IPL Score Prediction Using Deep Learning

G.Parinitha\*1,P.Varun Teja2, V.Sujith3, Rame Swetha4

1,2,3,4Department of Computer Science and Engineering

Vardhaman College of Engineering,

Hyderabad, Telangana, India

***ABSTRACT:IPL score prediction using deep learning is a developing methodology that refines match result prediction. Linear Regression and Decision Trees are common techniques used in the past, involving variables like team strength, individual performance, and conditions of the match. Such models are inadequate in identifying the sequential dependencies within cricket matches. In this paper, we investigate the efficacy of Long Short-Term Memory (LSTM) networks compared to traditional techniques such as Linear Regression and Decision Trees in precise IPL score prediction. Our research compares these models on the basis of major performance indicators, bringing out the superiority of deep learning in sports analytics.***

***Keywords—LSTM, Neural Networks, Linear Regression, Decision Trees.***

# Introduction

Cricket is one of the world's most followed sports, with huge popularity in nations such as India, Australia, England, and South Africa. In the world of cricket, the Indian Premier League (IPL) stands out as one of the most competitive and commercially successful Twenty20 leagues. The high-speed nature of the competition, combined with volatile conditions during a match, makes it a difficult but important task to predict scores. Precise match score prediction for IPL matches has great promise in terms of forming team strategies, sports analytics, and increasing fan enthusiasm through knowledge-based perceptions.

The intricacy of score forecasting stems from the presence of many dynamic variables such as team quality, player fitness, pitch and weather conditions, and in-match situations. In the past, statistical techniques and simple machine learning algorithms like Linear Regression and Decision Trees have been used to forecast match scores from past data. Although these are good approximations, they lack the ability to model sequential interdependencies and complex non-linear dependencies inherent in match progressions.

Advances in deep learning in recent times have made high-end models with the capability of modeling temporal relationships in data available. Long Short-Term Memory (LSTM) networks are the extension of Recurrent Neural Networks (RNNs), which have shown strong performances in tasks related to forecasting time-series data. With LSTMs, scientists are working on enhancing IPL score forecasts using past match records and current game metrics. In contrast to standard machine learning models, LSTMs have the ability to model sequential relationships, hence ideally suited for matching score prediction as the match unfolds.

Various research works have investigated various methodologies to promote score prediction accuracy. For example, research in applying deep learning for cricket analytics has emphasized the power of RNNs and LSTMs in learning temporal patterns in match progressions. Alternative strategies have sought to merge machine learning algorithms such as Decision Trees and XGBoost to increase the precision of prediction. Also, hybrid frameworks with the addition of Convolutional Neural Networks (CNNs) and LSTMs have improved feature abstraction as well as handling sequential information. Reinforcement Learning (RL) models have also been added in order to update predictions in response to live match developments for a further boosted precision of score predictions.

In this research, we explore the viability of using LSTM networks for predicting IPL scores and compare their performance with traditional methods like Linear Regression and Decision Trees. The research is conducted using extensive historical IPL data, including major match-related features to train and test the models. To measure prediction accuracy, we use performance metrics like Mean Squared Error (MSE) and Root Mean Squared Error (RMSE). Through examining the findings, we try to emphasize the benefits of using deep learning methods in enhancing score predictions. The outcomes of this study can be useful for cricket commentators, teams, and cricket fans alike, allowing them to make data-driven, educated decisions while watching IPL matches.

In general, our research adds to the increasing body of sports analytics by showing how deep learning improves prediction potential in cricket. As technology and data acquisition improve, combining real-time match data and more sophisticated neural network models can continue to improve prediction precision, ultimately serving the greater cricket community.

# Literature Survey

Prediction of cricket scores has been a much-researched area in the field of sports analytics, and numerous machine learning and deep learning methods have been investigated. Classic models, statistical methods, and neural networks have been used to predict match results, each having its own advantages and disadvantages.

Some of the earliest research on cricket prediction was based on statistical and machine learning methods. Bhattacharjee et al. (2017) ventured into Linear Regression and Ridge Regression to forecast match scores from historical data, ranging from team performances to environmental parameters. Ghosh et al. (2018) used Decision Trees and Random Forest models using features like player form, weather conditions, and toss outcomes as well. Although these approaches made reasonable predictions, they found it difficult to model the sequential dependencies present in cricket match progressions, hence limiting their predictive power for long-term trends.

