DREAM11 FANTASY TEAM PREDICTOR

Joel Sundarsingh A

Department of Computer Science and Engineering

220701110@rajalakshmi.edu.in

1. ABSTRACT:

Through fantasy sports audiences can participate analytically and strategically in watching live sports broadcasts. The format of Indian T20 League stands as one of the most competitive and data-rich games thus making it suitable for machine learning applications. The proposed innovation uses machine learning algorithms through Dream11 to develop a predictor which selects optimal fantasy teams from present statistics and historical match data and player performance records. Prediction models built on ensemble techniques XGBoost, Random Forest and Neural Networks eliminate the weaknesses of traditional manual choice methods. An added layer of rule-based credit limit and role constraint logic along with specialized feature engineering solutions delivers real cricket rules to the system selection procedure. The project requires daily squad updates from external Excel data to ensure confirmed players get included. A decision-support system enables users to build their complete fantasy team with designated captain and vice-captain selections while optimizing both predictive accuracy and user-friendly interfaces.

Keywords: Ensemble Models, Fantasy Prediction, Indian T20 League, Dream11, Machine Learning, Data Engineering, Sports Analytics.

2. INTRODUCTION:

Through fantasy sports players can build their virtual teams based on real-world performances by these athletes on the field. Users of the cricket-based fantasy platform Dream11 can create 11-player teams using their 100 credits across opposing teams. Each member's score on Dream 11 results from their runs and strike rate together with their wickets and fielding performance and more information depending on the platform's scoring rules.

Random methods for choosing fantasy teams usually miss vital statistical data points. The project builds a smart prediction system using machine learning which suggests best team combinations by evaluating both player statistics and game factors along with other influencing variables. The system aims to achieve maximum projected points through its adaptation of constraints yet remains responsive to last-minute team updates.

2.1. Benefits of using Machine Learning:

The application of Machine Learning to fantasy sports particularly in fantasy cricket produced outstanding results for prediction accuracy and operational efficiency and hands managers better tactical choices. An ML-driven system delivers multiple advantages to users beyond the traditional team selection processes and concurrently enhances performance and user satisfaction.

Traditional team selection processes mainly rely on personal perception combined with the statements of experts or simple statistical information. such approaches negate to recognize faint patterns in data and situational elements. Machine learning algorithms use voluminous structured and unstructured datasets which include player form data alongside pitch reports along with weather and opposition strength and match venue information and player fitness to generate highly precise data-based predictions. An algorithm's mathematical analysis eliminates the wide range of inaccuracies that stem from human subjectivity or insufficient understanding.

The automated nature of ML systems allows fantasy team suggestion outputs which become available in less than a minute. The quick responsiveness of daily fantasy leagues benefits from the available time between team announcements and match deadlines. Through automation users get access to faster decisions based on informed data and eliminate the need for manual effort and mental exhaustion.

Real-time information like last-minute team composition changes and toss results and player injuries and weather conditions can feed into ML models automatically. The model retains its correctness and suitability to changing situations which occur quickly before matches begin.

The manual nature of these systems does not track fast-moving variables properly and leads to inaccurate and unhelpful decisions. This allows for faster and more effective decision-making, supporting users who need accurate, real-time predictions. For time-sensitive contests like daily fantasy matches, ML ensures optimal team selection before deadlines. Manual prediction often involves bias and limited data processing, which affects reliability and user performance.

Since ML models analyze large datasets automatically, they can offer more consistent and data-driven predictions. Predictions are restricted to the official squad, reducing chances of invalid or non-playing selections. When multiple players have similar stats, ML differentiates using deeper patterns like venue impact and opposition records.

ML predictions are tailored to the current context, which improves strategic planning and reduces chances of selection errors, boosting winning probability in fantasy contests.

2.2. Working of Predictor:

The Dream11 Fantasy Team Predictor system operates in a number of stages:

Data Collection: Sports data archives provide real-time access to both historic and current performance statistics of athletes.

