Next-Gen Team Builder Using Predictive AI

Team 67

Inter IIT Tech Meet 13.0

Contents

1 PROBLEM STATEMENT

The goal of this project is to develop a machine learning-based solution that helps users create optimized, winning fantasy cricket teams on Dream11 by leveraging historical player data, match conditions, and other relevant features. The project involves building a robust machine learning model to predict **best dream points**, along with providing detailed feature explainability to make the recommendations transparent. Additionally, the solution will feature an intuitive user interface (UI) with **Generative AI**, offering interactive audio and video guides to assist users in team-building, game predictions, and strategic team formation suggestions.

1.1 UI Requirements and User Categorization

To meet the diverse needs of users, two distinct user interfaces need to be developed:

1. **Model UI (for Data Scientists and Professionals):** The Model UI is designed for data scientists and professionals who are interested in analyzing the model's accuracy, tuning it across training and testing dates. Metrics for the training and testing sets have to be logged.
2. **Product UI (for Normal Users):** The Product UI is designed for regular Dream11 users who want to build their fantasy teams. The main objective here is to provide an easy-to-use interface where users can input team details and receive predictions for the best 11 players based on various match-related features.

1.2 Problem Formulation

Model UI: The Model UI is built for data scientists and professionals to train the model, check for model accuracy, and analyze validation metrics.

- Ensure proper training and testing data handling.
- Track performance metrics and logs for analysis.

2 SOLUTION METHODOLOGY

The primary goal of this project was to predict Dream11 scores using cricket match data sourced from Cricsheet. The process involved multiple stages of **data extraction**, **preprocessing**, **exploratory data analysis** (EDA), **feature engineering**, **model training**, and **post-model explainability**.

2.1 Data Extraction and Preprocessing

This data was available in JSON format, which we downloaded using wget for processing.

- **Data Format:** The match data for each cricket game was stored in a separate file named matchId.json, where matchId represented the unique identifier for each match. Each file contained detailed commentary and meta-information about the match, such as the city, venue, and teams.
- **Chunked Data Extraction:** Given the large volume of ball-by-ball data, we used chunked extraction to avoid overloading the system's memory.
- **Data Conversion:** The data was initially stored in a JSON format, which was not ideal for processing. Therefore, the JSON files were converted into dataframes, in the form of ball-by-ball data. Each entry in the dataframe represented a single ball delivered in a match, with columns like ball_number, runs_scored, wicket_taken, and others. Given the size of the data, intermediate RAM dumps were employed to handle memory limitations.

2.2 Exploratory Data Analysis (EDA)

We conducted comprehensive EDA ,to engineer features for better model performance

- **Feature Distribution Plots:**

  We plotted the distribution of various features, such as runs scored, wickets taken, and strike rates, to understand their spread and identify any skewness. For features with heavy skewness, we applied log normalization to make them more suitable for modeling.

