataset was collected from Github and Kaggle, sizes are (350899, 15) and (4037, 9) respectively..

## C. Cricket Score Prediction Using Machine Learning Techniques

Cricket’s captivating nature, especially in the One Day International (ODI) format, has long fascinated a global audience. However, the sport’s unpredictability has posed challenges to reliable match outcome predictions. This research paper presents a machine learningbased model for predicting the total first-inning score of ODI cricket matches. To analyze the data, we use various data preprocessing techniques to clean and organize the data collected from 1188 ODI matches. The selected features for the prediction task include runs, wickets, overs, runs\_last\_5, wickets\_last\_5, striker, and non-striker. Using various metrics, we evaluate the performance of di!erent machine learning models, such as Random Forest, Gradient Boosting, XGB, LGBM, CatBoost, AdaBoost, ANN, and MLP. Our results show that Random Forest has the highest accuracy in predicting the final score of an ODI cricket match. By segmented modeling on the Random Forest, we improve the results to 87%. The study concludes by highlighting the potential applications of the developed model in predicting the outcome of future cricket matches and providing insights into the performance of teams and players.

## D. Sport analytics for cricket game results using

## machine learning: An experimental study

In the ever-evolving world of cricket, the T20 format has captured the imaginations of fans worldwide, intensifying the anticipation for match outcomes with each passing delivery. This study explores the realm of predictive analytics, leveraging the power of machine learning to alleviate the suspense by forecasting T20 cricket match winners before the first ball is bowled. Drawing on a rich dataset encompassing factors such as past team performance and rankings, a diverse ensemble of predictive models, including logistic regression, support vector machine (SVM), random forest, decision tree, and XGBoost, is meticulously employed. Among these, the random forest Classifier emerges as the standout performer, boasting an impressive prediction accuracy rate of 84.06%. To assess the real-world applicability of our predictive framework, a post-case study is conducted, focusing on the high-stakes World Cup T20 matches of 2022, where England emerges as the triumphant team. The dataset underpinning this study is meticulously curated from ESPN CricInfo, ensuring the robustness of our analysis. Moreover, this paper extends its contribution by offering a comprehensive comparative analysis, scrutinizing performance metrics such as accuracy, precision, recall, and the F1-score across benchmark machine learning models for cricket match prediction. This in-depth evaluation not only validates the efficacy of our models but also sheds light on their superior execution time and statistical robustness, further bolstering their utility in the realm of cricket outcome forecasting.