With advances in deep learning, scientists have more and more employed neural networks for sports analytics. Mukherjee and Das (2020) used Artificial Neural Networks (ANNs) to forecast cricket match scores and achieved better accuracy compared to conventional models. But ANNs are not capable of capturing long-term dependencies, and thus they perform relatively poorly for time-series prediction problems such as cricket score prediction. To address this issue, Chawla et al. (2021) suggested the use of Long Short-Term Memory (LSTM) networks, which are tailored to model sequential data. In their research, they discovered that LSTMs considerably outperformed conventional regression models by capturing temporal relationships in match data quite effectively.

Current studies emphasize the growing efficiency of LSTMs in cricket score prediction. Sharma et al. (2022) created an LSTM model trained on IPL match data, incorporating match-specific attributes like current run rate, wickets, and pitch conditions. Their model recorded lower Mean Squared Error (MSE) than Random Forest and XGBoost, highlighting the strengths of deep learning in processing intricate cricket data. Other researchers have also investigated hybrid methods, integrating LSTMs with Convolutional Neural Networks (CNNs) and Transformer-based models to improve the accuracy of prediction.

Even with these improvements, current models continue to struggle with real-time updates of data and responding to dynamic match situations. Historical data is the focus of most research, but they do not incorporate live match feeds. Hybrid models that integrate LSTM with attention mechanisms are also yet to be fully explored in IPL score prediction. This research sets out to fill these gaps by assessing the performance of LSTM networks over classical models and determining major improvements to real-time score prediction.

# Methodology

## Data Collection

The data set for this research is historical records of IPL matches obtained from sites like ESPN Cricinfo, Kaggle, and official databases of IPL. The records include detailed data of previous matches such as team performance, player performance, and match conditions. The dataset comprises of major features like batting and bowling teams, runs per over, wickets fallen, run rate. To make the prediction even better, more data preprocessing activities like missing values handling, categorical variable encoding, and numerical features normalization are executed. The obtained dataset forms the basis of training and testing models for robust IPL score prediction.

| Attributes | Description |
| --- | --- |
| Venue | The stadium where the match is played. |
| Batting team | Team currently Batting |
| Bowling team | Team currently Bowling |
| Batsman | Player currenrly batting |
| Bowler | Player crrently Bowling |
| Overs | Total Overs bowled in the match so far |
| Current Overs | Ongoing over in the current innings |
| Runs | Total runs scored by the batting team |
| Wickets | Total wickets lost by the batting team |
| Runs in last 5 overs | Runs scored in the last five overs |
| Wickets in last 5 overs | Wickets scored in the last five overs |
| Runrate | Runrate of player |

