NATIONAL RESEARCH UNIVERSITY
HIGHER SCHOOL OF ECONOMICS

Faculty of Computer Science
Bachelor's Programme "HSE and University of London Double Degree Program in Data Science and Business Analytics"

Programming Project Report on the Topic: Forecasting the results of sporting events

Fulfilled by:

Student of the group БПАД223
Pankova Anzhelika Sergeevna _____ _____
(signature) (date)

Assessed by the Project Supervisor:

Rudakov Kirill Aleksandrovich
Teacher
Faculty of Computer Science, HSE University

_____ _____
(signature) (date)

Moscow 2025

Contents

Annotation

Футбол остаётся одним из самых популярных видов спорта в мире, объединяя миллионы болельщиков и привлекая внимание как профессионалов, так и любителей. Одним из самых престижных и масштабных футбольных турниров является Лига чемпионов УЕФА — ежегодный клубный турнир, где участвуют сильнейшие команды Европы. Каждый матч этого турнира — не только яркое спортивное событие, но и объект повышенного интереса со стороны аналитиков и игроков в сфере беттинга. С развитием технологий прогнозирование исходов спортивных событий всё чаще основывается не только на интуиции или опыте, но и на математических моделях и алгоритмах машинного обучения. Ставки на спорт становятся всё более популярными, а точные прогнозы позволяют не только повышать эффективность ставок, но и углубляться в понимание игры. Данная работа посвящена исследованию существующих подходов к прогнозированию результатов футбольных матчей, с акцентом на матчи Лиги чемпионов. В рамках проекта рассматриваются различные методы машинного обучения, способные обрабатывать статистику команд, историю очных встреч, новости о команде и другие факторы, влияющие на исход матча. Также разрабатывается собственная модель прогнозирования, основанная на анализе доступных данных и применении современных алгоритмов машинного обучения. Основной целью является создание инструмента, способного автоматически предсказывать наиболее вероятные исходы матчей Лиги чемпионов с высокой степенью точности.

Football remains one of the most popular sports in the world, bringing together millions of fans and attracting the attention of both professionals and amateurs. One of the most prestigious and large—scale football tournaments is the UEFA Champions League, an annual club tournament where the strongest teams in Europe participate. Each match of this tournament is not only a vivid sporting event, but also an object of increased interest from analysts and players in the betting industry. With the development of technology, predicting the outcome of sporting events is increasingly based not only on intuition or experience, but also on mathematical models and machine learning algorithms. Sports betting is becoming more and more popular, and accurate predictions allow not only to increase the effectiveness of betting, but also to deepen the understanding of the game. This paper is devoted to the study of existing approaches to predicting the results of football matches, with an emphasis on Champions League matches. The project examines various machine learning methods capable of processing team statistics, the history of face-to-face meetings, news about the team and other factors affecting the outcome of the match. We are also developing our own forecasting model based on the analysis of available data and the use of modern machine learning algorithms. The main goal is to create a tool capable of automatically predicting the most likely outcomes of Champions League matches with a high degree of accuracy.

Key words : prediction, machine learning, outcomes, algorithms, betting

1 Introduction

Nowadays, betting is quite popular among football fans as bets make watching games more attractive and ticklish. Forecasting of results is hard, thus making profit from betting is a controversial question because of many factors that can influence on the outcome of exact match. However, there are some existing methods that helpful in predicting results of sport events.

Important definitions

Beautiful Soup - Python library for extracting data from HTML and XML files. It provides a simple and convenient way to extract data from web pages and makes it easier to work with that data.

Requests - Python module that is used to make working with HTTP requests easier. Helps in passing parameters to URLs.

Poisson Distribution — a probability distribution that models the number of events occurring within a fixed interval of time or space, given a known average rate and independence between events. Commonly used in sports forecasting to estimate goal probabilities.

XGBoost Poisson — an extension of the XGBoost algorithm tailored for count data, using Poisson regression as the objective function to model goals scored in football matches.

LightGBM Poisson — a variant of the LightGBM gradient boosting framework optimized for Poisson regression tasks, particularly useful for predicting goal counts based on historical match data.

Grid Search CV — a method for hyperparameter tuning that systematically works through multiple combinations of parameter values to determine the best-performing model. It uses cross-validation to evaluate each combination and selects the one that maximizes performance metrics such as accuracy or F1-score.

Correlation Matrix — a table showing correlation coefficients between multiple variables, used to assess the strength and direction of relationships among features in a dataset. In this project, it helps identify how sentiment scores relate to match outcomes and team performance indicators.

Accuracy — a classification metric that measures the ratio of correctly predicted

instances (both true positives and true negatives) out of all predictions made.

Precision — a classification metric measuring the proportion of positive identifications that were actually correct. In other words, how many selected items are relevant.

Recall — a classification metric measuring the proportion of actual positives that were identified correctly. It shows how many of the relevant items were selected.

F1-score — a weighted average of precision and recall, providing a balanced measure of a model's performance. It reaches its best value at 1 and worst at 0.

Log-loss — a metric that evaluates the confidence of a model's predictions by penalizing incorrect classifications more heavily when the model is highly confident in them. It is commonly used in probabilistic classifiers and is especially valuable when the output of the model is a probability score. Lower values indicate better performance.

BERT Sentiment — a natural language processing technique using the BERT model to analyze and classify text sentiment with high accuracy. Unlike traditional methods, BERT captures context and semantic meaning, making it suitable for analyzing complex expressions in news articles.

VADER Sentiment — a rule-based tool from the NLTK library that provides sentiment scores based on text input. It is especially effective for short, social media-style texts and gives a compound score that reflects the overall sentiment of the input.

My strategy and plan

Strategy

At the start of my project, I aimed to develop a model capable of predicting football match outcomes. I decided to use Python and its libraries to achieve this goal, building a system that can automatically update relevant data and compare any two teams from a selected competition. My focus was on the UEFA Champions League — one of the most prestigious football tournaments, where top European clubs compete against each other. I explored various sources of match statistics and planned to use data such as goals scored and conceded, home or away advantage, possession percentage, fouls, shots on target, saves, and corner kicks — among other relevant performance indicators. Throughout the project, I studied various research papers and prediction methodologies. During this process, I discovered that the Poisson distribution can be particularly useful in forecasting match outcomes — especially in predicting exact scores, which added a new and exciting dimension to my approach. Additionally, I planned to explore the possibility of tracking news articles related to football clubs to determine whether positive or negative news could influence a team's performance on the field. This sentiment-based analysis could provide deeper insights into non-statistical factors affecting match results.