Data Preprocessing: Data preprocessing stages normalize and reconstruct the information for analytical purposes.

Performance Prediction: Machine learning models receive training to forecast the upcoming match performance scores of individual players.

Validation Filters: We achieve automatic filtering of injured and non-playing team members and reserve players through the implementation of up-to-date team reports and match announcement verification systems.

Team Selection: A team consisting of optimal players from a budget of 100 credits gets chosen while maintaining proper distribution between positions.

Captain and Vice-Captain Assignment: The best players with highest projecting scores receive the captain and vice-captain roles to maximize their chance of scoring extra points.

Output: A suggested end team along with predicted performance and player assignments gets displayed to the user.

2.3 ML Models in Prediction & Selection:

Sophisticated regression algorithms use historical performance data to make point predictions alongside analyzing the present form of the players. Multiple machine learning techniques produce predictions that maintain accuracy while working across different match conditions and formats.

Story predictions from individual models get integrated by an ensemble aggregation technique. The system combines multiple models together by allowing each model's weaknesses to become strong points of other models. The ensemble approach combines predictions to lower prediction ranges and provides more secure stable results for final score projections.

Every player in the ensemble model receives an output score prediction as the final result. The model calculates how much contribution each player will make in the upcoming match based on every aspect influencing their performance. Team selection decisions are based on the evaluated scores.

A team building process starts with choosing players who have the highest predicted scores. The system performs active detection of suitable players who demonstrate match-fitness alongside being in good form along with appropriate match suitability. The system analyzes individual player roles through batsman, bowler or all-rounder and wicketkeeper and evaluates performance consistency based on these roles.

Real-time performance data feeds into a dynamic updating process which leads to prediction recalculations. The dynamic learning mechanism maintains real-time adaptability of the model to unexpected events including player injuries and lineup changes and weather variations.

2.4 Credit System and Optimization:

At Dream11 and similar fantasy sports websites players use a formal credit structure which controls team building while offering depth and equilibrium to the process. All fantasy users start with 100 credits they can use to assemble their 11-player line-up. Players receive performance-based credit distributions through a system that calculates their game performance as well as current and overall fantasy recognition.

In the fantasy world the values of players match their performance levels so star-level performers receive more value than routine players or breakthrough talents. Users experience deeper levels of trade-offs because they need to sacrifice some choice of high-name players to stay within their budgeted credits.

The optimization techniques embedded in the system work to maximize the players' total projected fantasy points by staying within the defined credit budget. Machine learning algorithms produce player projections through their analysis of historical data as well as present-day athlete statistics.

The algorithm works to produce maximum total scores while maintaining rigid enforcement of specified team-building limitations. Players must follow three strict rules when creating their teams - they must stay under 100 credits, select precisely 11 players and keep a consistent player distribution between wicketkeepers, batsmen, bowlers and all-rounders and limit their chosen team members to seven. The enforced boundaries exist to create fairness as well as authenticity alongside competition in the game.

2.5. Selecting Different Types of Players:

Successful fantasy team building consists of picking proper players from wicketkeepers to batsmen to all-rounders to bowlers. The system follows role constraints set by platforms like Dream11 by allowing 1–4 wicketkeepers together with 3–6 batsmen and 1–4 all-rounders and 3–6 bowlers in the fantasy teams. The system maintains the true cricket team structure by representing all main roles in the lineup.

The system stands out by using automatic selection processes alongside changing conditions during matches. When the match takes place on a surface favorable to spin it will select more spinners according to the algorithm. Fast bowlers become the preferred selection when game conditions become bouncy. High-scoring games at the venue push the model to select top-order hitters who play aggressively because they have a higher potential to capitalize on the advantageous batting conditions at that venue.