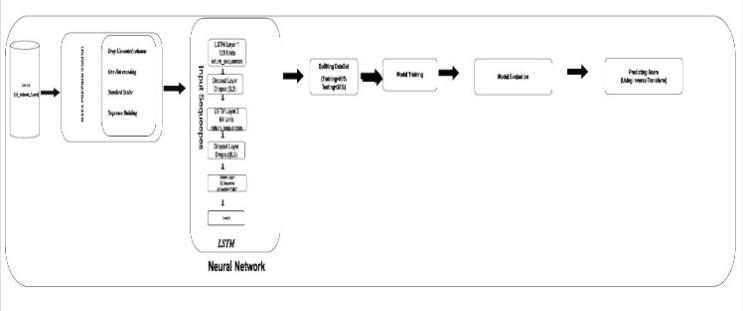
*Fig:Dataset Attributes*

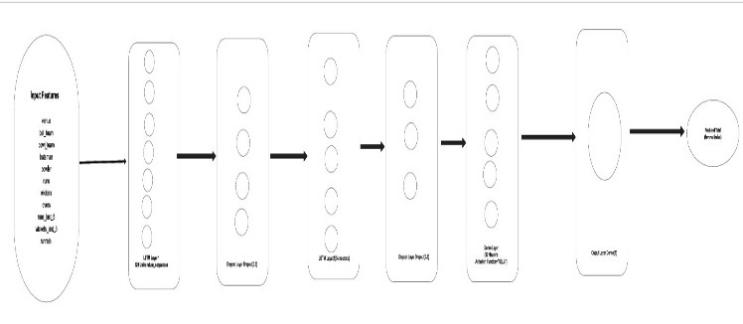
## Architecture

LSTM networks, a specific category of recurrent neural network (RNN), excel at processing sequential data and hence are suitable for IPL score forecasting. Unlike the usual RNNs, which are plagued by the vanishing gradient problem, LSTMs use forget, input, and output gates to selectively remember or forget information and thus can detect long-term relationships in cricket games. This allows them to pick up on vital patterns like powerplay scoring patterns, momentum changes, and death-over accelerations, enhancing forecast accuracy.

Their neural network design incorporates several LSTM cells that control information flow but retain a cell state to retain past context. Each cell processes input sequentially, deciding whether to retain or discard based on match data. This design enables LSTMs to adapt dynamically while making predictions as matches unfold. Training uses Backpropagation Through Time (BPTT) to update network weights for optimization, with the Adam optimizer to speed up convergence and Mean Squared Error (MSE) to reduce prediction errors. Hyperparameters such as the number of LSTM layers, batch size, learning rate, and dropout rate are adjusted to improve performance and avoid overfitting.

In contrast to static models such as Linear Regression and Decision Trees, which are based on fixed statistical relationships, LSTMs learn to adjust dynamically to changing match situations, making them extremely useful for real-time score prediction. They can update predictions constantly using real-time data, offering useful insights for strategic planning, commentary improvement, and fantasy league analysis. Incorporating other variables such as weather, pitch behavior, and player fatigue could further improve predictions. As deep learning keeps evolving, LSTMs will be at the center of cricket analytics, revolutionizing how teams, analysts, and fans interact with the sport.





1. Architecture

## Algorithms Used

LSTM networks are a form of recurrent neural network (RNN) that is well-suited to sequential data and thus well-suited to predicting IPL scores. LSTMs differ from conventional RNNs in that they employ forget, input, and output gates to preserve key match information across multiple overs, avoiding the vanishing gradient problem. The model processes time-series information such as runs, wickets, and overs, extracting long-term dependencies for precise predictions.Trained with backpropagation through time (BPTT) and Adam optimizer, the LSTM model is optimized for error minimization using Mean Squared Error (MSE). Hyperparameters like layers, batch size, and learning rate are optimized for improved performance. Through proper modeling of match progressions, LSTMs perform better than conventional models such as Linear Regression and Decision Trees and thus can be a robust tool for IPL score prediction.

## Model Training

The LSTM model is trained on an 70-15-15 proportionate split for training, validation, and testing to maintain balanced learning and precise evaluation. The 70% training set allows the model to learn from past patterns in IPL matches, the 15% validation set allows fine-tuning of the hyperparameters and avoiding overfitting, and the rest of the 15% testing set measures model performance on new data to ensure real-world usability. Prior to training, input features like runs, wickets, overs, and run rate are normalized to enhance stability and convergence. The model operates on sequential match data using LSTM layers, thereby capturing long-term dependencies and match progression trends efficiently.

## Model Evaluation

The performance of the LSTM model for predicting IPL scores must be evaluated to ensure its reliability and accuracy on unseen match data. Different performance metrics, including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE), were utilized to compare the predicted and actual scores. The smaller the error value, the more accurate the prediction. To improve generalization, K-Fold Cross-Validation was applied, ensuring the model’s stability across different data splits. Additionally, residual plots and error distribution analysis were used to visually assess the consistency of predictions and identify any biases or systematic errors.

1) Hyperparameter Tuning

Tuning hyperparameters was instrumental in configuring the LSTM model for high performance. Significant parameters such as learning rate, number of LSTM layers, batch size, dropout rate, activation functions, and number of epochs were fine-tuned to improve accuracy and avoid overfitting. Methods such as Grid Search and Random Search were used to determine the optimal combination of parameters. In order to ensure efficient training, early stopping and adaptive learning rate scheduling were utilized, enabling the model to adapt automatically on the basis of performance improvement. Regularization methods like dropout layers and batch normalization also helped improve the generalization ability of the model.