Details arisen during the process

During the project, I faced several practical challenges related to data collection and preparation. One of the first tasks was finding a reliable source that provides detailed match statistics for the UEFA Champions League. Many websites offered only basic data — such as goals scored or possession percentage — and lacked deeper metrics like xG (expected goals), shots on target, or player injuries.

Eventually, I found a suitable website and built a custom scraper using `BeautifulSoup` and `requests`. Scraping allowed me to collect fresh, up-to-date information directly from web pages, which gave more flexibility compared to static datasets or APIs. This approach also helped gather extended stats for each individual match — including venue, weather, and team lineups — which were crucial for more accurate predictions.

To improve model performance and generalization, I used data from the last 7 seasons — over 1500 matches in total. A larger dataset allowed models to better capture patterns and variations across teams and playing conditions. It also helped reduce overfitting and increased confidence in the results.

Overall, this stage taught me how important it is not only to find good data but also to structure and clean it properly before feeding it into machine learning models.

2 Already known methods

Monte-Carlo method

The Monte-Carlo method is based on simulating a large number of random scenarios

to estimate the probability of a desired outcome. It relies on the idea that if an event can be repeated under identical conditions multiple times, its likelihood can be approximated by observing the frequency of outcomes across these repetitions.

In this approach, specific conditions are defined and remain constant throughout the simulation. However, one limitation of the method is its reliance on historical data: for rare or unique events, it may be difficult to generate sufficient simulations due to the lack of comparable past occurrences.

This technique is widely used in various domains, including sports betting. In football match prediction, the method involves modeling different scenarios with varying input factors — such as recent performance, player injuries, and venue — to calculate the probabilities of possible match outcomes.

To illustrate how this method might be applied, consider a Champions League match between FC Barcelona and Napoli, which took place on February 21 as part of the Round of 16. Since the match was held in Spain, FC Barcelona played as the home team. The table below demonstrates a simplified version of how the Monte-Carlo method could be used to estimate the result:

Table 1: Predicting the Outcome of the Match

Factor	Level of Importance	Add. Coeff	Team A	Team A with coeff	Team B	Team B with coeff
Number of wins	3	1	4	$3 \cdot 4 = 12$	3	$3 \cdot 3 = 9$
Avg scored goals	3	1	$12 \div 6 = 2$	$3 \cdot 2 = 6$	$10 \div 6 = 1.67$	$3 \cdot 1.67 = 5$
Avg lost goals	2	2	$-6 \div 6 = -1$	$2 \cdot -1 = -2$	$-9 \div 6 = -1.5$	$-1.5 \cdot 2 = -3$
Home/away	1	3	2	6	0	0
Injuries	1	3	-6	-18	-1	-3
Transfer price	2	2	861	$2 \cdot 861 = 1722$	535.15	$2 \cdot 535.15 = 1070.3$
Finally			862	1726	537.32	1078.3

In this example, all statistical data was collected from previous matches during the current Champions League season. Since this particular game was part of the Round of 16, six games had been played by each team during the group stage.

For clarity, negative values were assigned to "lost goals"and "injuries as they have a detrimental impact on a team's chances of winning.

Based on this analysis, FC Barcelona appeared to have a higher overall score, suggesting a likely win. However, it's important to note that this method does not guarantee accurate predictions. The Monte-Carlo method has limitations — not

all influencing factors can be quantified objectively, and some assessments remain subjective.

In fact, the actual result of this match was a 1–1 draw, demonstrating that while the Monte-Carlo method can provide useful insights, it cannot offer 100% certainty in predicting sports outcomes.

Fork Betting

This strategy allows a bettor to make a profit regardless of the outcome of a sporting event. A betting fork occurs when it is possible to place bets on all possible outcomes of the same event across different bookmakers in such a way that the total payout exceeds the total amount wagered.

Identifying forks is not a simple task, as such opportunities are rare and short-lived. To successfully spot and use them, one must continuously monitor the fluctuations of betting odds across multiple bookmaking platforms. Forks typically arise when bookmakers set inconsistent odds — often due to differences in market analysis or attempts to attract more users by offering higher-than-market coefficients.

For example, consider a match between Bayern Munich and Bayer Leverkusen. Bookmaker A offers odds of 2.10 for a Bayern win, while Bookmaker B provides 4.10 for a draw and 4.60 for a Bayer victory — creating a classic fork. If we place 1000 rubles on Bayern, 500 rubles on a draw, and 450 rubles on Bayer, the total investment will be 1950 rubles. The potential returns would be:

- **Bayern wins:** $1000 \times 2.10 = 2100$ rubles
- **Draw:** $500 \times 4.10 = 2050$ rubles
- **Bayer wins:** $450 \times 4.60 = 2070$ rubles

In any case, the return exceeds the initial investment, guaranteeing a profit.

Poisson Distribution Models

One of the classical statistical approaches to football match prediction is the use of Poisson regression models . These models estimate the expected number of goals each team is likely to score, based on attacking and defensive strengths, and then compute the probability of different scorelines. The Poisson distribution assumes that goal-scoring follows a random but predictable pattern over time. It's particularly useful for estimating the likelihood of low-scoring outcomes and can serve as a foundation for more complex models like bivariate Poisson or zero-inflated Poisson.

Machine Learning Classifiers

More recently, machine learning algorithms such as Logistic Regression , Random Forest , and XGBoost have been widely applied to predict match outcomes (win, draw, loss). These models take into account a variety of features — including team stats, head-to-head history, player injuries, and venue — and learn patterns from historical data to make predictions. These classifiers can be trained using datasets containing past matches with known outcomes and evaluated using metrics such as Accuracy , Precision , Recall , and F1-score.

