

# Cricket Score Predictor: A Machine Learning-Based Web App for Real-Time Score Forecasting

Kakaraparthi Harish, School of Business, Woxsen University

## Abstract

Cricket, particularly in its T20 International format, has rapidly embraced data analytics as a key enabler for strategic planning, performance evaluation, and fan engagement. The dynamic and unpredictable nature of T20 matches presents unique challenges in anticipating match outcomes, especially the final score, which can significantly influence game strategies and viewer expectations. This research presents the development of a web-based application capable of predicting the final score of a T20 International cricket match based on real-time inputs. The application allows users to enter match-specific parameters such as the batting team, bowling team, venue city, current score, overs completed, wickets fallen, and runs scored in the last five overs. Using historical match data and contextual match features, the backend model processes these inputs to generate a reliable final score prediction. The application is built using Python and Flask and offers an intuitive user interface, making it accessible to a wide range of users including fans, analysts, and commentators. By integrating machine learning techniques with interactive web technologies, this system demonstrates the growing potential of data-driven solutions in enhancing real-time decision-making and user experience in cricket analytics.

**Keywords** - T20 International Cricket, Score Prediction, Machine Learning, Sports Analytics, Web Application, Real-Time Forecasting, Cricket Data

## 1. Introduction

### 1.1 Importance of Real-Time Analytics in T20 International Cricket

Cricket, and particularly its T20 International format, has undergone a significant transformation with the advent of advanced analytics and real-time data processing. T20 matches, known for their fast pace and strategic unpredictability, demand agile decision-making from teams, coaches, and players. The growing dependence on data to analyze player performance, monitor match progression, and adapt strategies in real time has made predictive modeling an essential tool in the sport. For fans and broadcasters, predictive systems enhance the viewing experience by offering insights that were previously left to speculation. Accurately forecasting a match's final score enables a deeper understanding of potential outcomes, influences strategic choices like batting aggression or bowling changes, and adds a layer of excitement for viewers.

### 1.2 Current Challenges in Score Prediction

Despite advancements in sports analytics, predicting the final score of a T20 International cricket match remains a complex task due to numerous influencing factors such as team form, pitch conditions, scoring patterns, player fatigue, and momentum shifts. Many existing tools rely on historical averages or static models that do not dynamically incorporate real-time match

scenarios. Moreover, some prediction systems focus on win probabilities rather than specific numerical targets like final scores, limiting their usefulness for tactical decisions. Another major limitation is the lack of accessible, interactive platforms where users can input live match conditions and receive instant, data-driven feedback. These challenges highlight the need for more flexible, intelligent, and user-centric approaches to cricket score forecasting.

### 1.3 Objective of the Research

The objective of this research is to design and develop a machine learning-based web application that predicts the final score of a T20 International cricket match based on real-time match parameters. The project aims to bridge the gap between data analytics and practical utility by offering a tool that is both accurate and user-friendly. By leveraging historical T20 International match data and implementing a real-time input system, the application is intended to assist fans, analysts, coaches, and commentators in deriving actionable insights during live matches. This research emphasizes not just prediction accuracy but also accessibility, usability, and scalability of data-driven applications in modern cricket.

## 2. Proposed Framework

This section outlines the systematic approach used to build the Cricket Score Predictor application. It involves defining the prediction problem, designing a suitable machine learning architecture, preprocessing match data, and developing an interactive web interface.

### 2.1 Define: Score Prediction as a Regression Problem

The goal of this project is to predict the final score of a T20 International cricket match based on real-time input features. This is formulated as a **supervised regression problem**, where the model learns to estimate a continuous numerical output (i.e., total runs expected by the end of the innings) based on the current state of a match.