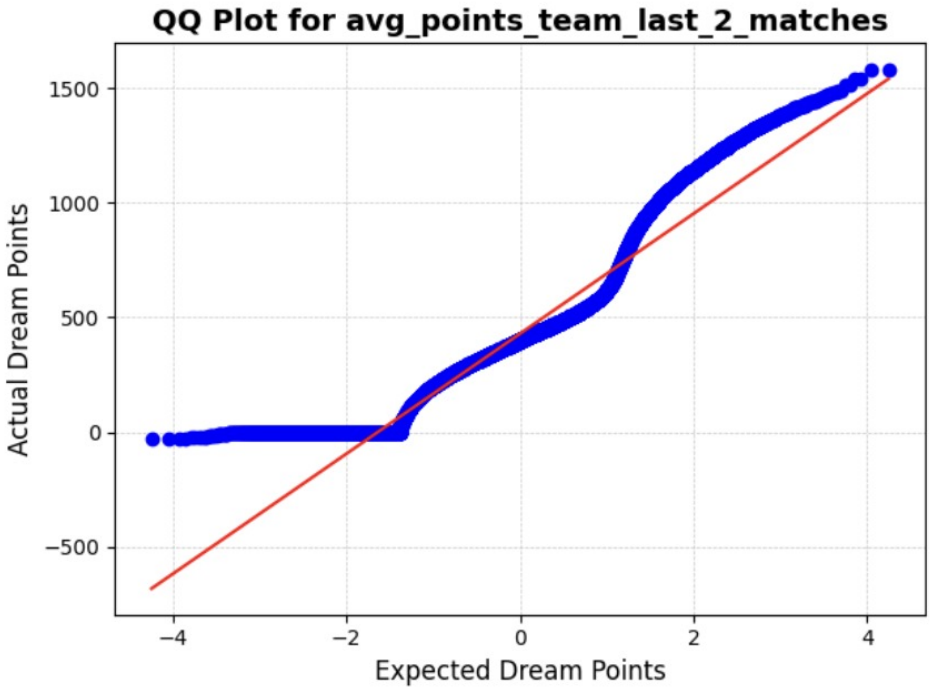

Figure 1: QQ Plot

The QQ plot in Fig 1 visually compares the distribution of actual Dream Points to a theoretical normal distribution. The blue dots represent the actual Dream Points, while the red line represents the expected values from a normal distribution.

  - The blue dots deviate significantly from the red line, especially in the lower and upper tails. This indicates that the distribution of actual Dream Points is not **normally distributed**.
  - There is a large cluster of points around the middle section of the plot, suggesting a concentration of Dream Points in this range.
  - The tails of the distribution appear to be heavier than those of a normal distribution, meaning there are more extreme values (both high and low) than expected.

  This plot depicts the distribution of actual average points for the team in last 2 matches, with Dream points, and it is left skewed, so we need to normalize the data .

- **Correlation Analysis:** Understanding how different features relate to each other is crucial for model development. We created heatmaps to visualize the correlation between features.

- **Feature Ranking with Recursive Feature Elimination (RFE)**

  We used **Recursive Feature Elimination** (RFE) to rank the importance of various features. This helped us narrow down the feature set to the most relevant variables for the Dream11 score prediction.

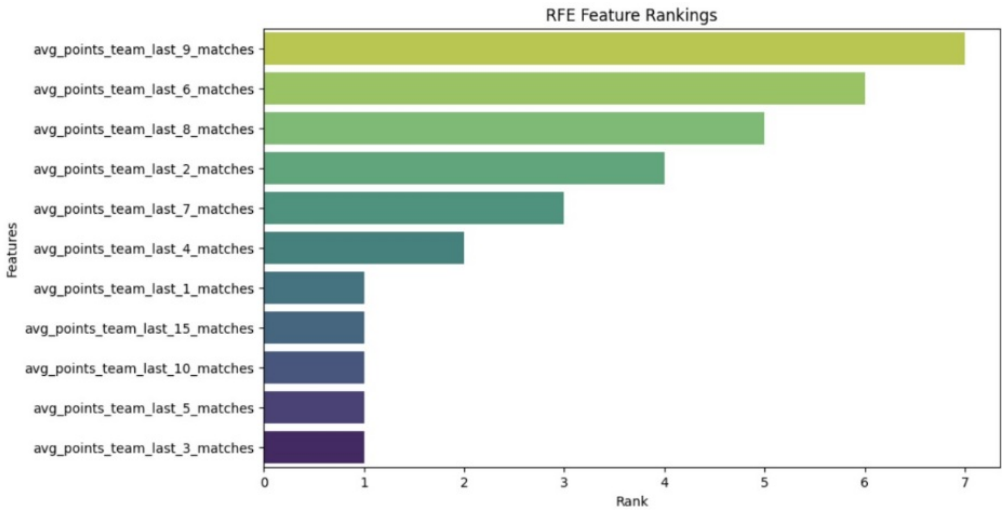

Figure 2: Feature Rankings by RFE

The graph in Fig 2 shows that features such as **average points of team in last 9 matches** and **average points of team in last 6 matches** have the **least importance**, as they are ranked higher, indicating they provide the least significant predictive power for the model. Features with longer time spans like average points of team in last 15 matches and smaller spans like average points of team in last 2 matches are ranked lower, suggesting they contribute highly to the model.

Selecting a mix of rolling averages (1, 2, 5, 10, 15) ensures capturing both immediate trends and more stable, long-term patterns, enhancing the model's ability to predict future performance effectively.

2.3 Feature Engineering

Based on insights gained from the EDA phase and user research, we identified key features that are known to impact a player's Dream11 score. These included factors like **venue**, **player form**, and **playing conditions**.