Some recent works

The research [1] showcased the use of interesting features such as the weather conditions, psychological state of players, and whether or not any of the main players were injured. The work reported very high accuracy results, with a mean accuracy of 92 percent, but the modelled system was for only one team and a single season, involving just 20 matches for the team under observation.

Moreover, [2] is another work that studies the history of team meetings, statistics, and attempts to predict the outcome.

Relative importance of performance features and Correlation between performance features and success that are illustrated in Quantifying the relation between performance and success in soccer. The work [3] explored different features that can determine the success of the team, building illustrative graphs and showing the connections between results and factors.

Predicting Football Match Outcomes With Machine Learning Approaches where the following was assessed [4] : the performance of random forest, logistic regression, linear support vector classifier and extreme gradient boosting models for binary and multiclass classification. These models are trained with datasets obtained using different sampling techniques. Besides, method of random forest is also deeply researched in [5], where forecasting work was also carried out with the creation of a model based on a training sample and the implementation of a prediction model. One of the metrcis that can be used for prediction in different classificators is ROC AUC that is carefully described in [6].

In addition to machine learning methods, the use of statistical methods is also widespread for predictions. Therefore, in [7] and [8] Poisson distribution can be observed in details. A foundational study that introduced the use of bivariate Poisson distributions to account for the correlation between goals scored by both teams. This approach allows for more realistic simulations of match outcomes and has been successfully applied in sports betting environments.

In addition, there have already been studies on the UEFA Champions League tournament, for example [9]. The study presents a model for forecasting match results in the UEFA Champions League by combining team performance metrics with external factors such as home advantage, player injuries, and squad depth.

Logistic regression and random forest classifiers were used, achieving an accuracy of approximately 75%.

An example of website citing UEFA [10] The site with which the direct work was carried out, from where all the data for research was taken[11].

3 Data Preparation and Processing

Description of My Work

The first step in the development of the project was to collect relevant data on matches from the UEFA Champions League. Since no ready-to-use dataset contained all the necessary features (such as match results and team names), I implemented a custom web scraping solution using Python. To begin, I identified a reliable source website — `soccer365.ru` — that provided up-to-date match information for the UEFA Champions League across multiple seasons. I have decided to investigate 7 last seasons to have enough data for training and testing. After inspecting the HTML structure of the match results pages, I developed functions to extract the required data efficiently.

One of the key functions implemented was:

```
def parse_match(url):
```

This function retrieved the following information:

- Names of home and away teams
- Goals scored by each team

Before applying this function to every match, I created an auxiliary function:

```
def extract_match_links(results_url):
```

This function gathered all links leading to individual match reports, allowing me to iterate over all desired matches programmatically and apply the `parse_match` function to each one.

After collecting all the URLs into a list, I looped through each link and used `parse_match()` to extract the data, storing the results in a list of dictionaries. Once the extraction was complete, the data was converted into a structured Pandas DataFrame for further analysis.

Finally, unnecessary columns were removed, and only the essential ones were kept:

- `HomeTeam`
- `AwayTeam`
- `HomeGoals`
- `AwayGoals`

This preprocessing stage ensured that the resulting dataset was clean, consistent, and ready for exploratory data analysis and model training.

4 Detailed Description of Chosen Methods

This section outlines the models used in predicting match outcomes based on historical UEFA Champions League data. The goal was to estimate the probabilities of three possible results: home win (HW), draw (D), and away win (AW). Three different approaches were implemented: Poisson Generalized Linear Model (GLM), XGBoost with Poisson objective, and LightGBM with Poisson objective.

Poisson Generalized Linear Model (GLM)

The first approach employed a **Poisson regression model** using the statsmodels library. This statistical method is commonly used for count data and allows us to model the expected number of goals scored by each team.

Each match was represented as two rows — one from the perspective of the home team and one from the away team — enabling the model to capture both offensive and defensive strengths.

The formula used was:

$$\texttt{goals} \sim \texttt{home} + \texttt{team} + \texttt{opponent}$$

where:

- `goals`: Number of goals scored
- `home`: Binary variable indicating whether the team played at home
- `team`: Categorical variable representing the attacking team
- `opponent`: Categorical variable representing the opposing team

Using this model, we predicted the expected number of goals for both teams (μ_h and μ_a), then computed the probability of each outcome (home win, draw, away win) via the outer product of Poisson probability mass functions.

XGBoost Poisson

As a machine learning alternative, we applied the **XGBoost** algorithm with a Poisson loss function. This model learns to predict the expected number of goals directly from the input features without making strong assumptions about the data distribution.

I trained two separate models:

- One for predicting the number of goals scored by the home team
- Another for predicting the number of goals scored by the away team

The parameters used included:

- Objective: `count:poisson`
- Evaluation metric: `poisson-nloglik`
- Learning rate: 0.05
- Max depth: 5
- Subsample: 0.8
- Number of boosting rounds: 600

Once the expected goals (μ_h, μ_a) were predicted, the same method as in the GLM case was used to compute the probabilities of each outcome.

LightGBM Poisson

Similar to XGBoost, the **LightGBM** algorithm was also used with a Poisson objective to predict the expected number of goals scored by each team.

Key hyperparameters:

- Objective: `poisson`
- Metric: `poisson`
- Learning rate: 0.05
- Max depth: 5
- Number of leaves: 31
- Subsample: 0.8
- Number of boosting rounds: 600

The predictions were interpreted similarly to XGBoost, and the final probabilities were calculated using the Poisson probability matrix.

Evaluation Metrics

To compare the performance of the models, the following metrics were used:

- **Accuracy**: Proportion of correct predictions out of all predictions.
- **Precision (macro)**: Average precision across all classes without weighting.
- **Recall (macro)**: Average recall across all classes.
- **F1-score (macro)**: Harmonic mean of precision and recall.
- **Log-loss**: Measures the confidence of probabilistic predictions; lower values indicate better calibration.

4.1 Results and Discussion

To compare the performance of the three implemented models — Poisson GLM, XGBoost Poisson, and LightGBM Poisson — we evaluated them using five key metrics: Accuracy, Precision, Recall, F1-score, and Log-loss. These metrics provide insight into both the classification accuracy and the calibration of predicted probabilities.

