Football Betting Support Tool

By Aaron Bankole

Goldsmiths University of London

Computer Science, 2025

Student Number: 33776493

Abstract

Sports betting for popular sports such as football is widely prevalent worldwide. "So, the perception of the game and the prediction of results is of general interest to fans, coaches, media and gamblers. Although predicting football results is a very complex task, the football betting business has grown over time." [1] Through various adverts, posters, and the like, sports betting has become a widespread industry. "Most streets in the capital of England have a betting shop with the most significant bookmakers being William Hill, Betfred, Ladbrokes, Coral and Paddy Power." [2]

To this end, I believe that a tool to aid the prediction of bettor outcome is relevant now more than ever. In this project, the objective is to create a dashboard for sports betting that collates previous bets, and in the future, collates current bets and aggregates these to produce bets with the best probable odds. A major aim is to utilise machine learning for the analysis of previous data so that bets can produce outcomes that are more beneficial for the user based on past data to have a more fulfilling and rewarding experience.

Introduction and Literature Review

My project focuses on creating a dashboard for sports betting that collates previous bets, and in the future, collates current bets and aggregates these to generate bets with the best possible odds. I plan to incorporate machine learning to then analyse previous data so that bets can produce outcomes that are more favourable for the user based on past data to have a more fulfilling and profitable experience.

The main idea behind this project is to offer support to people who are placing bets using services such as Bet365, William Hill, Paddy Power etc. A lot of people who place bets tend to lose out because sports betting is extremely unpredictable [3]. This article from *Responsible Gambler* discussed the Sports Betting Industry, understanding the common mistakes that are made during betting and the structure of the market as, "designed to favour the sportsbook, making it difficult for bettors to come out ahead in the long run. The market's efficiency means that even the best bettors struggle to achieve long-term success, as the odds are calculated to reflect all available information." There is also a problem of people becoming increasingly addicted to betting due to influence from many external sources that exist, whether they be friends of family, commentators, adverts etc. Many bettors fall victim to the gambler's fallacy, the mistaken belief that past events can influence future outcomes in a sporting event. For example, some might think that a team is "due" for a win after a losing streak, leading them to place a bet based on this faulty logic.

This article from *Men's Health* [4] discusses the nationwide impact of betting and how it has become an indispensable part of the everyday life of a football fan, constantly checking odds and analysing every obscure Premier League player in the hopes of securing a big win. The article also discusses how Britain has now acknowledged that

gambling is a serious problem, dubbed as "Changing the Game."; my project is inspired by the difficulty that comes when placing a bet and expecting to make a profit. My idea is also inspired by the crisis of football betting and how problematic it is in that betting is widely available and can be addictive leading to catastrophic effects for some people.

Software's like Bettin.gs [5] create betting portfolios for their readers, analysing performances and following betting experts without automated betting trackers. With real-time odds comparison across multiple bookmakers, users can quickly find the best value for their bets, improving their chances of success. Bettin.gs also offers live betting features, allowing users to place wagers as events unfold, adding excitement and flexibility to their betting strategies. They also have a large community of bettors that users can interact with worldwide to discover the most popular bets based on all registered bets. Moreover, Bettin.gs allows users to follow their favourite tipster and get instant notifications when they have registered a bet. Furthermore, the platform promotes responsible gambling by offering tools for setting limits and tracking betting activity to curb addictions and prevent users from having horrible experiences. However, they only offer basic features for free and incentivise a pro version that they want users to pay for that gives them the ability to see all the most popular bets, to auto settle bets and track and an unlimited following of the most successful users. The site is also restricted to payment in only dollars, so for certain countries that don't have a way to exchange to dollars they are unable to use the professional service, due to the differing legal frameworks it's also hard to create a service that works around all of them, so the service is limited or unavailable to some. Its reliance on user-generated content and algorithmic predictions can lead to inconsistent accuracy affecting the overall reliability of the platform for serious bettors. I want to make my platform free to ensure that it is easy to access and that there are no restrictions that come with using the platform so that users are allowed to place bets, see all statistics and have Application Programming Interface (API) and widget access. Unfortunately, to keep this service running, it would need to have advertisements to cover the cost that is required to run the service.