III. PROPOSED METHODOLODY

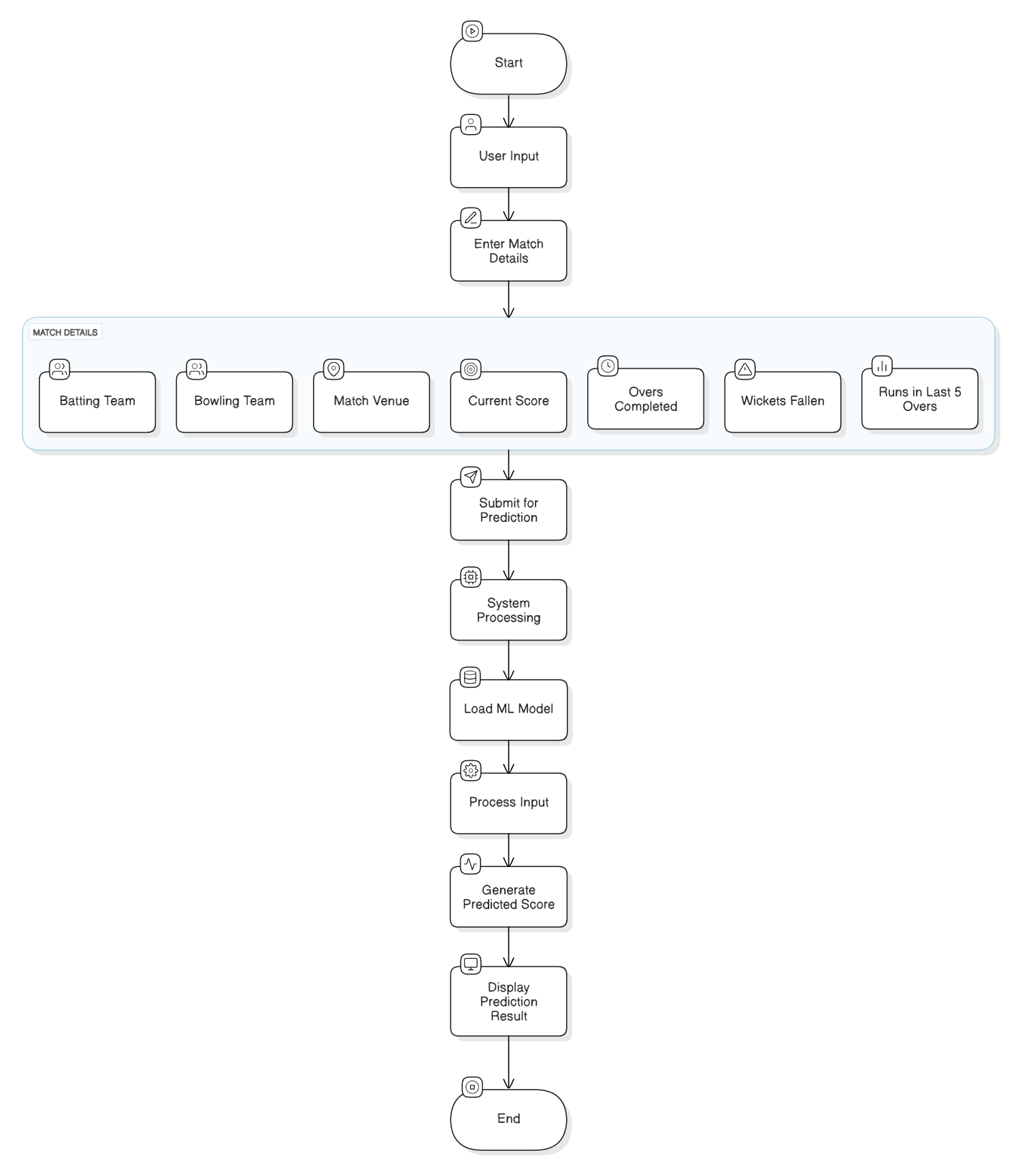


Fig. 1. Workflow of the Cricket Score Prediction System

The cricket score prediction system uses a structured method to get precise and useful results. The flow starts at the User Input stage - here, users offer necessary information about the match. This data contains details like the Batting Team, Bowling Team, Match Venue, Current Score, Overs Completed, Wickets Fallen next to Runs in the Last 5 Overs. These features help document the existing dynamics of the match. After data entry the user sends it for prediction.

After that the system goes to Processing. The input becomes ready plus changes to match what the model needs. Either Random Forest or XGBoost, the Machine Learning Model starts to examine the data. It creates a predicted score. This score comes up on the interface and it tells users about possible results. Because detailed input features connect to a structured prediction pipeline, this simple workflow makes the prediction system more accurate. The modular design promises that the system adapts to later improvements besides feature additions.

## A. Data Collection and Preprocessing

The initial step gathers ball-by-ball cricket match data through various available sources including Kaggle and official cricket APIs. A total of 300 T20 matches in YAML format were processed through automatic parsing for conversion into structured CSV and Pandas DataFrame structures. A two-step data cleaning process followed the collection phase:

1) Level 1 Cleaning implemented two stages which normalized data by removing null values and duplicate entries and irrelevant columns and performing data normalization.

2) Level 2 Cleaning stage employed feature engineering to develop meaningful metrics which included runs\_last\_5 and wickets\_last\_5 and strike\_rate variables. The cleaned dataset was saved under dataset\_level1.pkl and dataset\_level2.pkl to enable future utilization.