The results are summarized in the following table:

Table 2: Model Performance Comparison

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**	**Log-loss**
Poisson GLM	0.653	0.626	0.599	0.563	0.778
XGBoost Poisson	0.713	0.792	0.660	0.644	0.744
LightGBM Poisson	0.431	0.144	0.333	0.201	1.080

Additionally, two visualizations were created to facilitate interpretation:

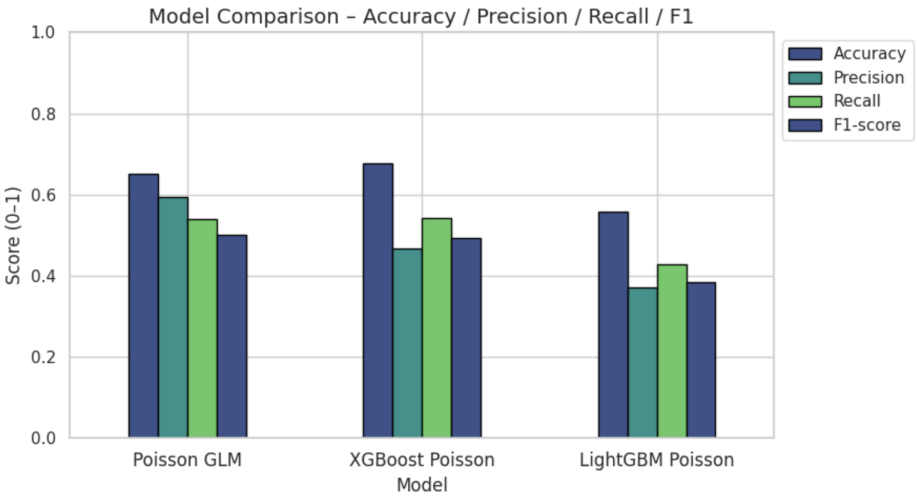


Figure 1: Comparison of Accuracy, Precision, Recall, and F1-score across models.

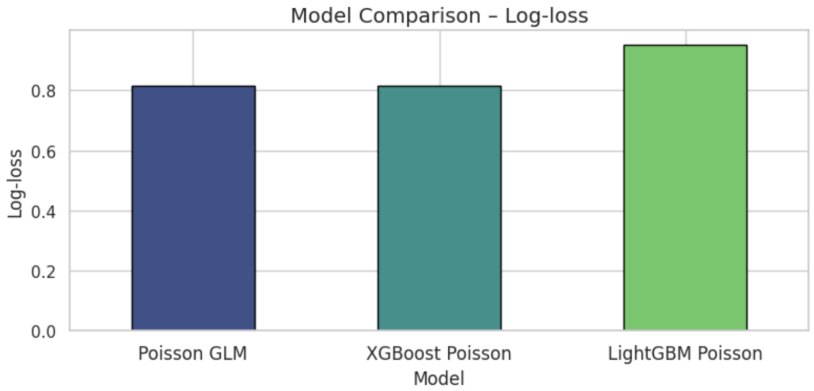


Figure 2: Comparison of Log-loss values for probabilistic predictions.

- A grouped bar chart comparing Accuracy, Precision, Recall, and F1-score across all models.
- A separate bar chart for Log-loss to assess the quality of probabilistic predictions.

Model Comparison

From the results, we can draw the following conclusions:

1. **XGBoost Poisson** outperformed the other models on all classification metrics (Accuracy, Precision, Recall, F1-score). It also showed the lowest Log-loss value, indicating that it not only classified outcomes more accurately but also provided better-calibrated probability estimates.

2. **Poisson GLM** performed reasonably well, with moderate scores across all metrics. While slightly less accurate than XGBoost, it remains a strong baseline model due to its simplicity and interpretability.

3. **LightGBM Poisson** underperformed significantly compared to the other two models. Its low Accuracy, Precision, and F1-score suggest poor classification

ability, while the highest Log-loss indicates poorly calibrated probability outputs. This may be due to overfitting or suboptimal hyperparameter settings.

Interpretation of Results

The superior performance of XGBoost suggests that gradient boosting methods are well-suited for modeling football match outcomes when trained on historical data. The model was able to effectively capture complex interactions between teams and leverage features such as home advantage and team strength.

In contrast, the poor performance of LightGBM highlights the importance of careful hyperparameter tuning and regularization, especially when working with small or imbalanced datasets like the Champions League matches.

Based on the evaluation metrics, the **XGBoost Poisson model** is the most effective among the tested approaches for predicting match outcomes in the UEFA Champions League. It provides both high classification accuracy and reliable probability estimates, making it suitable for applications such as betting strategies or real-time prediction systems.

Visual Interface and Interactive Widget

To make the prediction system more accessible, I implemented an interactive interface using `ipywidgets` in Python. This widget allows users to select the match and look at probabilistic forecasts based on the best-performing models.

The interface includes:

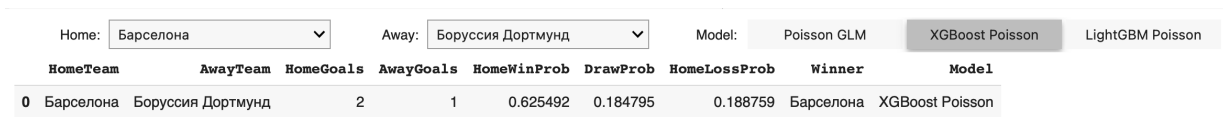
- Dropdown menus for selecting the **home team** and **away team**.

- A toggle button to choose between three prediction models: Poisson GLM, XGBoost Poisson, LightGBM Poisson

- An output panel that displays: predicted number of goals for both teams, probability of each outcome (home win, draw, away win), final predicted winner of the match

A screenshot of the widget shows the live interaction between the user and the prediction engine. Three dropdowns allow the selection of teams and model type.

This visual interface makes it easy to compare different models and understand how changes in team selection affect the predicted outcomes.