- **Venue and League:** We found that the venue and league conditions significantly influence the outcome of a match. For instance, certain venues tend to favor batsmen while others are better suited for bowlers. We engineered features like venue_points based on historical data at specific venues, and league_conditions. While creating these features we **doubled the rows** with swapping the teams, to make selection of team 1 and team 2 **invariant** of the predicting column.
- **Player Performance Metrics:** As part of the feature engineering process, we created multiple player performance metrics, such as:
  - **Runs per ball:** The average number of runs scored per ball by a player.
  - **Runs per over:** The average runs scored per over.
  - **Sixes per ball:** The number of sixes by the player, normalized by the number of balls faced.
  - **Sixes per match:** The total number of sixes hit by a player in a match.
  - **Current ICC Rankings:** Can be a good estimate of their playing levels.
- **Rolling features:** One of the critical insights from user research 6 was the importance of understanding the form of the player. Since the form of a player largely correlates with their past performances we incorporated Time Series Models like **ARIMA** and Caret to plot their **time series dependence**.

  However the performance is limited to only **3 months of past performances** (which is approximately **15 matches**). So we used this as a feature and introduced rolling average features of all the data points for the last 1,2,3,4,5,10,15 matches. The performance in the last match is given more weightage with linear decay in the other ones of their recent performances.

2.4 Model Development and Evaluation

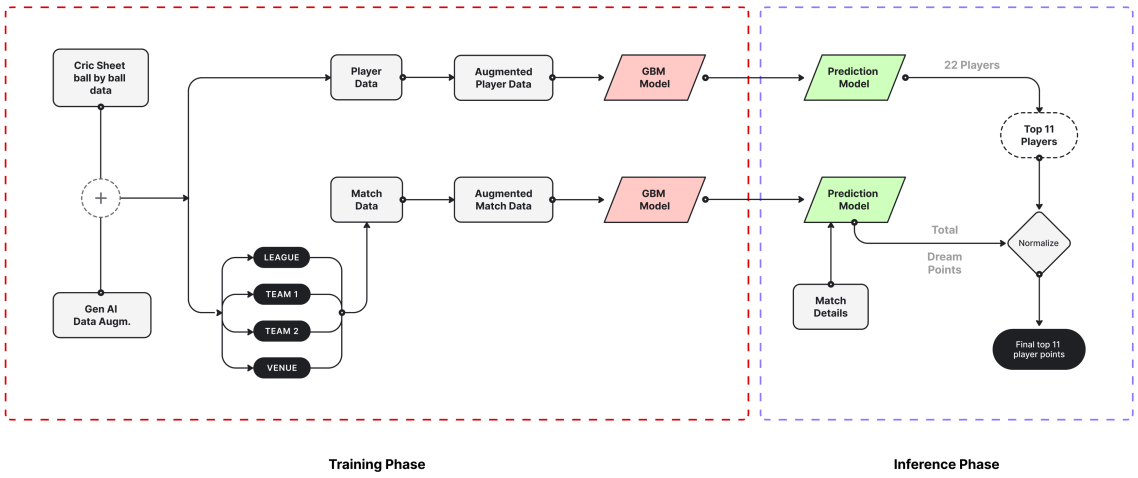

Figure 3: DREAM - *Dynamic Ranking Ensembled Analysis for Matches*

We devised a unique ensembling technique to incorporate the features from both the architectures, "**DREAM**" - **Dynamic Ranking & Ensembled Analysis for Matches**. Previously we tried out two approaches of predicting player points and match points. The player points did work on standard matches with equal opposition, while the match points prediction was comparatively more accurate in skewed match performances by the teams, so we need to extract the best features out of these two approaches and hence DREAM.

- **Model Selection:**For this task, we initially explored multiple regression models, including Linear Regression, Random Forests, and Gradient Boosting. The choice of model was based on its ability to handle both numerical and categorical data and its robustness against overfitting.
- **Model Training and Hyperparameter Tuning:** We used techniques such as grid search to improve the model's performance. The training process was monitored using Weights & Biases.
- **Performance Evaluation:** The model's performance was evaluated using the mean absolute error (MAE)& mean absolute percentage error(MAPE). We also visualized the error distribution on W&B.

We used ball-by-ball cricket data, augmented with Generative AI-generated player categories such as batsman, bowler, and all-rounder, to build two dataframes: the Player DataFrame and the Match DataFrame.

- In the **Player DataFrame**, we included average runs, wickets, and dream points from the last 15 matches, with weights being more to the recent ones.
- In the **Match DataFrame**, we added average points from the top players in recent matches at the same venue and within the same league.