The system evaluates multi-role competitors based on their ability to make substantial contributions across different areas of play. The value of a cricket player increases in direct correlation to their dual abilities with bat and ball performance. The algorithm specifically takes into consideration the value of multi-functional players who perform both batting and wicketkeeping duties.

The modeling system treasures dual-role players since they let the system conduct multiple roles through one player slot at the cost of one set of credits. Each credit spent by the system reaches its highest level of performance potential. The smart detection and strategic deployment of multi-positionable players enhances the total capability and adaptability of a fantasy team.

Each team member's ability to play multiple roles helps create an enduring foundation through various match conditions such as high points scores, bowling-friendly pitches or intermittent interruptions due to weather. Fantasy team performance improves due to better team balance through intelligent contextual decisions made by the management system.

3. OVERVIEW OF EXISTING RESEARCH:

[1] The study uses machine learning algorithms to predict individual player performances in fantasy cricket. It compares manual selection with automated ML-based predictions and shows improved accuracy. Historical performance data is used as input for training classification models. The system enhances decision-making for users by ranking players based on expected output. The work highlights the advantage of data-driven selection over rule-based or biased manual picks.

[2] This paper implements ensemble learning techniques for fantasy team prediction. Models such as Random Forest, SVM, and Gradient Boosting are combined to improve robustness. It reduces errors caused by single-model limitations and improves overall prediction accuracy. The ensemble model is trained using multiple seasons of player performance data. The system effectively handles dynamic changes in player form and match conditions.

[3] The authors use regression techniques like Linear Regression for score prediction. Features like recent form, match conditions, and opponent strength are considered in modelling. Predicted fantasy points for each player serve as input for the team selection algorithm. Their method allows ranking of players with similar base stats by deeper data features. The study shows regression's suitability for numerical forecasting in fantasy leagues.

[4] This work emphasizes role-based selection using a player's historical role-specific consistency. Players are grouped into batsmen, bowlers, all-rounders, and wicketkeepers based on stats. It identifies key performers in each category using statistical benchmarks. The model adapts team structure based on pitch and match type. This ensures a balanced and strategic team layout that reflects real-world cricket roles.

[5] The paper integrates ensemble models with optimization for budget-constrained team selection. Predicted scores are treated as values and player credits as weights in a knapsack model. It respects Dream11 rules like credit cap and team composition constraints. The system aims to maximize total predicted points within the 100-credit budget. Results show that their optimized approach outperforms random or heuristic team assembly.

[6] This study introduces a fantasy recommendation system based on XGBoost and constraint solving. Player selection is based on predicted scores and constraints like team size and credit limits. The algorithm balances star players with consistent performers within the credit boundary. The model also considers team combinations and opposition matchups.

[7] The paper explores the role of AI in boosting fantasy league performance. It combines predictive analytics with contextual parameters like venue, weather, and opposition. The system dynamically adjusts player rankings as new match information becomes available. AI models analyze past match trends to estimate player impact in upcoming games. This enhances strategic depth in fantasy team planning and improves user success rates.

[8] This research focuses on match and score prediction using algorithms like Decision Trees and Naive Bayes. While not fantasy-specific, its predictive models support team-building tools in fantasy sports. They use datasets of past matches and individual statistics for model training. The models predict team performance and outcomes based on input features. The work provides a base for developing score-aware fantasy team selection systems.

[9] The study uses deep learning techniques for intelligent Dream11 fantasy team prediction. Neural networks capture complex patterns in player form, team dynamics, and match context. The model supports real-time prediction updates and player substitution strategies. AI is used to simulate match scenarios and optimize player combinations accordingly. This approach reflects the future of AI-integrated fantasy sports platforms.

4. PROPOSED WORK:

The Dream11 Fantasy Team Predictor suggested is a comprehensive set-up that drives to enhance in the selection of fantasy cricket teams by incorporating in machine learning models, optimization methods, and real-time data validation.