## B. Feature Engineering

The target prediction (final score) mainly depends on three extracted significant features. The model uses Batting team, bowling team, venue and match type as contextual features. A set of game features includes the current score combined with the number of completed overs alongside the wickets taken while also revealing runs and wickets from the past five overs and striker and non-striker status. The model incorporates a set of time-based features which include over progression alongside powerplay indicator and innings phase. The implementation maintained proper encoding techniques on the input characteristics including categorical data encoding via one-hot and numerical data scaling for machine learning model consumption.

## C. Model Selection and Training

Three ensemble learning algorithms were utilized to solve the regression prediction challenge of cricket final scores. The Random Forest Regressor constructs multiple decision trees from which it calculates an average output in a bagging-based ensemble framework.

The algorithm demonstrates resistance to outliers and minimizes overfitting by using bootstrapped selection among randomly chosen features. XGBoost Regressor (XGB) represents an efficient gradient boosting framework which uses both L1/L2 regularization to optimize its performance while enhancing speed. XGBoost has a sequential error correction system which adds L1/L2 regularization features to stop overfitting during prediction.

The LightGBM Regressor (LGBM) stands as a gradient boosting framework that delivers high efficiency and scalable operation. This algorithm adopts leaf-wise growth trees that deliver higher speed and reduced memory usage than XGBoost even though it operates differently from level-wise growth modes. Stable preprocessing methods from scikit-learn standardized the processing of all three models. Categorical features such as batting team and bowling team received one-hot encoding transformation while the venue remained unchanged.

Feature standardization for numerical values. A 80:20 train-test split occurred before the models performed optimization through cross-validation using GridSearchCV on their hyperparameters. The evaluation of generalization capability for each model used three performance metrics that include R² Score together with Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). Deployment of the XGBoost model was selected as the best solution although both Random Forest and LightGBM models were integrated with a frontend feature for user-selected model analysis.

## D. Prediction Workflow

The end-to-end prediction system shows the operations as depicted in Figure 1. The pipeline steps are

The system accepts current match parameters through its user interface from the user. The system prepares data by encoding and formatting it to match the specifications that were established during model training. The processor directs the modified input to the chosen model either RFR or XGB which generates the forecasted final score. The system provides the user with a predicted score together with optional confidence metrics or prediction intervals in the output display stage.

# IV. RESULTS AND DISCUSSION

The assessment of Random Forest Regressor (RFC), XGBoost, and LightGBM models regarding cricket score prediction constitutes the following part. A cleaned dataset which was feature-engineered from over 300 T20 matches provided the data for evaluation of the three models through train-test splits with an 80:20 ratio. Four regression metrics evaluated the results including R² Score together with Mean Absolute Error (MAE) and Mean Squared Error (MSE) and Root Mean Squared Error (RMSE).

## A. Model Performance Evaluation

The performance assessment of the three models appears in Table I alongside Fig. 5. The results showed XGBoost reaching the best overall performance through its R² score of 0.9802 which indicates reliable prediction accuracy. Both XGBoost and LightGBM showed equivalent performance characteristics yet LightGBM needed less time to train and demanded fewer resources. Random Forest provided a less accurate yet interpretable and stable performance that made it effective for baseline modeling activities.

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **R² Score** | **MAE** | **RMSE** |
| Random Forest | 0.9386 | 5.56 | 8.28 |
| XGBoost | 0.9802 | 4.20 | 5.30 |
| LightGBM | 0.9781 | 4.33 | 5.38 |