We then used these dataframes to train Gradient Boosting Machine (GBM) models:

- The **Player Model** predicts the dream points a player will score in the next match.
- The **Match Model** predicts the total points scored by the top 11 players.

Finally, we selected the top 11 players based on the Player Model's predictions and adjusted their predicted points to match the output of the Match Model. This gave us the final predicted fantasy points for the top 11 players in the upcoming match.

2.5 Model Explainability with SHAP and Natural Language Processing

One of the key aspects of this project was ensuring that the predictions were interpretable to end users. We utilized SHAP values to explain the model's predictions. SHAP provides insights into how each feature contributes to a particular prediction, allowing users to understand why a certain player might score high or low on Dream11.

- **SHAP Integration:** SHAP values were integrated into the model to offer transparency regarding feature importance. This allowed us to pinpoint the most influential factors in predicting Dream11 scores, such as recent form, venue, and weather. The remaining data was passed on to Gemini API for conversion to natural language to the end user.

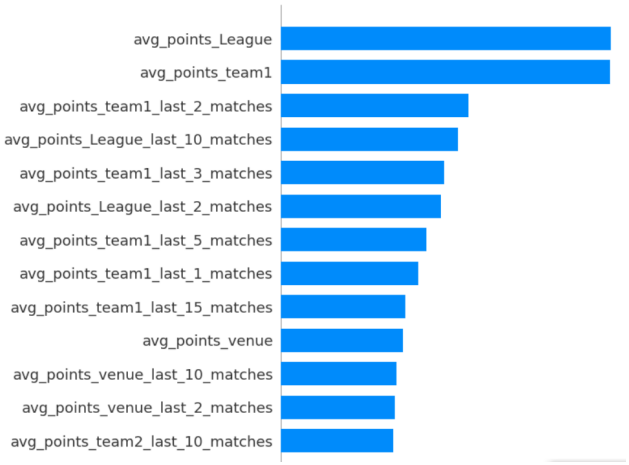

Figure 4: Top Features by SHAP

2.6 Product-UI Features for User Engagement

Drawing from our competitive analysis [B] and user research [6] highlighting Dream11's complex UI issues, we have designed a streamlined dashboard that prioritizes ease of use. This solution addresses the needs of our young user base (92% aged 18-24) who prefer quick, intuitive interactions.

- **Clean, Intuitive dashboard:** Displays ongoing and upcoming matches, ensuring ease of navigation.
- **One-click match and contest joining:** Reduces friction for users, enabling faster participation.
- **Mascot-Driven pre-match Presentation:** To address the 61% of users struggling with player performance prediction, we have introduced Dream, our AI-powered mascot that makes complex data digestible. The mascot interface is user-friendly, and is powered by Gen-AI based match insights through pre-match analysis videos.
- **AI-Assisted Team Selection:** Catering to the 45% of users open to AI recommendations and 33% finding statistics processing difficult, we have developed a balanced approach of guided team creation, AI-powered team bias suggestions and flexible player selection controls.
- **Captain/Vice-Captain Pre-Selection:** Strategic position selection informed by real-time data analysis, maximizing points potential, setting us apart from competitors like MyTeam11 and FanFight.
- **Interactive Player and Match Statistics:** Provides real-time data, AI-driven performance predictions, and in-depth player impact analysis, presented through intuitive visualizations that enhance user understanding and decision-making.
- **Justification for AI-Generated Teams:** Building trust through transparency, which is important especially for the 45% of users who are open to AI recommendations. We have provided clear explanations and justified each AI-suggested player selection.