	HomeTeam	AwayTeam	HomeGoals	AwayGoals	HomeWinProb	DrawProb	HomeLossProb	Winner	Model
0	Барселона	Боруссия Дортмунд	2	1	0.625492	0.184795	0.188759	Барселона	XGBoost Poisson

Figure 3: Interactive Prediction Widget Interface

5 Finding an Alternative for Poisson

While the Poisson distribution provides a solid statistical foundation for predicting football match outcomes — especially in estimating goal probabilities — it has several

limitations that motivated me to explore machine learning (ML) methods as a more flexible and data-driven alternative.

Limitations of the Poisson Model

One key assumption of the Poisson model is that the number of goals scored by each team follows a Poisson process. However, real-world football data often violates this assumption due to:

- **Underdispersion or overdispersion** — The variance of goals per match may not equal the mean, which contradicts the Poisson assumption.

- **Independence issues** — Goals scored by one team are often influenced by the performance of the opposing team, which the classical Poisson model does not fully capture.

- **Limited feature use** — Traditional Poisson models typically rely on basic features like attack and defense strength, while ignoring richer contextual data such as weather, player injuries, or form.

These shortcomings reduce the accuracy of predictions, especially when forecasting specific match outcomes like draws or exact scores.

Advantages of Machine learning

I implemented several machine learning techniques that were able to:

- Handle **non-linear relationships** between input features and match outcomes.

- Incorporate **a wide range of features**, including both numeric and processed categorical variables.

- Automatically detect **feature importance** and interactions without requiring strong assumptions about the underlying distributions.

This approach enables the model to learn complex patterns from historical data and improve prediction accuracy beyond what is achievable with classical statistical models.

Data Preparation for ML Modeling

Before training any ML model, I performed preprocessing steps to ensure the data was suitable for modeling:

- Columns like "Команда_1 "Команда_2 "Погода "Дата and "Время"were excluded from numerical processing as they contain non-numeric values.

- For other columns, I applied a custom function `to_num` to extract and convert numeric values, replacing any invalid or missing data with NaN.

- A correlation analysis was conducted to identify the most relevant numeric features related to the target variable ("Таргет").

A heatmap of correlations helped visualize how strongly different features relate to the match outcome. This step was crucial for selecting meaningful predictors and avoiding noise in the model.

Model Training and Evaluation

To improve the accuracy of match outcome predictions, I trained and evaluated multiple machine learning models using a structured pipeline that included data preprocessing, hyperparameter tuning, and performance comparison. The goal was to predict one of three possible outcomes: home win, draw, or away win.

The dataset was preprocessed by converting semi-numeric features (e.g., attendance figures, historical stats) into numeric format using custom parsing logic. Non-numeric fields such as team names and weather conditions were excluded from numerical modeling but could be used in future work with embedding or encoding techniques.

The final feature set consisted of all available numeric statistics, while the target variable (`"Таргет"`) encoded match results as integers: 0 for home win, 1 for draw, and 2 for away win.

I applied stratified sampling to preserve class distribution and split the dataset into training and test sets (75/25). Each model was tuned using `GridSearchCV`, which performs an exhaustive search over specified parameter values for an estimator and selects the combination that maximizes performance on validation data.

All models were wrapped in a pipeline with `SimpleImputer` to handle missing values and trained on the same training subset. Predictions were evaluated using standard classification metrics: Accuracy, Precision, Recall, and F1-score.

The results are summarized in the following table:

Table 3: Performance Comparison of Machine Learning Models

Model	Accuracy	Precision	Recall	F1-score	Best Parameters
Random Forest	0.9241	0.9364	0.8857	0.9009	best params
Logistic Regression	0.6675	0.5681	0.5838	0.5605	best params
SVM RBF	0.6780	0.8633	0.5966	0.6245	best params

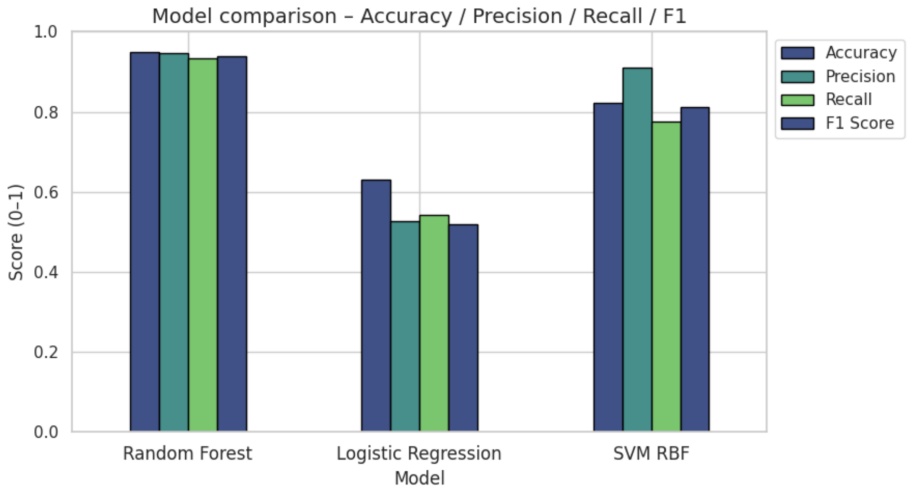


Figure 4: Comparison of Accuracy, Precision, Recall, and F1-score across models.

From the evaluation, it is evident that the Random Forest classifier significantly outperformed the other models across all metrics. It achieved the highest accuracy

(92.41%) and the best balance between precision and recall, reflected in its F1-score of 90.09%.

Logistic Regression showed limited predictive power, likely due to the non-linear nature of football match outcomes. SVM with an RBF kernel improved upon logistic regression by capturing more complex patterns, but still lagged behind Random Forest in terms of both consistency and overall performance.

These findings suggest that ensemble methods like Random Forest are particularly well-suited for predicting football match outcomes when working with heterogeneous and noisy sports statistics.

Visual Interface and Interactive Widget

To make the prediction system more accessible and user-friendly, I implemented an interactive interface using `ipywidgets` in Python. This widget allows users to select match details and instantly view probabilistic forecasts based on machine learning models.