Table 2: Model Evaluation

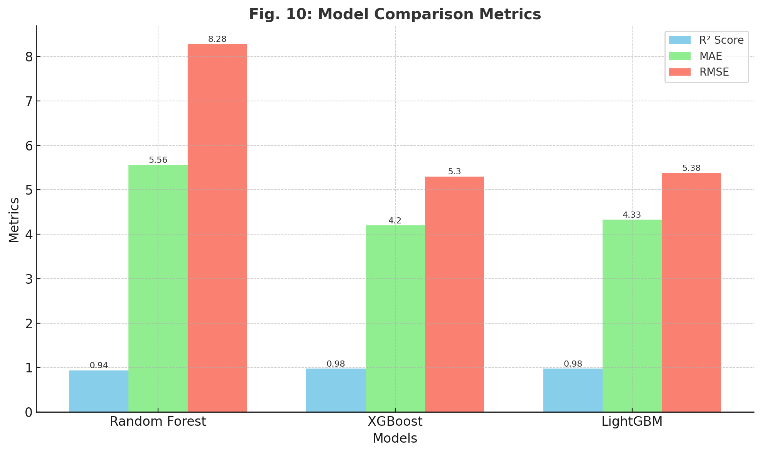


Fig.7: Bar Chart of Model Comparison Metrics

## B.Analysis of Results

The metrics show that ensemble gradient boosting techniques (LightGBM and XGBoost) achieve superior accuracy and error reduction than the bagging methods (RFC). XGBoost manages large collections of features effectively while maintaining lower errors during operation because it uses sequential learning algorithms. The high performance of LightGBM remains competitive alongside its faster computational operations. The Random Forest model showed reliability when used during the testing and iterative phases of model development despite its reduced speed at understanding complex patterns. The deployment stage performed prediction checks against unknown test data samples where the predicted outputs showed discrepancies of less than 10 runs in most tested situations. Evidence shows that the model produces accurate predictions across different pitch situations and match scenarios.

## C.Real-Time Testing

The Streamlit-based frontend enabled users to check real-world usability by letting them enter live match parameters including the teams batting and bowling and numbers for 5 overs completed and runs scored. The real-time tests of the models demonstrated their response timing as well as system stability.

The large ensemble of Random Forest required additional processing time until it produced its predictions. The models XGBoost and LightGBM displayed rapid responsiveness yet LightGBM exhibited the quickest response time out of these three models. Example predictions:

1) Input: India vs Australia, 110/3 in 13.2 overs

RFC: 160, XGB: 171, LGBM: 170 (Actual: 173)

2) Input: Pakistan vs England, 90/2 in 11 overs

RFC: 150, XGB: 159, LGBM: 158 (Actual: 162)

## D.Discussion

The similarity between original and forecasted scores proves that the selected features and tuned model hold valid scientific strength. Cricket game rules allow for small prediction variations due to its natural unpredictable nature. Users benefit from multiple integrated models because it enables them to view predictions alongside assessment of prediction confidence through the user interface.

Future developments become possible due to the achieved outcomes. Adopting statistics from the match along with current field data together with weather features available through APIs would enhance both accuracy and model reliability.

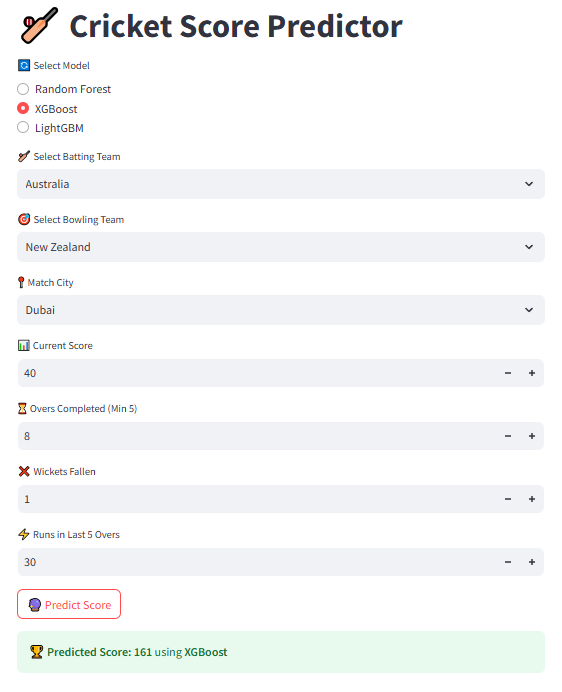


Fig.7: User Interface with model selection

V. FUTURE DIRECTIONS

The research provides an established base to predict cricket scores in real-time by using structured datasets alongside combination machine learning models. The system can be improved through multiple pathways to boost its performance and precision.