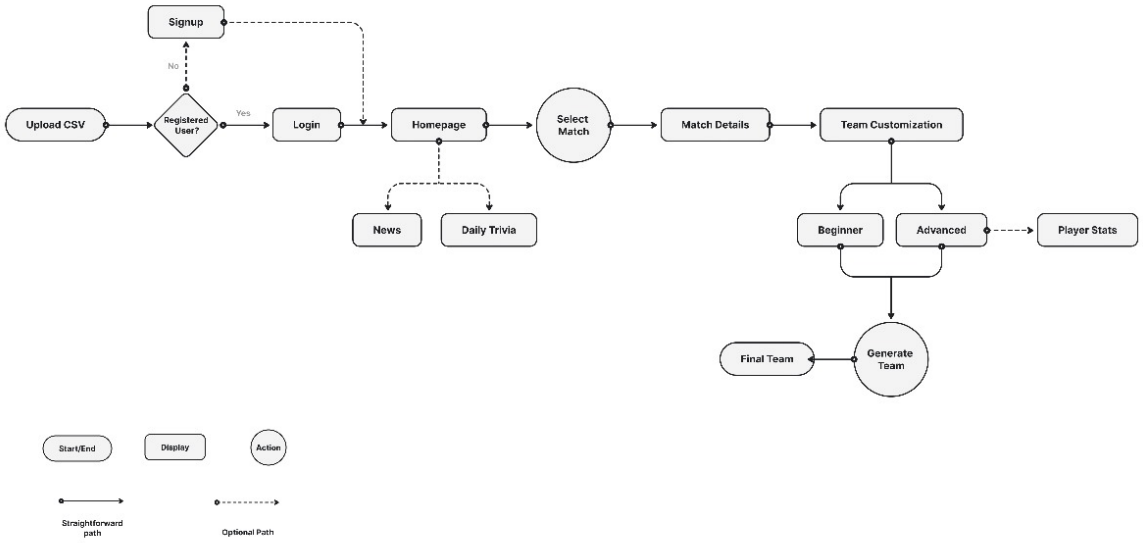

Figure 5: User Flow

- **Engagement Features Beyond Team Building:** Targeting the 20% occasional players identified in our research by integrating news feed for user engagement, gamified quizzes and leaderboards.
- **User Flow** The user flow is designed to provide a smooth, intuitive journey, guiding users from match discovery to team selection with minimal friction. It ensures quick access to key features like AI-powered recommendations and real-time statistics, enhancing user engagement and decision-making.

3 ANALYSIS AND EXPERIMENTATION

To improve the performance of our fantasy cricket prediction model, we conducted several analyses and experiments across various stages of model development. The primary aim was to reduce prediction error and enhance the model's ability to generate accurate player and team performance estimates. Below is a detailed breakdown of the key steps taken during this phase:

1. **Principal Component Analysis (PCA)**
   - **Feature Independence Analysis:** We began with Principal Component Analysis to assess the independence and relevance of the features used . By applying PCA, we were able to reduce the feature set to the most influential components, retaining 97% of the original feature variance.
   - **Feature Selection:** After performing PCA, we selected the most significant features that contributed to predicting player performance.
   - **Drawback:** MAE was increased by 10% on dropping the features.
2. **Model Selection and Experimentation** We experimented with a variety of machine learning models to identify the best approach for predicting player performance.
   - **CatBoost and XGBoost Regressors:** We began by testing gradient boosting algorithms like CatBoost and XGBoost, which are known for their high performance in handling large datasets and non-linear relationships. Both models performed well in terms of accuracy, but we needed to optimize them further to improve MAE.

- **Polynomial Regressor:** As part of our experiments, we also explored polynomial regression, which allows the model to capture nonlinear relationships between features.
- **Artificial Neural Networks (ANN):** We also experimented with Artificial Neural Networks (ANN) to model the complex relationships between the features.
- **Drawback:** All these models failed to capture both the innovation component and the time series component of the data, and hence we came up with your ensembling architecture - **DREAM**.

3. **Prediction Targets and Error Reduction** Initially, we attempted to predict individual player points directly. However, this approach resulted in an average Mean Absolute Error (MAE) of approximately 23 points per player, which was too high for reliable team-building recommendations.

   - **Team Points Prediction:** To reduce error propagation and improve the overall prediction accuracy, we switched to predicting team points rather than individual player points. This approach significantly reduced the MAE, as it averaged out individual player errors and provided more stable predictions for the entire team.
   - **Feature-Specific Prediction:** We also experimented with predicting specific player stats, such as runs, sixes, and fours. For predicting runs, we achieved an MAE of around 8 per player, while sixes and fours had a much lower MAE of around 1 per player. This granular approach helped in improving prediction accuracy for individual performance components.

4. **Correlation Analysis** We observed that the correlation between predicted points on Day-1 (D-1) and the final match day (D day) was quite low. This suggested that additional features related to match conditions, such as venue, tournament type, and opponent-specific factors, were needed to enhance prediction accuracy.