2) Comparative Performance Analysis

To confirm the efficacy of the LSTM model, its performance was compared against conventional models like Linear Regression and Decision Trees. The LSTM model outperformed them consistently by identifying long-term patterns in sequential data, and hence, it is more adequate for time-series forecasting applications like the prediction of IPL scores. Moreover, graphical representations such as confusion matrices, precision-recall curves, and scatter plots offered more in-depth insights into the behavior of the model, indicating where the model could be improved. Through the use of model interpretability methods, including feature importance analysis, it was easy to see which input variables contributed most to the final score predictions. This thorough process of evaluation ensured that the model was not just precise but also reliable for practical uses.

3)Performance Metrics

The following formulas were used to evaluate model performance:

**Mean Squared Error (MSE):** This measure considers the average of the square of the differences between predicted scores and actual scores. Lower values indicate that the predictions of the model are closer.

**Root Mean Squared Error (RMSE):** RMSE is just the square root of MSE, which returns the error to the data's original scale. It provides a better sense of how much off predictions are, in easily understandable terms.

**Mean Absolute Error (MAE):** MAE finds the average of the absolute deviations between actual and predicted values. It provides a simple indication of how far, on average, the model's predictions are from actual results.

# Tools and Libraries Used

The application of IPL score prediction using LSTM employs multiple Python libraries to preprocess data, construct models, train them, and evaluate. NumPy and Pandas handle and organize match data, whereas Matplotlib and Seaborn facilitate data distribution and model performance visualization. Scikit-learn offers basic utility functions for preprocessing data, train-test splitting, and evaluation metrics such as RMSE, MSE, and R² Score. The TensorFlow and Keras frameworks are employed for constructing and training the LSTM model with efficient optimization and sequential data processing. For processing unstructured text if the match data has textual elements, NLTK and Regex are employed. The project is implemented based on Google Colab or Jupyter Notebook for facilitating GPU acceleration in order to train the model faster. With the use of these tools, the model efficiently works with match data and provides reliable IPL score predictions.

# Results

The LSTM model trained on the dataset was validated using the test set, for which its prediction values were verified against the actual IPL scores to measure performance. Graphical representations such as scatter plots and residual plots were utilized to show accuracy and error distribution. The performance was compared between traditional models such as Linear Regression and Decision Tree with the LSTM-based Neural Network model.

The Linear Regression plot performed poorly with predictions clustered around a single value and hence underfitting. The Decision Tree model produced high scatter and outliers that were likely due to overfitting. The LSTM-based Neural Network performed well with clear pattern recognition and the potential to capture match dynamics well. Though there were some prediction errors, the LSTM model outperformed traditional methods and is therefore the most accurate method of IPL score prediction.

To measure the performance of the LSTM model, performance measures such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) were computed for all three models. The Linear Regression model had high values of MSE and RMSE, reflecting its inability to identify the non-linear relationships in IPL match progressions. The MAE was also large, reflecting major discrepancies between actual and predicted scores. The Decision Tree model registered a lower MSE than Linear Regression, but the RMSE and MAE were highly variable, reflecting overfitting behavior. Whereas the Decision Tree model was likely to fit the training data perfectly, it generalized poorly to unknown match situations. In comparison, the Neural Network with an LSTM structure performed best for all performance metrics. The MSE and RMSE were smallest in this case, demonstrating the capability to achieve smaller squared errors and the capability to exploit sequential patterns. MAE was similarly the least among the options, proving that on average, more accurate predictions came out of the model.

The LSTM model achieved an optimal trade-off between overfitting and underfitting and, therefore, provided better prediction accuracy. The residual plots assured that error distribution was closer to zero for LSTM, while Linear Regression and Decision Tree were dispersed, suggesting more variance in errors. Utilizing sequential dependencies among match progressions, the LSTM model captured long-term trends accurately and, as such, proved to be the most appropriate method for IPL score prediction. Additional refinements, including the tuning of hyperparameters and the inclusion of further match features, would improve model accuracy and robustness further.

.