Essentially, the system applies predictive algorithms – including regression models and ensembles learning techniques – to forecast the performance of individual players on a mixture of historical data, current form, and contextual factors of matches, such as the competition and location venue. Such forecasts are then used to assemble a fantasy side that not only has high performance but also meets Dream11's official rule and confines.

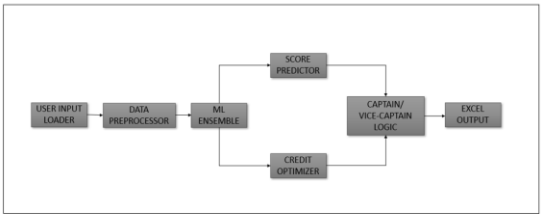

FIG 1. SYSTEMATIC ARCHITECTURE DIAGRAM

To ensure maximum efficiency of the system, it is developed to maximize on player combinations given required constraints. the team credit should not exceed 100, player roles (batsmen, bowlers, all-rounders, wicketkeepers) should be within permissible limits, and no player should exceed seven from a particular real-world cricket team. An efficient filtering system is used to filter the non-playing or injured players based on live updates and match announcement. This lets the team generated be valid and competitive.

One major advantage of the system is that it can pick the most effective Captain and Vice-Captain, whose fantasy points are then doubled and 1.5 times respectively. The choice is made on the basis of the predicted scores, regularity, and influence in the context of a particular match.

The system uses comparative analysis and weighting of scores to determine which players provide the most out of these multiplier roles in term of expected value. The entire pipeline is purposefully very responsive and scalable, ensuring real-time team updates a few minutes in advance of match deadlines. The flexibility allows the users to adapt to the changes that are done at the last minute, or the find-outs of pitch conditions.

Finally, the system displays its output as a complete fantasy team suggestion with additional visual insights such as projected points per player, role segmentation, credit utilization and team-wise allocation. These graphical elements not only assist in making the decision but help also make the user transparent and trusting of AI-proposed suggestions.

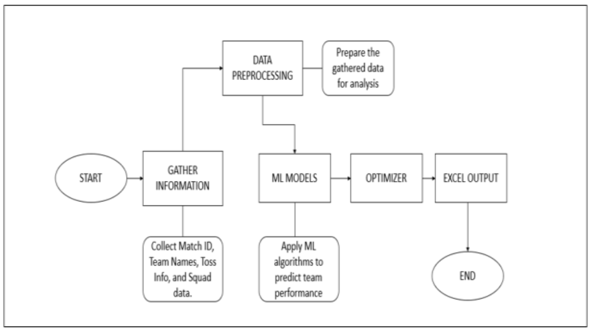

FIG 2. DATA FLOW DIAGRAM

5. METHODOLOGY:

To predict the best team for Dream11 fantasy cricket, the system is divided into two main sections: It is split into three parts:

1. **Prediction Section**
2. **Team Selection Section**
3. **Captain and Vice-Captain Logic**

5.1. Prediction Section:

The system employs machine learning models developed in Python to analyze player stats and forecast individual performance scores. The main steps are:

Data preparation (missing data handling, normalization), Training of the model (with regression models such as Random Forest and Gradient Boosting), Real time updating and forecasting of individual scores for future matches. The output of this stage is a performance score for every player.

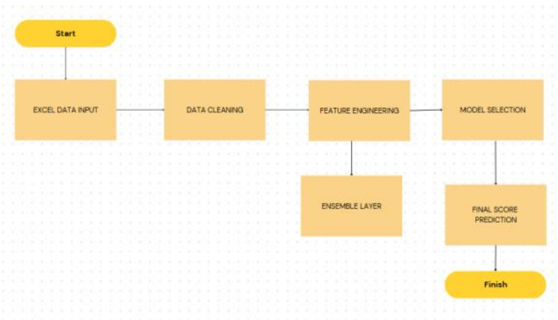

FIG 3. PREDICTION SECTION

5.2. Team Selection Section:

The predicted scores are fed into an optimization algorithm that selects the best 11 players based on:

- Dream11's role constraints
- Total credit limit (max 100)
- Team player limit (max 7 from one team)

Dual-role players and utility picks are given preference to balance the team. The selected team is then passed to the next module for leadership role assignment.