4 RESULTS

We used Gradient Boosting Regressor and got the best results at the following hyperparameter values :

```
n_estimators=84, learning_rate=0.1, max_depth=17, min_samples_split=15,
min_samples_leaf=15, subsample=0.9, max_features=1.0, random_state=5,
loss='absolute_error', tol=0.012770236105969923
```

We also experimented on the **T20I data available** on cricsheet and achieved the following results within the testing period from 01-07-2024 to 10-11-2024. :

- Total MAE : 36.41
- RMSE : 47.25
- R2 Score : 0.91

The given data had a very large number of irregularities, so when it is trained on a specific league or format or for a very short time frame, we achieved good results (with MAE around 32 and R2 score over 0.9

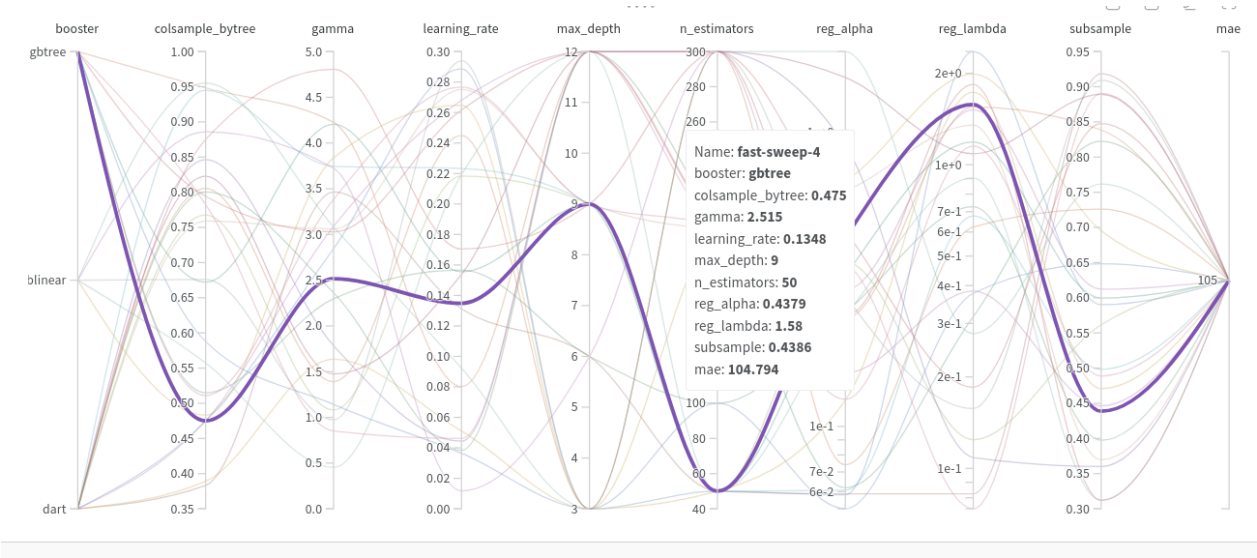

Figure 6: WanDB Hyperparamter Tuning

We generated results with the following statistics, while training on the **entire dataset** till 30-06-2024, and tested on the **entire dataset**.

- Total MAE : 104.79
- RMSE : 141.14
- MAPE : 21.91
- R2 Score : 0.69

5 TECHNICAL CHALLENGES

1. **Dataset Size and High Dimensionality:** The dataset, containing extensive cricket statistics across multiple formats (T20s, ODIs, Tests), included numerous features such as player performance, match data. This large volume of data posed significant memory and processing challenges.

   To address these issues, we implemented a chunk-based processing approach, dividing the dataset into smaller, more manageable pieces for sequential processing. This approach significantly reduced memory load. Additionally, RAM dump analysis was conducted to monitor system memory usage during peak processing periods. By analyzing memory consumption, performance bottlenecks were identified and the processing pipeline was optimized.
2. **Data Cleaning and Quality Issues:** The dataset contained 15-20% inaccurate CricSheet data, affecting multiple match types and causing inconsistencies across sources such as ESPNcricinfo, ICC, and others. We conducted cross-referencing with trusted sources to identify and correct discrepancies.

6 FUTURE SCOPES

1. **Expert Opinion Integration:**

   Dream11 currently offers expert-curated advice. The proposed enhancement integrates this guidance directly into the predictive model, assigning higher weights to expert inputs for relevant features.
2. **Multiple User Leaderboard System**

   - **Global Leaderboards:** Rank users based on their performance across all contests.

- **Regional and Group Leaderboards:** Introduce segmented leaderboards for localized competition, based on geography or user-created groups.
- **Real-Time Updates:** Ensure dynamic updates during live matches to maintain engagement. This feature would introduce gamification, motivating users to participate more actively.