Other softwares such as RebelBetting, [6] are platforms that advertise themselves as tools that turn sports bets into investments by searching for profitable bet situations by searching for the best odds to find profitable situations. Their servers gather and analyse over one million odds every few seconds to take advantage of these profitable situations. They use two methods, which are value betting and sure betting, to be able to guarantee profits. The problem with RebelBetting is that it can be time-consuming to require users to actively compare the better bets and monitor odds to maximise profits. In smaller markets, where there are fewer bookmakers, there will be less profitable situations as there aren't as many available, making RebelBetting a less effective tool and creating more of a double-edged sword instead of a reliable betting tool.

This article from SDLC Corp [7] lists the tools and resources essential for any sports bettors. It mentions Statistical Analysis Tools like Data Analytics Software that measure

a player's performance, team form, head-to-head records, and more. A lot of betting dashboards provide built-in analytics that let users examine historical data, making it easier to make informed decisions. Also, highlighting Third-Party Analytical Platforms that offer deeper insights and Odds Comparison Tools, which aggregate odds and save time by handpicking the best available odds for any given event, there are odds comparison tools built into dashboards, so they become the place for their users to have all the tools needed. Bet tracking Software is crucial for a bettor's improvement, as it keeps a record of all bets, they have placed to analyse betting patterns and common tendencies for bettors to be able to improve their performance.

I aim to incorporate all these features into my project, but also to include a disclaimer to users as sports bets are highly unpredictable and can swing one way or the other unexpectedly. It's important to note that this dashboard isn't a guarantee for users to win big on bets and wasn't intended to be; therefore, they should and will be advised to exercise caution and discernment when placing bets.

There are also Bankroll Management Tools to ensure bettors can control and dissuade them from reckless and impulsive bets that are likely to incur a loss. But this has not being implemented in my project as it is not an objective to create a platform to create sports bet. Rather my objective is to provide information to enable bettors make informed decisions using the tools provided on the platform I am creating.

I think it would be an indispensable service if there were a new betting platform that can differentiate itself from existing ones by focusing on innovation, user experience, and advanced technology. By leveraging artificial intelligence and machine learning, it can offer personalised betting recommendations and predictive analytics, personalising the experience for every user, integrating features such as real-time, in-depth statistics, interactive betting options, and adjustments that enhance the thrill of live betting while reducing the risk. A focus on responsible gambling with customisable limits, self-exclusion options, and educational resources could appeal to bettors seeking more control over their habits. By improving the ease of use and prioritising customer support, a new platform could build stronger trust and loyalty, offering a more engaging and ethical betting experience.

Creating a successful betting dashboard requires comprehensive research across several key areas to ensure it meets user needs, complies with regulations, and provides a seamless and engaging experience. First, it requires gathering a large dataset of historical betting odds, outcomes, and other relevant variables such as team statistics, player performances, and market movements. I plan to get these from an English betting online archive [8], where there are lists and times of every game played, the referee of each game and the goals scored to keep a metric of the results, this is however specifically filtered for the Premier League as for a starter for the dashboard.

I also have a database of players that I will use for the corresponding seasons to get every player playing in the Premier League at that specific time. This data will serve as

the foundation for training a machine learning model. The next step is to preprocess the data to ensure it is suitable for model training, addressing issues like missing values, outliers, and data normalisation, after this model is trained it needs to test for accuracy and be further refined. After this, the tool can be integrated with betting data to monitor odds and identify where the profitable solutions will occur. However, the tool's effectiveness depends on the quality of the data and the nature of betting markets, meaning it must be constantly updated and adjusted by repeating this process after a certain period. Finally, usability testing and continuous user feedback are essential to refine the dashboard's functionality and design, ensuring it remains competitive and user-centred over time.

I see that my project fits in with the modules I am taking, primarily due to it being machine learning based and one of my bigger modules is machine learning. I was also inspired by the dashboard that I had created last year and wanted to create something like that, as football is an interesting sport that garners a lot of conversation. This also ties in with databases as I am using data sets that I am processing and normalising databases so that they can then be used. To do this, I also would like to use Jupyter Notebook to generate graphs using the datasets, which ties in with my Data Visualisation module.