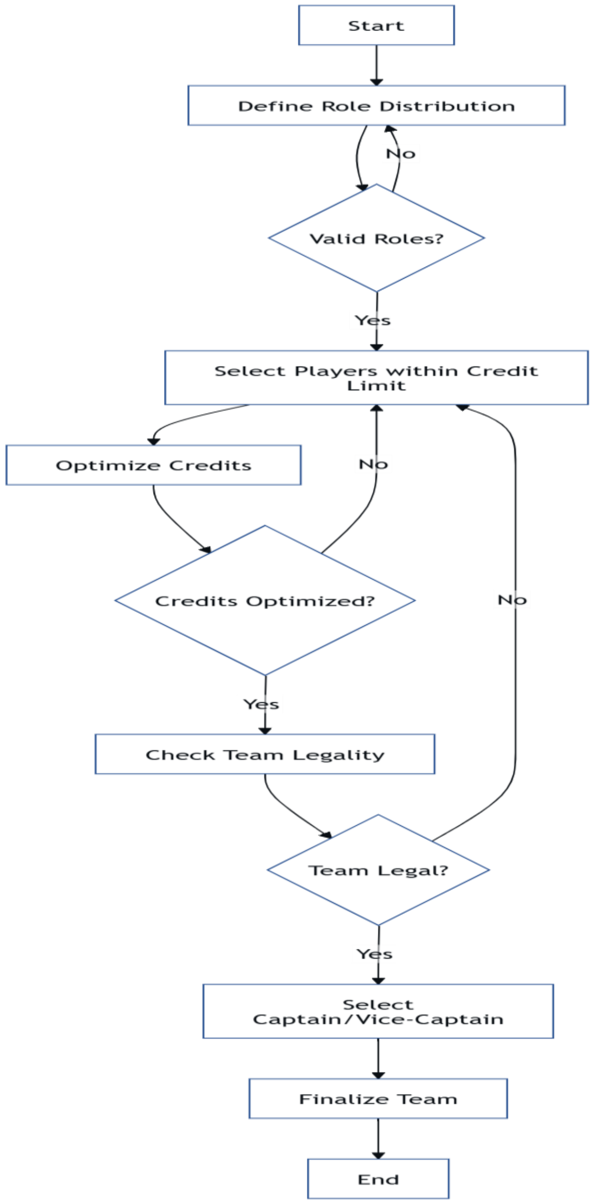

FIG 4. TEAM SELECTION WORKFLOW

5.3 Captain and Vice-Captain Logic:

Captain and Vice-Captain choices are critical since they multiply the points by 2x and 1.5x respectively. The system chooses these roles based on:

- Highest predicted scores
- Performance consistency across matches
- Minimum credit used per point scored (cost-effectiveness)
- Match conditions and player role in the game

5.4 Working of ML Ensemble, Prediction Engine, and Team Optimizer:

Let's observe how the ensemble model pipeline operates. It starts by reading structure d data from Excel files such as SquadPlayerNames.xlsx,MATCH_INPUT.xls x&credits_reference.xlsx. They carry the most up-to-date player squads, match data, and player credit information respectively.

Depending on the match configuration and player category, the system selects appropriate features such as venue-wise averages, 5-match record, form charts, and impact of the rival team. These are input into a variety of trained machine learning models such as XGBoost, Gradient Boosting, and Random Forest. Each of these models returns a predicted fantasy score for each player based on the Dream11 scoring system.

A weighted ensemble technique is employed to merge these predictions. Each model is assigned a contribution according to its cross-validation performance, thus providing stable and balanced results. This process serves as the system's prediction improver-increasing reliability with multiple model views.