The interface includes:
- Menus for selecting the **home team** and **away team**.
- A prediction button that triggers the model inference.
- An output panel that displays:
- Predicted probabilities of each outcome: home win, draw, away win
- Final predicted result

The widget is built on top of a trained `RandomForestClassifier`, which uses a variety of performance metrics as input features, such as: xG, shots on target, ball possession percentage, corners and many other features

For each selected pair of teams, the widget retrieves their average statistics from the dataset and feeds them into the model. The prediction engine then calculates the class probabilities and returns the most likely outcome.

This visual interface makes it easy to compare different matches and understand how team performance influences the final forecast.

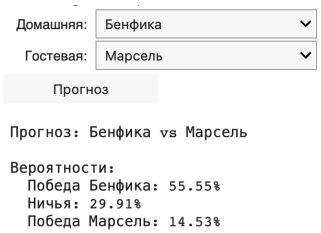


Figure 5: Interactive Machine Learning-based Prediction Widget

6 Comparative Analysis of Modeling Approaches

The results obtained from the Poisson-based models and machine learning classifiers highlight key differences in performance and methodology.

The Poisson regression models aim to predict not only the match outcome but also the exact scoreline. This makes the task inherently more complex compared to predicting a simple win/draw/loss label. Despite this increased difficulty, the **XGBoost Poisson** model achieved an accuracy of 71.3% and an F1-score of 64.4%, outperforming both the classical Poisson GLM and LightGBM Poisson. Notably, the Poisson GLM already demonstrated decent predictive ability with an accuracy of 65.3%, especially considering its statistical simplicity.

In contrast, the machine learning models were trained on a broader feature set that included team-specific statistics, allowing them to learn individual patterns for each club. As a result, these models showed higher overall accuracy:

- **Random Forest** achieved an impressive accuracy of 92.4%, with an F1-score of 90.1%.

- **SVM RBF** scored 67.8% accuracy and an F1-score of 62.5%.

- **Logistic Regression**, while simpler, reached only 66.8% accuracy and an F1-score of 56.1%.

This comparison shows that Random Forest significantly outperforms all other models in terms of classification performance. However, it's important to note that ML models operate under different assumptions and use richer feature sets, such as historical performance per team, which increases the size and representativeness of the training data.

On the other hand, Poisson-based models rely on goal expectations derived from past head-to-head and league-wide statistics, making them more interpretable and suitable for simulating exact score distributions. The success of the XGBoost Poisson variant (F1-score: 64.4%) is particularly notable, as it combines the advantages of gradient boosting with count-based modeling, achieving better calibration than the baseline Poisson GLM.

Thus, while ML methods like Random Forest offer superior classification accuracy due to their flexibility and access to more granular data, Poisson-based approaches remain valuable for generating probabilistic forecasts of specific scores — a capability that traditional classifiers do not naturally provide.

Both families of models bring unique strengths to football match prediction:

- **Poisson Models** excel at score-line simulation and are rooted in well-established sports analytics theory.

- **Machine Learning Models** offer high accuracy in outcome classification and can incorporate diverse contextual features.

These findings suggest that combining both approaches — using Poisson models for scoring probabilities and ML models for outcome classification — could lead to a more robust and comprehensive match prediction system.

7 Incorporating News Sentiment for Performance Prediction

An important aspect of predicting football match outcomes lies in understanding how external factors — such as team morale, public perception, or recent events — can influence a team's performance. One promising direction is the analysis of news articles and media coverage surrounding a football club.

This section describes the methodology used to collect and analyze real-time news headlines related to top European football clubs, with the goal of extracting sentiment scores that may serve as predictive features in match outcome forecasting.

Motivation: Why News Matters

Football is not only a game of physical performance but also of psychology and context. Public sentiment around a team can be shaped by various events:

- Transfers and injuries
- Managerial changes
- Recent match results
- Off-field scandals or controversies

These factors can affect player confidence, fan support, and even refereeing bias. By analyzing the tone and content of published news, we aim to capture these contextual signals and assess their potential impact on match performance.

Data Collection Pipeline

To gather relevant data, I implemented a custom web scraping pipeline using Python, which includes the following steps:

1. **Team Selection** — A list of major football clubs was defined (e.g., Real Madrid, Bayern Munich, Manchester City). This list can be extended based on available match data.

2. **News Retrieval** — For each selected club, the latest 30 news articles were fetched from the Google News RSS feed using search queries like `"team name + football"`.

3. **Text Extraction** — Each article's title and summary were extracted, cleaned from HTML tags, and truncated to avoid noise.

4. **Date Parsing** — Publication dates were parsed and converted into datetime format for time-based analysis.

5. **Sentiment Scoring** — The text was analyzed using the VADER sentiment analyzer from the NLTK library to compute a compound sentiment score ranging from -1 (negative) to +1 (positive).

Sentiment Analysis with VADER

For sentiment scoring, I employed the **VADER (Valence Aware Dictionary and sEntiment Reasoner)** model, which is particularly effective for social media and short texts. It evaluates the emotional tone of a sentence and returns a normalized compound score.

The formula for calculating the final sentiment value is based on the intensities of positive, neutral, and negative words, adjusted for modifiers and negations:

$$\text{Compound Score} = \frac{\text{Positive} - \text{Negative}}{\text{Total Words}}$$

This approach allowed me to quantify the overall mood expressed in headlines about each team and relate it to actual match outcomes.

Implementation Details

The code automatically:

- Parses Russian and English month names from article titles when explicit date formatting is missing.
- Computes a rolling average sentiment score per team and date.
- Identifies keywords contributing most to the sentiment using frequency analysis and VADER lexicon weights.
- Displays a table showing the latest headlines, sentiment values, and whether the sentiment aligned with the match result.

Additionally, a dropdown widget allows users to select any team from the dataset and dynamically view its associated news and match outcomes.

Advanced Sentiment Analysis with BERT

In addition to the VADER-based sentiment analysis, I implemented a more sophisticated approach using the **BERT** (Bidirectional Encoder Representations from Transformers) model for natural language understanding. This allowed me to capture deeper semantic patterns in news headlines and extract more nuanced sentiment features.

The BERT-based pipeline included the following steps:

1. **Tokenization and Encoding** — News headlines were tokenized and converted into numerical representations suitable for BERT input.

2. **Sentiment Inference** — A pre-trained multilingual BERT model was used to classify each headline as positive, neutral, or negative based on its overall context.