Design Spec: Requirements

The project I wish to make is a dashboard that has tools to measure metrics about multiple players across multiple seasons. This collects data about current players and each season that they have played within the Premier League, such as goals, assists, and minutes. Using a Machine Learning Model, like Linear Regression, I hope to use this to predict a player's performance within the next season. Alongside this, I also plan to implement web scraping to collect previous bets for matches. Using previous season data, we can predict bets for upcoming seasons.

List of Functional Requirements

1. User Registration and Authentication

- Allow users to register using a valid email address and password.
- Send a confirmation email to verify new accounts.
- Allow users to log in with their credentials.
- Provide password recovery and reset functionality.
- Support two-factor authentication (2FA) for enhanced security.

2. User Profile Management

- Allow users to view and update personal profile information.
- Display the user's betting history and current bets.
- Show user balance, deposits, withdrawals, and earnings.

3. Match and League Information

- Display upcoming football matches from supported leagues.
- Show match details including team names, date, time, venue, and odds.
- Allow users to filter matches by league, country, or date.
- Update match outcomes in real-time or as soon as results are finalized.

A description of user needs

General Users (Casual and Regular Bettors)

Need to register and access the system easily

- Users need a simple sign-up/login process.

- They may want social or mobile login options (Google, Apple, Facebook).

Need to browse and explore matches

- Users want a clear and up-to-date list of football matches, sorted by leagues, dates, and teams.

- They need access to relevant match data (teams, time, venue, stats, form).

Admin and System Operators

Need to manage users and accounts

- Admins need to verify users, resolve issues, suspend accounts, and monitor suspicious activity.

Need to manage match data and betting markets

- Admins need tools to add/edit matches and update odds.

Need compliance and reporting tools

- Admins need tools for generating usage, financial, and risk reports.
- They need to ensure the system complies with local gambling regulations.

Testing Plan

A testing plan for the prototype of the football support betting app will focus on validating core user flows, ensuring functional reliability, and identifying usability issues before full-scale development. The plan will include unit testing of critical functions such as user registration, login, and inputting of data to verify backend logic and data integrity. Integration testing will confirm that components (match data feeds and odds updates) function correctly. A priority will be given to error cases, such as inputting the incorrect data into the models, to assess how the system handles errors.

Prototype – Failed Tests

```
# # Combine all into one DataFrame
# league_df = pd.concat(all_seasons, ignore_index=True)

# # Convert numeric columns safely
# numeric_cols = ['W', 'D', 'L', 'GF', 'GA', 'GD', 'Pts', 'xG', 'xGA', 'xGD', 'xGD/90']
# league_df[numeric_cols] = League_df[numeric_cols].apply(pd.to_numeric, errors="coerce")
# league_df = league_df.dropna(subset=numeric_cols)

# # --- 3. Build Dashboard Widgets ---
# season_dropdown = widgets.Dropdown(
#     options=sorted(league_df["Season"].unique()),
#     value="2020-2021",
#     description="Season:"
# )

# def update_dashboard(selected_season):
#     season_data = league_df[league_df["Season"] == selected_season].copy()

#     # Display league table
#     display(season_data[["Rk", "Squad", "W", "D", "L", "GF", "GA", "GD", "Pts", "Top Team Scorer"]]
#             .sort_values(by="Pts", ascending=False).reset_index(drop=True))

#     # Bar chart of team points
#     fig = px.bar(
#         season_data.sort_values(by="Pts", ascending=False),
#         x="Squad", y="Pts", title=f"Premier League {selected_season} Points",
#         text="Pts", height=450
#     )
#     fig.update_traces(textposition='outside')
#     fig.update_layout(xaxis_tickangle=-45, margin=dict(t=40, b=100))
#     fig.show()

# # Interactive output
# widgets.interact(update_dashboard, selected_season=season_dropdown)
```

```
🔽 Loading 2016-2017...
❌ Failed to load 2016-2017: 403 Client Error: Forbidden for url: https://fbref.com/en/comps/9/1526/2016-2017-Premier-League-
Stats
🔽 Loading 2017-2018...
❌ Failed to load 2017-2018: 403 Client Error: Forbidden for url: https://fbref.com/en/comps/9/1889/2017-2018-Premier-League-
Stats
🔽 Loading 2018-2019...
❌ Failed to load 2018-2019: 403 Client Error: Forbidden for url: https://fbref.com/en/comps/9/3232/2018-2019-Premier-League-
Stats
🔽 Loading 2019-2020...
❌ Failed to load 2019-2020: 403 Client Error: Forbidden for url: https://fbref.com/en/comps/9/3233/2019-2020-Premier-League-
Stats
🔽 Loading 2020-2021...
❌ Failed to load 2020-2021: 403 Client Error: Forbidden for url: https://fbref.com/en/comps/9/10728/2020-2021-Premier-League
-Stats
--------------------------------------------------------------------
```