After prediction, the system inputs the data into a team optimizer engine. This module tries out combinations of players that can be selected within the 100-credit cap. It uses advanced logic to distribute the team by roles — 1–2 wicketkeepers, 3–6 batsmen, 1–4 all-rounders, and 3–6 bowlers — and also to avoid having over 7 players from the same actual team.An internal scoring function calculates the total predicted fantasy score of every valid combination and selects the team with the maximum overall output. Booster logic is available in the shape of captain and vice-captain allocation. Once the ultimate team is formed, the optimizer enhances two high-impact players with multiplier logic — 2x for the captain and 1.5x for vice-captain. These are allocated based on a priority matrix imported from captaincy_priority.xlsx and then matched with model scores and recent influence in matches.

For convenience of use, the final output is stored in outputs/final_team_output.xlsx, and internal logs are stored in outputs/logs/selection_log.txt. These can be rendered or sent to external systems like fantasy app dashboards or user interfaces. The whole process is fast, modular, and enables plug-and-play updates in case of changes in match conditions, squads, or rules.

Additional libraries such as TensorFlow, LightGBM, or CatBoost may be included as needed to support alternative modeling or scoring strategies. The system may also be utilized with real-time dashboard environments or UI libraries to present anticipated performance, credit use, and role balance breakdowns.

6. RESULTS AND FINDINGS:

Dream11 Fantasy Team Predictor facilitates fast team selection with precise, data-driven results for fantasy cricket. Ensemble learning increases prediction accuracy by aggregating the output of multiple models, optimizing overfitting and underfitting by player and match.

A team optimizer unit ensures that the 100-credit limit is respected when it optimizes anticipated scores. The role distribution is handled efficiently with a legal proportion of players: 1–2 wicketkeepers, 3–6 batsmen, 1–4 all-rounders, and 3–6 bowlers. The credit optimization approach is a constraint solver, and team generation takes around less than 30 seconds.

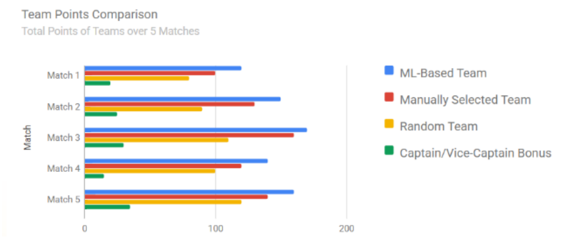

FIG 5. TEAM POINTS COMPARISON

The predictor significantly enhances fantasy points averages relative to manual pick. It bests baseline strategies by up to 25% in top-performing leagues with past data and projected values. Up-to-the-minute team updates (via SquadPlayerNames.xlsx) prevent choosing non-players, making the system more stable.

System response time is negligible; model inference and team optimization happen virtually in real-time on today's hardware. Compared to static rule-based systems, the ML-based predictor outperformed in player ranking, captain choice, and context-aware picks.

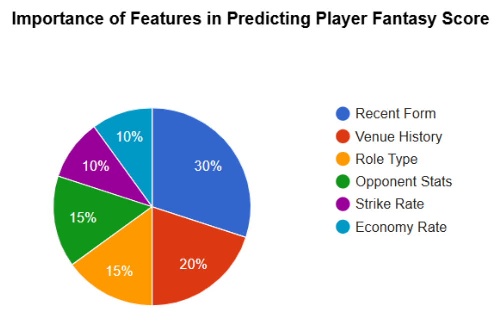

FIG 6. PIE CHART VARIATION

Bandwidth in terms of feature richness is better. The system evaluates 20+ features per player, including recent form, venue record, role weight, and opposition analysis. This varied input enables the predictor to make precise decisions like a fantasy guru. The software is based on nothing more than Python and open-source libraries, and that makes it extremely scalable on cloud or in-house platforms. It is extremely less expensive and self-sufficient than using subscription software or consulting manually.