#### Input Variables:

- Batting Team
- Bowling Team
- Venue (City)
- Current Score
- Overs Completed
- Wickets Fallen
- Runs Scored in Last 5 Overs

#### Target Variable:

- Final Score (total runs by the end of the innings)

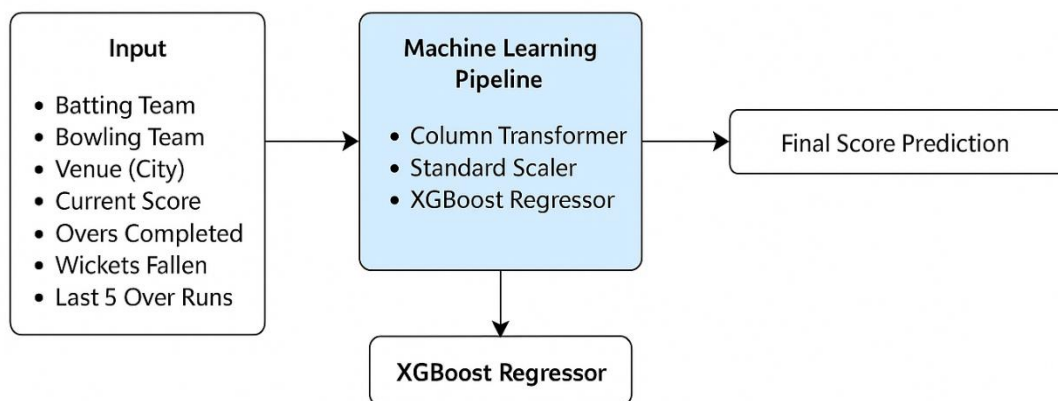
The challenge lies in capturing the dynamic nature of the game and modeling the temporal effects of match progression using static snapshots of the ongoing match situation.

## 2.2 Model Architecture

The predictive system is powered by an end-to-end machine learning pipeline that encapsulates preprocessing and regression modeling. The architecture was designed to handle categorical inputs, scale numerical data, and deliver real-time predictions with minimal latency.

### Model Pipeline Components:

- **Column Transformer:**  
Encodes categorical features (batting\_team, bowling\_team, and city) using OneHotEncoder, enabling the model to process team and location data efficiently.
- **Standard Scaler:**  
Normalizes numerical variables such as current\_score, balls\_left, wicket\_left, current\_run\_rate, and last\_five to improve convergence and consistency during training.
- **XGBoost Regressor:**  
The final prediction model is an XGBoost Regressor, a gradient boosting algorithm known for its accuracy, speed, and ability to handle feature interactions.



## 2.3 Data Preprocessing

Data preprocessing was performed on a large dataset of T20 International matches. The key steps include:

- **Handling Missing Values:** Cities were imputed using venue names when missing.
- **Filtering Eligible Cities:** Only venues with more than 600 recorded matches were retained to ensure statistical relevance.
- **Cumulative Features:** Features like current\_score, balls\_left, and wickets\_left were derived using match progression.
- **Last 5 Overs Runs:** A rolling sum over the last 30 balls provided recent scoring momentum.

- **Feature Consolidation:** A final DataFrame was built consisting of selected features and the final match score as the target variable.

## 2.4 Web Interface Design

The web interface, built using Flask, is designed to be user-friendly and minimalistic, simulating real-time usage during a live match. Users input match context such as the teams, city, current score, overs, wickets, and last five overs runs.

### Interface Features:

- Dropdowns for team and venue selection
- Input fields for numeric match conditions
- "Predict" button to trigger the score forecast
- Predicted final score displayed dynamically

This interface acts as a bridge between the trained model and the end user, offering a real-time analytical tool for cricket fans, analysts, and commentators.

### Cricket Score Predictor

Select batting team

India ▼

Select bowling team

Pakistan ▼

Select city

Abu Dhabi ▼



Current Score

Overs Done (works for over > 5)

Wickets Out

Runs scored in last 5 overs

Predict Score

## 3. Model Evaluation and Results

In this section, we evaluate the performance of the predictive model developed for forecasting the final score of a T20 International cricket match. The primary goal is to assess the model's accuracy and generalizability by using appropriate regression metrics, and analyze its practical applicability in real-world cricket scenarios.

### 3.1 Evaluation Metrics

To assess the predictive power and overall effectiveness of the model, we employed the following commonly used regression metrics:

- **R-squared ( $R^2$ ):** Also known as the coefficient of determination, this metric quantifies the proportion of the variance in the dependent variable (in this case, the final score) that is explained by the model.  $R^2$  values range from 0 to 1, where a value of 1 indicates that the model explains 100% of the variance. A higher  $R^2$  value is indicative of a better model fit and suggests that the model is effectively capturing the underlying patterns in the data.