The code above shows a screenshot of a failed web scrape due to too many frequent requests resulting in the 403 error. This happened to due to frequent requests when debugging. I made the mistake of not saving the code to a html or CSS file.

The code shows a list of seasons which are concatenated into a singular large data frame **(league_df). ignore_index=True** ensures the new data frame has a continuous index.

numeric_cols = [...]: Defines a list of column names that contain numerical data. These columns represent various statistics like wins (W), draws (D), losses (L), goals for (GF), goals against (GA), goal difference (GD), points (Pts), and more specific metrics like **'xG', 'xGA', 'xGD', 'xGD/90'.** The column data is then converted into a numeric data type.

options=sorted(League_df["Season"].unique()): The options for the dropdown are populated by the unique values in the "Season" column of the League_df, sorted alphabetically or numerically.

This data is then formatted as a dropdown widget.


```
# Failed Tests

In [95]: # league_df.loc[:, "top_6"] = league_df["Rk"] < 6
         # league_df.loc[:, "relegated"] = league_df["Rk"] >= 17

         # X = league_df[numeric_cols]
         # y_top6 = league_df["top_6"].astype(int)

         # X_train, X_test, y_train, y_test = train_test_split(X, y_top6, test_size=0.3,

         # model_top6 = RandomForestClassifier(n_estimators=100, random_state=42)
         # model_top6.fit(X_train, y_train)
         # print("🏆 Top 6 Prediction:")
         # print(classification_report(y_test, model_top6.predict(X_test)))

         # y_relegation = league_df["relegated"].astype(int)

         # X_train, X_test, y_train, y_test = train_test_split(X, y_relegation, test_size

         # model_relegation = RandomForestClassifier(n_estimators=100, random_state=42)
         # model_relegation.fit(X_train, y_train)
         # print("⛔ Relegation Prediction:")
         # print(classification_report(y_test, model_relegation.predict(X_test)))
```

The above cell is code for 2 previous models to collect the 6 teams and the bottom 3 teams for relegation. This uses a Random Forest Model to combine the output of multiple decision trees to reach a single result such as whether a team is in the top 6 or whether a team is being relegated. This is no longer used due to the previous web scrape being denied due to too many frequent requests.

Finished Product
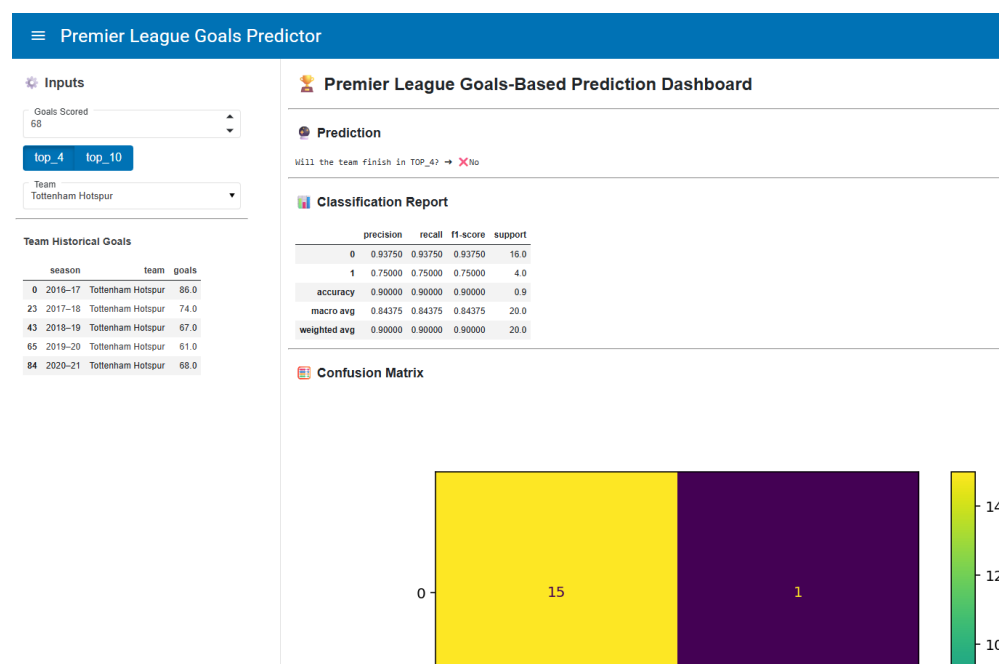
The above screenshot shows the finished product adding all the elements together to create the dashboard.