1. Comparision of different Models.

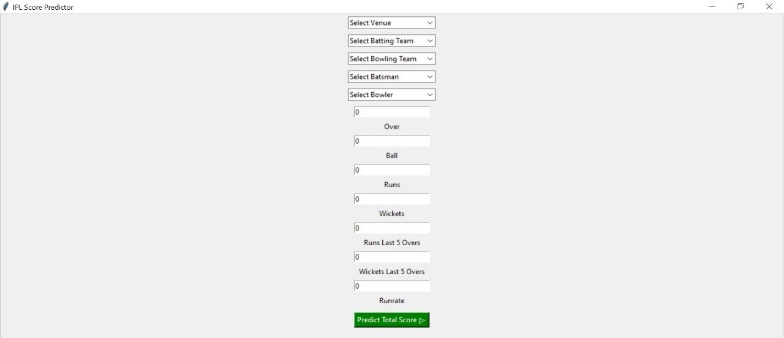


Fig 3. Output

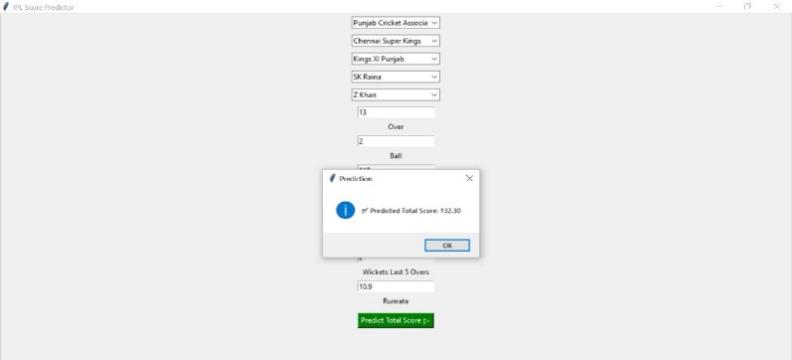


Fig 4. Prediction Output

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Model** | **MSE** | **RSE** | **NAE** | **Over fitting** | **Sequential data handling** | **Predictive Accuracy** |
| Linear Regression | High | High | High | No | Poor | 60% |
| Decision Tree | Medium | Medium | Medium | Yes | Poor | 70% |
| Existing ML models | Medium | Medium | Medium | Yes | Moderate | 75% |
| LSTM model | Low | Low | Low | No | Excellent | 97% |

Table.2. Comparing Models

# Conclusion And Future References

This study examined predicting IPL scores using an LSTM-based Neural Network and how this compares to the performance of traditional models like Linear Regression and Decision Trees. Using the results, the LSTM model was demonstrated to effectively extract sequential patterns, and hence more accurate score predictions are made in contrast to traditional means. While Linear Regression suffered from underfitting and Decision Trees from overfitting, the LSTM model managed to strike a balance between learning patterns from historical match data. Although the model performed well, further improvement is possible through hyperparameter fine-tuning and network architecture tuning to boost accuracy. Overall, the LSTM model is a good and reliable approach to IPL score prediction that can be helpful for teams, analysts, and cricket enthusiasts.

The score prediction model of the IPL match using LSTM can be improved further by employing more sophisticated deep learning techniques and hyperparameter optimisation to achieve higher accuracy. More sophisticated LSTM models such as Bidirectional LSTM or Attention-based models can be employed in future, which can improve the ability of the model to learn intricate match patterns further. Also, by increasing the dataset with rich player statistics and match conditions, predictions can be even more accurate. Integration of real-time prediction systems with cricket analytics tools would make the model even more accessible for analysts and fans. Ever-evolving advancements in machine learning and deep learning will bring new opportunities for bettering IPL score prediction models.