3. **Dream11 Export**

   Users can easily integrate their optimized teams into Dream11 with a single click, reducing manual effort, and easy adoption.

4. **User Accounts and Personalization**

   - **User Profiles:** Track user history, preferences, and frequently selected players.
   - **Personalized Recommendations:** Provide tailored team suggestions based on past performance and preferences.
   - **Cross-Device Synchronization:** Ensure that user data is accessible across devices for a consistent experience. This feature will create a more personalized, engaging environment, improving user retention.

5. **User Staking**

   - Users can assign varying confidence levels to their predictions, earning rewards for accurate assessments.
   - Risk-and-reward mechanics will enhance gameplay by incentivizing accuracy and strategic thinking.

A User Survey

1. **Summary of User Research:**

   A recent fantasy cricket user survey provides valuable insights into user behavior and AI adoption. The data shows a young user base, with **92%** of respondents in the **18-24** age group. Among these users, **57%** identify as avid cricket fans following all formats, while **25%** primarily follow India's matches. Regarding participation patterns, **20%** are occasional players, with peak engagement during major tournaments like the IPL and World Cup. The survey revealed promising attitudes toward AI technology—**45%** of users are open to exploring AI-based recommendations if presented simply, while merely **6%** showed resistance to the technology. Team-building emerges as a significant challenge, with **61%** of users struggling to predict player performance and **33%** finding it difficult to process complex statistics. These pain points highlight the need for tools that simplify data analysis and deliver clear insights. Users specifically want predictive performance metrics, comparative player analysis, and transparent AI reasoning. These findings point to a clear opportunity: developing user-friendly, AI-powered features that strike the right balance between automated recommendations and user autonomy, while addressing the core challenges of accuracy and complexity in fantasy cricket.

Figure 7: User Insights on Fantasy Cricket Engagement and AI Assistance

B Competitive Analysis

Platform	User Base	Market Share	Product Offerings	User Engagement Model	Revenue Model	Strengths	Weaknesses	Marketing & Promotion
Dream 11	250M+	Highest (almost 90%)	Primarily cricket, with some football, kabaddi, and basketball	High, driven by a mix of real-time recommendations, points-based rewards, and frequent contests.	Entry fees for paid contests, partnerships, and advertisements.	Strong brand recognition, large user base, partnerships with sports leagues, and integration of AI for team recommendations.	Regulatory Risks, Limited multimedia features, complex UI	Heavy advertising, sponsorships with sports teams, and strategic partnerships with leagues.
My Team 11	20M+	2nd largest	Fantasy cricket, football, and kabaddi, with a focus on regional leagues.	Engages users through recommendations and regional league access.	Entry fees for contests and subscriptions.	Strong engagement in tier-2 and tier-3 cities with regional league focus.	Limited advanced player insights and lack of sophisticated prediction algorithms.	Focus on digital ads, regional marketing, and influencer collaborations.
My 11 Circle	10M+	Considera ble	Cricket, football, and unique contests that allow users to compete against celebrities like "Beat the Expert."	High, with challenges that allow users to compete against celebrities.	Entry fees, sponsored contests, and referrals.	Celebrity endorsement and unique challenges like "Beat Sourav Ganguly."	Limited real-time player insights; mostly static recommendations .	Aggressive ad campaigns featuring sports icons, creating a fan-centric community.
MPL	75M+	Significant	Cricket, multiple other fantasy sports, and a range of casual games.	Very high due to extensive gamification elements and points.	Game fees, ads, and partnerships.	Diverse offerings beyond fantasy sports, highly gamified experience.	Lack of sophisticated fantasy analysis	Focuses on gaming events, referral bonuses, and strong digital presence.
FanFight	6M+	Niche player	Primarily fantasy cricket with some other sports.	Moderate; known for its low-cost entry points.	Entry fees for various contests.	Simple UI, low entry fees, and accessible to entry-level users.	Limited features and advanced insights, which may not appeal to more engaged users.	Focused on grassroots marketing and affordability.
Gamezy	1M+	-	Cricket, football, and casual games like rummy.	High due to a mix of fantasy and casual games.	Entry fees, game purchases, and ads.	Good UI, real-time match stats, and mobile-first experience.	Limited support for in-depth player analytics.	Social media campaigns and incentives for daily players.

Figure 8: Competitive Analysis