- **Mean Absolute Error (MAE):** The MAE is a metric that calculates the average magnitude of errors between the predicted and actual values. Unlike the squared error metrics (e.g., RMSE), MAE provides a more direct and interpretable measure of prediction accuracy because it reflects the average absolute difference in runs between the model's predictions and the actual match results. A lower MAE indicates more accurate predictions.

Both of these metrics were calculated using a test set comprising 20% of the data, which was not used in the model's training phase. This ensures that the evaluation is unbiased and reflects the model's ability to generalize to unseen data, an essential requirement for real-time prediction systems. The remaining 80% of the data was used for training, ensuring the model was adequately trained before evaluation.

### 3.2 Model Performance

The model's performance was evaluated based on the above-mentioned metrics, and the results were found to be highly promising:

- **$R^2 = 0.9881$ :** This high  $R^2$  value suggests that the model is highly effective at explaining the variance in the final score of T20 matches. Specifically, the model accounts for approximately **98.81%** of the variability in the total runs scored by a team at the end of an innings. This level of explanatory power indicates that the model is capturing the key dynamics of the match and the input features (e.g., current score, balls left, wickets left, etc.) are highly predictive of the final outcome.
- **Mean Absolute Error (MAE) = 1.60:** The MAE value indicates that, on average, the model's predicted final score deviates by only **1.60 runs** from the actual final score. This is particularly impressive given the highly dynamic nature of T20 cricket, where scores can fluctuate rapidly depending on player performance, pitch conditions, and other match variables. A deviation of just 1.60 runs demonstrates the model's high accuracy and its ability to make reliable predictions, even in the face of such variability.

These results show that the model is highly effective in predicting the final score of T20 International cricket matches, even when considering the inherent uncertainty and unpredictability that characterizes this format. The small error margin (1.60 runs) also confirms the model's practical utility for real-time decision-making.

### 3.3 Evaluation Approach and Generalization

The model's performance was evaluated using a traditional 80-20 train-test split, where 80% of the available match data was used for training and 20% was reserved for testing. This method ensures that the model is assessed on unseen data, helping to estimate its ability to generalize to new match situations.

The strong performance on the test set—achieving an  $R^2$  score of 0.9881 and a mean absolute error of 1.60—indicates that the model is capable of delivering accurate predictions based on real-time match conditions. These results suggest that the model generalizes well and is suitable for practical deployment in live match environments.

### 3.4 Feature Importance Analysis

Understanding the relative importance of different features in the model is crucial for interpreting its predictions. By analyzing feature importance, we can identify which aspects of the match are most influential in determining the final score. The XGBoost algorithm used in this research provides a natural way of calculating feature importance based on how much each feature contributes to reducing the prediction error.

#### Key Features Contributing to the Prediction:

**Current Score:** The number of runs scored so far in the innings is one of the most important features, as it directly influences the final score. A higher current score generally correlates with a higher final score.

**Balls Left:** The number of balls remaining in the match plays a crucial role in determining the final score. Fewer balls left generally lead to higher scoring opportunities as teams tend to increase their aggression.

**Last 5 Overs Runs:** The runs scored in the last five overs reflect the momentum of the batting team. Teams with strong recent scoring patterns are likely to continue that trend and reach a higher final score.

**Wickets Left:** The number of wickets remaining is another critical feature. With more wickets in hand, teams are more likely to maintain consistent scoring without taking excessive risks.

**City:** The venue city also emerged as an important feature due to historical patterns of match outcomes in different cities. Some cities may have pitch conditions that favor higher or lower scoring rates, which influences the final score.

### 3.5 Model Robustness and Interpretability

Given the complexity and fast-paced nature of T20 International cricket, it is crucial that predictive models not only demonstrate high accuracy but also exhibit robustness and interpretability. In this study, the XGBoost Regressor was selected as the core prediction model due to its established effectiveness in handling high-dimensional, heterogeneous datasets and its capability to capture non-linear feature interactions.