Header

- **"Premier League Goals Predictor"**: The main title of the application, indicating its purpose.

- A hamburger menu icon (three horizontal lines) is on the left, adding interactivity to the dashboard as the sidebar slides in and out.

Left Sidebar - Inputs Section:

- **"Inputs"**: A collapsible section header.

- **"Goals Scored"**: An input field with the value "68". This is where a user can manually enter the number of goals a team has scored for a given season.

- **"top_4" / "top_10" buttons**: These are toggle or radio buttons. Currently, "top_4" is selected, so the prediction is focused on whether a team finishes in the top 4 positions. If "top_10" were selected, the prediction changes to whether the team finishes in the top 10.

- **"Team" dropdown**: A dropdown menu displaying "Tottenham Hotspur". This allows the user to select a specific Premier League team.

3. Left Sidebar - Team Historical Goals Section:

- **"Team Historical Goals"**: A section displaying a table of historical goal data for the selected team (Tottenham Hotspur in this case).

- The table has three columns:

  - **season**: The football season (e.g., 2016-17, 2017-18).

  - **team**: The team name (consistently "Tottenham Hotspur").

  - **goals**: The number of goals scored by that team in that specific season.

- The data shown is for Tottenham Hotspur from the 2016-17 season (86 goals) up to the 2020-21 season (68 goals).

4. Main Content Area - "Premier League Goals-Based Prediction Dashboard":

- This is the primary area displaying the prediction results and performance metrics of the underlying model.

- **Trophy icon and "Premier League Goals-Based Prediction Dashboard"**: A subtitle reinforcing the dashboard's purpose.

5. Prediction Section:

- **"Prediction"**: A collapsible section.

- **"Will the team finish in TOP_4?"**: A question the dashboard aims to answer.

- **"Yes / No" indicator**: Currently, it shows "No" with a red cross icon next to it, indicating that based on the current inputs (Tottenham Hotspur, 68 Goals Scored, predicting for TOP 4), the model predicts that Tottenham Hotspur will not finish in the top 4.

6. Classification Report Section:

- **"Classification Report"**: A collapsible section displaying a table that summarizes the performance of the machine learning model used for prediction.

- This report shows metrics for each class (in this case, "0" and "1", likely representing "Not Top 4" and "Top 4" respectively):

  - **precision**: The ratio of correctly predicted positive observations to the total predicted positive observations.

  - **recall**: The ratio of correctly predicted positive observations to all observations in actual class.

  - **f1-score**: The weighted average of Precision and Recall.

  - **support**: The number of actual occurrences of the class in the specified dataset.

- It also includes:

  - **accuracy**: The overall accuracy of the model.

  - **macro average**: Unweighted mean of the precision, recall, and f1-score for each label.

  - **weighted average**: Mean of the precision, recall, and f1-score for each label, weighted by support (the number of true instances for each label).

7. Confusion Matrix Section:

- **"Confusion Matrix"**: A collapsible section displaying a visual representation of the model's performance.

- A heatmap is shown, with two distinct colour blocks:

  - A large yellow block with the number "15" in it.

  - A smaller purple/darker block with the number "1" in it.

- The axes are labelled "0" and "1", corresponding to the classes in the classification report.

- A colour bar on the right indicates the mapping of colours to values (from approximately 10 to 14, with some colours extending beyond).