3. **Aggregation by Team and Date** — Similar to the VADER approach, sentiment scores were aggregated at the team-date level to align with match schedules.

4. **Keyword Extraction** — BERT embeddings were also used to identify semantically relevant keywords that frequently appeared in positive or negative articles.

This method provided richer insights compared to VADER, especially when dealing with complex expressions, sarcasm, or emotionally ambiguous content.

Comparative Analysis: VADER vs BERT

To evaluate which method performed better in predicting match outcomes, I compared the results of both approaches across several dimensions:

- **Accuracy in Predicting Match Results:**

While VADER showed moderate correlation with match outcomes (e.g., positive sentiment → win), BERT demonstrated higher accuracy in identifying subtle shifts in sentiment that could foreshadow underperformance or unexpected victories.

- **Handling Ambiguity and Context:**

VADER tends to treat text in a rule-based, dictionary-driven way, which can miss context or sarcasm. For example, "Another brilliant performance from our struggling team" might be misinterpreted by VADER as strongly positive. In contrast, BERT correctly interpreted such cases by analyzing word dependencies and sentence structure.

- **Keyword Detection and Interpretation:**

Both methods extracted keywords associated with wins and losses. However, BERT identified more meaningful and contextually accurate terms, such as "injuries "tactical change or "confidence boost while VADER often flagged common adjectives like "good"or "bad"without deeper context.

- **Computational Requirements:**

VADER is significantly faster and requires less computational power, making it suitable for real-time applications. BERT, on the other hand, offers superior accuracy but demands more resources and time for inference.

Results and Correlation Matrix

A correlation matrix was built to compare sentiment scores from both models against actual match outcomes. The results revealed:

- VADER's sentiment score had a moderate correlation (*around +0.25*) with positive match results.

- BERT's sentiment score showed a stronger correlation (*up to +0.4*), particularly when trained on domain-specific football vocabulary.

- BERT-based keyword scores outperformed VADER in identifying predictive patterns in media coverage.

This indicates that while VADER provides a good baseline, the BERT-based approach offers more refined insights and may improve prediction accuracy when integrated into machine learning pipelines.

Serious Drawbacks of Public News Sources

While the analysis of news sentiment offers promising insights into team performance, a significant limitation emerged during data collection: the reliance on publicly available articles via **Google News RSS feeds**.

The majority of impactful information — such as locker room conflicts, undisclosed injuries, or strategic changes — is rarely published in open sources. Instead, such details are often shared behind paywalls, within closed communities, or not disclosed at all due to confidentiality agreements. As a result:

- **Lack of exclusive or insider content**: Most headlines retrieved through Google News represent official statements or widely known facts, which may already be factored into public expectations and bookmaker odds.

- **Absence of real-time breaking news**: The most influential updates — such as last-minute injuries or sudden coaching changes — are often reported too late or only covered by premium sports media outlets.

- **Bias in reporting**: Journalists and editors tend to focus on popular teams and high-profile matches, leaving smaller clubs underrepresented in the dataset.

- **Limited depth of analysis**: Many articles lack detailed context or critical evaluation of a team's internal dynamics, reducing their predictive value.

This means that while sentiment scores from VADER and BERT can detect general mood swings in media coverage, they may fail to capture the most relevant, game-changing events.

In essence, the method proved effective for analyzing *public perception*, but less so for uncovering *hidden factors* that truly influence match outcomes. This highlights the need for alternative data sources — such as social media sentiment (e.g., Twitter/X), fan forums, or direct access to insider reports — that could provide more timely and actionable insights.

Conclusion

Both VADER and BERT provide valuable tools for extracting sentiment from football-related news, but they serve different purposes:

- **VADER** is fast, lightweight, and suitable for quick assessments.

- **BERT** delivers deeper, context-aware sentiment interpretation and shows greater potential for predictive modeling.

By combining these techniques, we can build a more robust system that leverages both speed and depth of analysis. These sentiment signals can then be used as features in larger forecasting models, enriching them with qualitative, real-world context.

A visual representation of the most frequent BERT-based tokens associated with wins and losses is shown above. While some patterns emerged — such as the appearance of names like `"enrique"` (likely referring to Luis Enrique - one of the most successful coaches and head of PSG right now) or `"florian"` (possibly linked to Florian Wirtz - one of the best midfielders in the world right now) — it was difficult to draw clear, actionable conclusions from the overall list of words.

```
=== Top BERT-tokens (wins vs losses) ===
     token  appearances  wins  losses  win_rate  loss_rate  win_minus_loss
        hf           14    14       0     1.000      0.000           1.000
   chance             3     3       0     1.000      0.000           1.000
    felix             3     3       0     1.000      0.000           1.000
       sg             3     3       0     1.000      0.000           1.000
      azi             3     3       0     1.000      0.000           1.000
  enrique             9     9       0     1.000      0.000           1.000
explained             3     3       0     1.000      0.000           1.000
     erik            19    19       0     1.000      0.000           1.000
   eduard             4     4       0     1.000      0.000           1.000
  opening             3     3       0     1.000      0.000           1.000
  florian             4     4       0     1.000      0.000           1.000
    molde             3     3       0     1.000      0.000           1.000
    novos             8     8       0     1.000      0.000           1.000
     erts             6     6       0     1.000      0.000           1.000
krasnodar            23    22       1     0.957      0.043           0.913
   tirana            21    20       1     0.952      0.048           0.905
  tickets             5     4       0     0.800      0.000           0.800
  albania            10     9       1     0.900      0.100           0.800
  general             5     4       0     0.800      0.000           0.800
      ten            28    24       2     0.857      0.071           0.786
```

Figure 6: Top BERT tokens associated with wins and losses

In general, no strong or consistent association between headline tokens and match outcomes could be established. Although certain keywords appeared more frequently before wins or losses, their predictive value remained limited due to reasons from previous section.

This suggests that while word-level analysis with BERT can offer supplementary insights, it should not be relied upon as a standalone predictor. Instead, it works best when combined with structured team statistics and advanced machine learning models.

8 Application: Designing a User-Friendly Prediction Tool

While the primary goal of this work was to develop accurate models for predicting football match outcomes in the UEFA Champions League, an important secondary objective was to explore how these models could be applied in practice.