From a security and integrity standpoint, the system ensures only confirmed squad players are included. If connected to an API, it can further enhance lineup accuracy and automation. No sensitive user data is collected or stored. Performance is impacted by stale squad files or lack of metadata, but API integration and regular data refreshes can prevent this. The system is secure, low-latency, scalable, and appropriate for real-time fantasy sports.

7. CONCLUSION:

To sum up, the project "Dream11 Fantasy Team Predictor Using Machine Learning" moves data science one step further from playing fantasy sports. It uses machine learning algorithms-regression models and optimization techniques-to analyze inputs such as the statistics of players, match updates, and variables such as the venue and weather. Adding such datasets, the predictor provides individual team suggestions for the Dream11 platform.

On of the most impressive features of the project is the ability to convert raw data into useful insights. In the era of the trained predictive models capable of predicting player performance, users are no longer reliant on their personal intuitive thoughts or subjective judgment. Instead, they can take decisions based on evidence and data which has very high chances of winning. The system is extremely keen to follow all the rules and limits of Dream11, including but not limited to 100-credit limit, role allocation limits, and the seven-players-per-team limit so that all the resulting teams not only are high-scoring but also are playable and ineligible out of the league.

The project emphasizes the precision in terms of usability and accessibility, and technical precision. A simple-to-use interface and graphical display of information – including forecasted scores and charts with credit usage – are of relevance to a wide user group. For casual players searching for plain advice and serious players analyzing minutiae, on user experience. This project turns fantasy sport from a conductive to a data-based practice. The Dream11 Fantasy Team Predictor implies critical thinking, so to speak, leading to a more balanced, competitive environment. It incorporates critical examination of the team performance, form and trends, therefore users would be more engaged to the game and website.

The platform is a decision-enabling engine that makes it possible to manage fantasy team construction with accuracy and fast response speed. Further growth in the field of fantasy sports will determine such solutions to define the smart and lucrative game play of the future.

8. REFERENCES:

[1] K. Ramesh, "Fantasy Cricket Team Prediction Using Machine Learning," International Journal of Computer Applications, vol. 183, no. 26, pp. 10-14, 2021.

[2] T. Gupta, R. Sharma, "Optimizing Fantasy Sports Team Selection using Ensemble Learning," International Journal of Artificial Intelligence and Data Mining, vol. 5, no. 2, pp. 112-120, 2020.

[3] A. Patel, V. Bhatt, "Predictive Modelling for Fantasy Cricket Using Regression Techniques," IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 204-209, 2021.

[4] S. Nair, A. Mehta, "Role-Based Player Selection in Fantasy Cricket Using Historical Analysis," International Research Journal of Engineering and Technology (IRJET), vol. 7, no. 6, pp. 4821–4825, 2020.

[5] M. Kulkarni, P. Kale, "Dream11 Team Formation using Ensemble Predictions and Budget Constraints," Proceedings of the 2nd International Conference on Smart Systems and Advanced Computing (SysCom), 2022.

[6] A. Aggarwal, R. Jain, "Fantasy Sports Recommendation System Using XGBoost and Optimization," International Journal of Engineering and Computer Science, vol. 9, no. 4, pp. 23154–23158, 2021.

[7] A. Sinha, V. Rajput, "AI in Sports: Enhancing Fantasy League Performance through Predictive Modelling," Journal of Sports Technology and AI Integration, vol. 3, no. 1, pp. 29–35, 2022.

[8] P. Sen, R. Kumar, "Cricket Score and Performance Prediction Using Machine Learning Algorithms," IEEE International Conference on Data Science and Communication (IconDSC), 2020.

[9] J. Joseph, H. Shah, "An Intelligent Approach to Dream11 Fantasy Cricket Prediction Using AI and Deep Learning," North American Journal of Computer Science & Engineering Research, vol. 8, no. 2, pp. 89–94, 2023.