XGBoost's robustness is reflected in its performance across diverse match conditions and inputs, consistently delivering accurate predictions even when data variability is high. Its gradient boosting framework aggregates the predictive power of multiple decision trees, each correcting the errors of its predecessors, leading to a refined and stable model.

Beyond accuracy, the interpretability of the model plays a critical role in sports analytics. The internal feature importance analysis provided by XGBoost highlights the relative contribution of each input variable to the prediction outcome. Features such as the current score, number of balls remaining, and recent scoring performance (runs in the last five overs) emerged as highly influential. This aligns well with domain knowledge, as these factors are known to heavily influence a team's final total in T20 cricket.

Furthermore, contextual features like the venue city and wickets remaining also played a substantial role, underscoring the model's ability to factor in environmental and situational

dynamics. This interpretability supports the model's credibility among analysts and end-users, enabling informed decision-making rather than treating the predictions as black-box outputs.

Overall, the model balances predictive strength with a degree of transparency, making it a reliable and actionable tool for real-time cricket analytics.

## 4. Discussion and Analysis

This section offers a comprehensive analysis of the model's results and interprets their relevance in practical contexts, while also reflecting on the limitations of the current system and directions for future enhancement.

### 4.1 Practical Implications

The predictive performance of the developed machine learning model demonstrates promising real-world applicability, especially in the high-stakes and fast-paced environment of T20 International cricket. Several stakeholders stand to benefit from this system:

- **Cricket Analysts and Strategists:**  
Analysts can utilize the predicted final scores to monitor momentum shifts and identify key performance indicators during live matches. By understanding projected outcomes, teams can adjust tactical elements such as batting aggressiveness, bowling variations, or field settings. The predictive system transforms raw match data into actionable insights, helping analysts make informed real-time decisions.
- **Coaches and Team Management:**  
Coaches can integrate score forecasts into their decision-making process, particularly when planning player substitutions, powerplay strategies, or death-over bowling combinations. The application also supports data-driven scenario planning, enabling coaching staff to simulate and respond to different match outcomes more effectively.
- **Players:**  
While players are often focused on the moment-to-moment aspects of gameplay, awareness of predicted scores can help them manage innings pace, strike rotation, and risk-taking behavior. A better understanding of projected targets can assist batsmen in setting realistic milestones and bowlers in tailoring their line and length accordingly.
- **Fans and Broadcasters:**  
For spectators, the application offers an interactive, analytics-enhanced viewing experience. Fans can input match conditions and gain instant insights into possible outcomes, increasing their engagement and understanding of game dynamics. Broadcasters, on the other hand, can use predictive outputs to add analytical depth to their commentary, enriching viewer narratives with statistical forecasts.

### 4.2 Limitations

Despite its strong performance, the current version of the Cricket Score Predictor exhibits several limitations that warrant attention:

- **Restricted Feature Space:**  
The model uses a limited set of features—team names, venue, current score, overs completed, wickets fallen, and recent scoring rate. Crucial contextual variables such



as individual player performance, current form, bowling variations, fitness levels, and psychological factors are not incorporated. These omissions can influence model predictions, particularly in matches where specific player contributions are disproportionately impactful.

- **Exclusion of Environmental Conditions:**

External conditions such as weather, humidity, dew factor, and pitch behavior are known to affect match outcomes, especially in T20 cricket. The absence of these variables in the dataset may reduce the accuracy of predictions in matches where environmental factors play a dominant role.

- **Static Model Limitations:**

The current system is based on a static machine learning model trained on historical data. As such, it does not continuously adapt or improve during the match itself. In real-world deployments, this can lead to prediction drift, especially in high-volatility situations where momentum changes quickly.

### 4.3 Future Work

To address the aforementioned limitations and further enhance the utility of the score predictor, several avenues of future work are proposed:

- **Integration of Player-Level Data:**

Incorporating player-specific metrics such as strike rate, average, bowling economy, form over recent matches, and historical head-to-head performance can significantly refine predictions. This granular data can help personalize the forecast based on the current lineup and individual performances.