- **Interpretation of Confusion Matrix:** Without knowing which axis represents "actual" and "predicted", or which number represents which class (0 or 1), a full interpretation is difficult. However, typically:
  - The diagonal elements (yellow, "15") represent correct predictions.
  - Off-diagonal elements (purple, "1") represent incorrect predictions (e.g., predicted "1" but actual was "0", or vice versa).
  - The "15" likely represents many correct predictions (e.g., correctly predicted that a team would not be in the top 4).
  - The "1" represents a single misclassification.

Conclusion

This project focuses on taking existing data and using Machine Learning Models, such as: Linear Regression and Neural Networks, to recognise patterns and attempt to predict future betting trends. Four different datasets are trained between both models, and each has its own outputs. Bets come in various forms, like the odds of individual matches or the final standings of the season, etc.

The project revolves around these key aspects: football match prediction evaluation, previous bets and performance evaluation. The performance evaluation focuses on the goals scored across past seasons and predicts how many goals will be scored in a later period. The match prediction takes bets from the previous Premier League seasons and uses Neural Networks to attempt to identify a pattern for a certain bet in the future.

However, the study has significant limitations. Although there are multiple datasets, actual predictions are hard to make as they cannot accurately predict what player will be on what team in the future. This drastically affects predictions as the teams will be different compared to previous data. Predicting bets is also extremely volatile as the odds are ever changing based on the current trends. For example, if Arsenal are in good form, the odds of them winning are likely to be high for each bet that can be placed. Therefore, future developments should connect to live bet makers (William Hill, Bet 365, etc.) to have accurate bets predicted. Furthermore, continuous optimisation of the model's architecture and algorithms is crucial.
In addition, future work should also account for the scale and diversity of datasets and incorporate additional factors that influence match outcomes. These factors include in-depth analysis of player psychological data, tactical changes by teams at different stages of the match, and real-time dynamic information from the surrounding environment of the match venue, such as local weather conditions, number of fans attending (either for home or away games). Because this project is being done on my machine and not a server, it is hard to store a massive amount of relevant data, so the sample size is limited. This was limited to data only being collected for 5 seasons (2016-2021). Future work would benefit from being conducted off a server to enable a larger scope of data set collection, training, and analysis.

As previously stated, I used a Linear Regression model to recognise patterns and attempted to predict future betting trends. In the future, the Linear regression model could be better used by prioritising bets and estimating value opportunities across a larger volume of games. This could lead to the expansion of the application being used for other sports such as basketball, tennis, horse racing, and much more through targeted adjustments and optimisations.

Despite the limitations listed above, I believe my project is beneficial for bettors as it would help them to make more informed decisions with data being readily available to them to consider when placing bets. This will allow the gambler to know the risk associated with the bet, thus having greater support in obtaining profit from sports betting.

References

[1] "Prediction of football match results with Machine Learning", Fátima Rodrigues Polytechnic Institute of Porto, https://www.researchgate.net/publication/363472367_Prediction_of_football_match_results_with_Machine_Learning (accessed July. 20, 2025).

[2] "Betting Shops London" , British Racecourses https://www.britishracecourses.org/betting-shops-london/ (accessed July. 20, 2025).

[3] "The surprising truth: Why is sports betting so hard? - sports betting guide," Responsible Gambler, https://rg.org/guides/sportsbetting-guides/why-is-sports-betting-difficult (accessed Nov. 8, 2024).

[4] Fair game? inside Britain's football betting crisis, https://www.menshealth.com/uk/mental-strength/a60742999/britains-football-betting-crisis/ (accessed Nov. 7, 2024).

[5] "Track and analyse your bets. improve your betting performance," Bettin.gs, https://bettin.gs/ (accessed Nov. 1, 2024).

[6] "Sports betting - value & sure betting software - RebelBetting," Value betting & Sure betting by RebelBetting, https://www.rebelbetting.com/ (accessed Nov. 6, 2024).

[7] A. K. Kutaval, "Essential tools and resources for sports bettors," SDLC Corp, https://sdlccorp.com/post/essential-tools-and-resources-for-sports-bettors/ (accessed Nov. 12, 2024).

[8] "England football results betting odds: Premiership Results & Betting odds," Football Betting - Football Results - Free Bets, https://www.football-data.co.uk/englandm.php (accessed Nov. 11, 2024).