*A.**Integration of External Data Sources*

Every aspect of the current system depends only on historical match data outcomes. The system development should enable integration of external APIs which will provide:

1) The application retrieves real-time ball-by-ball updates through its football or cricket broadcasting APIs from Cricbuzz and ESPNcricinfo.

2) External weather APIs should be integrated to handle environmental factors.

3) The input of Player Statistics APIs allows users to access recent statistics about strike rate and economy and player form. The model would improve its real-time simulation capabilities through these dynamic inputs which enable it to handle unpredictable match events***.***

*B. Deep Learning-based Alternatives*

The Random Forest and XGBoost ensemble methods deliver effective performance yet deep learning RNNs and LSTM models would provide enhanced capability to understand time-based relationships. The present score in cricket chiefly derives from the past overs together with wickets taken along with ongoing momentum variation. Using sequence models can:

1) Capture match momentum over time.

2) Model batting and bowling rhythm.

3) Incorporate over-wise performance trends.

New systems should test hybrid architectures made of CNN-LSTM and Transformer-based models to improve contextual understanding*.*

*C. Model Interpretability with Explainable AI (XAI)*

An increase in usage requires explanation methods that help users understand how the models take decisions. The combination of SHAP (SHapley Additive exPlanations) and LIME libraries helps achieve the following:

1) Feature importance visualization.

2) Score adjustment traceability.

3) Transparency for analysts and end-users.

The inclusion offers additional support for model audit activities combined with strengthened trust levels necessary for commercial implementation.

# VI. CONCLUSION

Structured machine learning techniques combined with strong data engineering produce trustworthy real-time analysis capabilities for sports analytics applications via Scorecard Prediction. A prediction system based on historical T20 cricket data now enables score forecasting of an inning through current match data containing overs played and wickets taken as well as team statistics. The initial stage extracted data from more than 3000 matches written in YAML format before applying a two-tier data cleaning process for constructing a proper dataset structure. High-value features in the platform required extraction followed by transformation so they could function effectively in model development. The accuracy assessment for over 3000 matches yielded 0.9802 R² from XGBoost which proved to be the top performing machine learning model among Random Forest and LightGBM.

# 

## ACKNOWLEDGMENT

We would like to express our sincere gratitude to our mentor and faculty members for their valuable guidance and continuous support throughout the course of this project. Their encouragement and insights greatly helped in shaping the direction and execution of this work.We would also like to thank our peers and the contributors to open-source platforms and datasets like Cricsheet, whose resources made this project feasible and effective.

## REFERENCES

1. Zoha Ahsan; Shahwaiz Ghumman; Attique Ur Rehman; Sabeen Javaid; Tahir Muhammad Ali; Azka Mir, “A Comprehensive Prediction Model for T20 and Test Match Outcomes Using Machine Learning” ,23-23 May 2024.
2. Md. Sabbir Hossain; Ummay Khadiza Rumpa; Md. Shakil Hossain; Sazzadul Islam Shovon; Md. Tahmidul Huque; Mahamudul Hasan, “One Day International Cricket Match Score Prediction Using Machine Learning Approaches”
3. Amna Zafar, “Cricket Score Prediction Using Machine Learning Techniques”
4. Sanjay Chakraborty , Arnab Mondal and Lopamudra Dey, “Cricket data analytics: Forecasting T20 match winners through machine learning”
5. Trevor Hastie, Robert Tibshirani, Jerome Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction", Springer Series in Statistics, 2nd Edition, 2009.
6. Tianqi Chen, Carlos Guestrin, “XGBoost: A Scalable Tree Boosting System,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.
7. Microsoft Learn, “Introduction to Machine Learning,” [Online]. Available: https://learn.microsoft.com/en-us/training/modules/intro-machine-learning/
8. scikit-learn Developers, “scikit-learn Documentation,” [Online]. Available: https://scikit-learn.org/stable/user\_guide.html