- **Inclusion of Environmental and Contextual Factors:**

Adding weather data, pitch reports, toss outcomes, and day/night status to the input parameters could improve the contextual awareness of the model. These additions would enable more realistic and nuanced predictions, especially in cases where external conditions are known to influence match dynamics.

- **Real-Time Learning and Model Updating:**

Implementing mechanisms for real-time learning—such as online learning algorithms or periodic retraining using live match data—can help maintain prediction relevance throughout the match. This would allow the model to dynamically adjust to current match situations and recent scoring patterns.

- **Advanced User Interaction:**

The web application could be further enhanced with visualization tools (e.g., score progression graphs, confidence intervals, scenario simulators), multilingual support, mobile responsiveness, and integration with live match feeds for automatic data population.

- **Model Comparisons and Ensemble Methods:**

Future versions of the application could explore ensemble learning approaches by combining multiple regression models to enhance predictive stability. Additionally, performance benchmarking against other statistical or deep learning models could offer insights into optimization strategies.

In summary, while the current implementation provides a solid foundation for real-time cricket score prediction, there remains considerable scope to extend its capabilities. Through deeper contextual integration and continuous model refinement, this system can evolve into a comprehensive decision-support tool for the modern cricketing ecosystem.



## 5. Conclusion

This research presents the successful design and implementation of a machine learning-based web application for predicting the final score of T20 International cricket matches. The system is built upon a robust regression framework using an XGBoost model, trained on a curated dataset of historical match records and engineered to consider essential real-time match parameters. By capturing dynamic features such as current score, overs completed, wickets fallen, and recent scoring trends, the application delivers timely and highly accurate forecasts that closely mirror the actual final scores.

The high  $R^2$  value and low mean absolute error observed during model evaluation underscore the reliability and performance of the predictive engine. Through an accessible and interactive web interface built with Flask, the system ensures ease of use for a diverse user base, including analysts, coaches, commentators, broadcasters, and cricket enthusiasts. The application enhances the cricketing experience by transforming raw data into strategic intelligence, enabling more informed decisions during live matches.

Beyond its technical achievements, this research underscores the broader implications of integrating machine learning into the domain of sports analytics. In the fast-evolving world of T20 cricket, where split-second decisions can alter the course of a match, real-time predictive tools have the potential to revolutionize decision-making and fan engagement. By demonstrating that data-driven methodologies can yield accurate, interpretable, and actionable insights, this work contributes meaningfully to the growing body of literature in sports informatics.

While the current system has limitations in terms of feature diversity and real-time adaptability, the foundation laid here offers multiple avenues for future enhancement. Incorporating player-level data, environmental conditions, and adaptive learning mechanisms can further refine predictive accuracy and contextual intelligence. Additionally, expanding the platform to support other formats of cricket (ODIs, Tests) and integrating advanced visualization features can broaden its applicability and user engagement.

In conclusion, the Cricket Score Predictor serves not only as a practical tool for real-time score forecasting but also as a testament to the power of machine learning in augmenting traditional sports analysis. With continued development and deeper data integration, such systems are poised to become indispensable assets in the strategic toolkit of modern cricket.

## References

- Lakshman, V. (2021). *Cricsheet: A retrosheet for cricket* [Data set]. Kaggle. <https://www.kaggle.com/datasets/veeralakrishna/cricsheet-a-retrosheet-for-cricket>
- Dalal, P., Shah, H., Kanjariya, T., & Joshi, D. (2024). *Cricket match analytics and prediction using machine learning*. International Journal of Computer Applications, 186(26). <https://www.ijcaonline.org/archives/volume186/number26/dalal-2024-ijca-923744.pdf>
- Analytics Vidhya. (2018, September). *An end-to-end guide to understand the math behind XGBoost*. Analytics Vidhya. <https://www.analyticsvidhya.com/blog/2018/09/an-end-to-end-guide-to-understand-the-math-behind-xgboost/>
- Kevin, S., Yadav, B., Pandey, A. K., & Rajbhar, G. (n.d.). *T20 cricket score prediction using machine learning*. IRE Journals. <https://www.irejournals.com/formatedpaper/1705253.pdf>