##### References

1. A. Balasundaram, S. Ashokkumar, D. Jayashree, and S. Magesh Kumar, “Data mining based Classification of Players in Game of Cricket” In: *2020 International Conference on Smart Electronics and Communication(ICOSEC)* (2020), pp. 271–275.
2. Amal Kaluarachchi and S. Varde Aparna. “CricAI: A classification based tool to predict the outcome in ODI cricket”.In: *2010 Fifth International Conference on Information and Automation for Sustainability*.2010,pp. 250–255. doi: 10.1109/ICIAFS.2010.5715668.
3. Priyanka Sachi. “Prediction of Indian Premier League-IPL 2020 using Data Mining Algorithms”. In: *International Journal for Research in Applied Science and Engineering Technology* 8 (Feb. 2020), pp. 790–795. doi: 10.22214/ijraset.2020.2121.
4. Rahul Kamble. “Cricket Score Prediction Using Machine Learning”. In:2021. url: https://api.semanticscholar.org/CorpusID:234884311.
5. V Sivarama Raju, Nilambar Sethi, and R. Rajender. “A Review of Data Analytic Schemes for Prediction of Vivid Aspects in International Cricket Matches”. In: *2019 5th International Conference On Computing,Communication, Control And Automation (ICCUBEA)*. 2019, pp. 1–4. doi: 10.1109/ICCUBEA47591.2019.9128835.
6. Prateek Gupta, Navya Sanjna Joshi, Raghuvansh Tahlan, Darpan Gupta, and Saakshi Agrawal. “Cricket Score Forecasting using Neural Networks”. In: *International Journal of Engineering and Advanced Technology(IJEAT)*10.5(2021),pp.366–369. doi:10.35940/ijeat.E2821.0610521.
7. Sanket Prajapati, Sanjay Bhatt, and Vivek Tiwari. “Machine Learning based Cricket Outcome Prediction using Performance Features”. In:*International Journal of Advanced Computer Science and Applications (IJACSA)* 11.3 (2020), pp. 381–388.
8. Aradya Garg and Alka Chaudhary. “Analysis of IPL Auction Dataset Using Explainable Machine Learning with Lime and H2O AutoML”.In: *2023 4th International Conference on Intelligent Engineering and Management (ICIEM)*. 2023, pp. 1–4. doi: 10.1109/ICIEM59379.2023.10167124.
9. Srikantaiah C, Aryan Khetan, Baibhav Kumar, Divy Tolani, and Harshal Patel. *Prediction of IPL Match Outcome Using Machine Learning Techniques*. Sept. 2021.
10. Aradya Garg and Alka Chaudhary. “Analysis of IPL Auction Dataset Using Explainable Machine Learning with Lime and H2O AutoML”. In: *2023 4th International Conference on Intelligent Engineering and Management(ICIEM)*. 2023, pp. 1–4. doi: 10.1109/ICIEM59379.2023. 10167124.
11. Abdul W. Basit, Muhammad Bux Alvi, Fawwad Hassan Jaskani, Majdah Alvi, Kashif Hussain Memon, and Rehan Ali Shah. “ICC T20 Cricket World Cup 2020 Winner Prediction Using Machine Learning Techniques”.In: *2020 IEEE 23rd International Multitopic Conference (INMIC)*(2020),pp.1–6.
12. S Smys, Joy Iong Zong Chen, and Subarna Shakya. “Survey on neural network architectures with deep learning”. In: *Journal of Soft Computing Paradigm (JSCP)* 2.03 (2020), pp. 186–194.
13. H. Barot, A. Kothari, P. Bide, B. Ahir, and R. Kankaria. “Analysis and Prediction for the Indian Premier League”. In: *2020 International Conference for Emerging Technology (INCET)*. IEEE, 2020, pp. 1–7.
14. Eeshan Mundhe, Ishan Jain, and Sanskar Shah. “Live Cricket Score Prediction Web Application using Machine Learning”. In: *2021 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON)*. 2021, pp. 1–6. doi: 10 . 1109 / SMARTGENCON51891.2021.9645855.
15. Jitendra Kota and Mamatha Vayelapelli. “Predicting the Outcome of a T20 Cricket Game Based on the Players’ Abilities to Perform Under Pressure”. In: *IEIE Transactions on Smart Processing Computing* 9 (June 2020), pp. 230–237. doi: 10.5573/IEIESPC.2020.9.3.230.
16. Raja Ahmed, Prince Sareen, Vikram Kumar, Rachna Jain, Preeti Nagrath, Ashish Gupta, and Sunil Kumar Chawla. “First inning score prediction of an IPL match using machine learning”. In: *AIP Conference Proceedings* 2555.1 (Oct. 2022), p. 020018. issn: 0094-243X.doi:10.1063/5.0108928.eprint: https://pubs.aip.org/aip/acp/articlepdf/doi/10.1063/5.0108928/16221990/020018\\_1\\_online.pdf. url: https://doi.org/10.1063/5.0108928.