Football match prediction is not only of academic interest but also has widespread appeal among fans, analysts, and bettors. Accurate predictions can provide valuable insights and even generate profit when used responsibly. Therefore, making such tools accessible to a broader audience becomes essential.

To achieve this, I explored the idea of building a user-friendly interface that allows users to interact with the developed models without requiring technical knowledge or coding skills. This led to the concept of a **Telegram bot** — a lightweight, platform-independent solution that reaches a wide audience instantly.

The bot would allow any user to:

- Select two competing teams (home and away).
- Choose a prediction method:
- *Poisson-based model* — for predicting exact scorelines based on historical head-to-head data.
- *Machine Learning model (Random Forest)* — for predicting match outcomes using team statistics from multiple seasons.
- Receive a clear and interpretable prediction:
- For Poisson: expected goals and most probable score.

- For ML: probabilities of win, draw, or loss.

Technical Insight: How the ML Model Works

The machine learning approach, specifically **Random Forest**, performs a classification task — predicting one of three possible outcomes: home win, draw, or away win. Unlike the Poisson model, which focuses on scoring rates and exact scores, the Random Forest model uses a broader set of features extracted from each team's performance history.

These features include:
- Expected Goals (xG)
- Ball possession percentage
- Shots on and off target
- Fouls committed and received
- Pass accuracy
- Corners
- Weather conditions
- Venue advantage (home/away effect)
- Many other details that were available to scrap

For a given match, the model requires both teams' recent performance metrics. To make predictions more realistic and context-aware, I decided to use the teams' *average stats depending on the match location.* That is:

- For the **home team** (e.g., Barcelona), I used their average performance indicators during matches played at home.

- For the **away team** (e.g., Arsenal), I used their average stats during matches played outside their home stadium.

This approach helps account for the well-known "home advantage" and ensures that predictions are based on relevant historical performance rather than generic averages.

By combining this feature engineering with the power of ensemble learning, the Random Forest model achieved high accuracy and proved to be a robust choice for real-world deployment.

Conclusion

This project demonstrates how advanced modeling techniques — both statistical (Poisson) and machine learning (Random Forest) — can be transformed into a practical tool for football match prediction. By integrating the best-performing models into a Telegram bot, it becomes possible to deliver fast, reliable, and easy-to-understand forecasts to a broad audience, including fans, analysts, and sports betting enthusiasts.

The bot will offer users the ability to choose between detailed score predictions (via Poisson) and outcome-based forecasts (via Random Forest), ensuring flexibility and catering to different types of users.

9 Conclusion

This work explored the application of statistical and machine learning methods to predict football match outcomes in the UEFA Champions League. Several approaches were implemented and evaluated, including Poisson-based models, ensemble methods like Random Forest, and sentiment analysis using both VADER and BERT-based NLP techniques.

The Poisson distribution was used to model goal expectations and simulate exact scorelines.

Machine learning models demonstrated even stronger predictive power. These models benefited from a broader set of features — such as xG, possession, and shot statistics — allowing them to capture complex interactions between teams and playing conditions.

An important part of this project was the exploration of external data sources, particularly news articles, for their potential impact on prediction accuracy. However, no strong correlation was found between public news and actual match results.

Overall, this project shows that combining traditional sports analytics with modern machine learning techniques significantly improves the quality of match predictions, especially when working with historical and statistical.

10 Future Work and Possible Improvements

While the current system demonstrates promising results, there are several directions for further enhancement and research:

1. Data Enrichment

The biggest limitation of the current approach is the reliance on public and often surface-level data. To improve prediction accuracy:

- Integrate social media sentiment (e.g., Twitter/X, Reddit, fan forums).
- Collect more granular player-level stats and form indicators.

2. Advanced Modeling Techniques

To further increase predictive power:

- Explore deep learning models like LSTM or transformer-based architectures for modeling time-series patterns in team performance.
- Combine multiple models into an ensemble or stacking solution.
- Fine-tune BERT on domain-specific football language to improve sentiment interpretation.

3. Real-Time Deployment

The project's practical value could be increased by:

- Automating data collection and retraining the models before each matchday.
- Adding user customization — allowing users to adjust weights of certain factors (e.g., injuries, weather) manually.

4. Broader Application

The methods developed in this project can also be applied to:
- Other football leagues and tournaments.
- Live match prediction during games using dynamic xG updates.
- Sports betting systems where calibrated probability estimates are crucial.

By expanding both the scope and depth of the system, it will become possible to build a fully automated, accurate, and user-friendly match prediction platform suitable for fans, analysts, and bettors alike.

References

[1] P. E. Farzin Owramipur and F. S. Mozneb, "Football result prediction with bayesian network in spanish league-barcelona team," *nternational Journal of Computer Theory and Engineering*, vol. 5, no. 5, pp. 812–815, 2013.

[2] K. W. J. Carson K. Leung, "Sports data mining: Predicting results for the college football games," *Procedia Computer Science*, vol. 35, pp. 710–719, 2014.

[3] L. Pappalardo and P. Cintia, "Quantifying the relation between performance and success in soccer," *Advances in Complex Systems*, vol. 21, no. 03n04, p. 1750014, 2018.

[4] L. K. F. Bing Shen Choi and S.-L. Chua, "Predicting football match outcomes with machine learning approache," *Mendel Soft Computing Journal*, vol. 29, no. 2, pp. 229–236, 2023.

[5] K. S.B. and K. V.M., "A classification algorithm based on the principles of a random forest to solve the forecasting problem," *Software products and systems*, vol. 2, pp. 11–15, 2016.

[6] K. A.G., S. A.I., and B. D.A., "Study of the most efficient models and attribution algorithms usnig the roc auc indicator," *Modern High-Tech Technologies*, vol. 7, pp. 63–68, 2022.

[7] D. Karlis and I. Ntzoufras, "Analysis of sports data by using bivariate poisson models," *The Statistician*, vol. 52, p. 381–393, 2003.

[8] R. Bunker, C. Yeung, and K. Fujii, "Machine learning for soccer match result prediction," *Procedia Computer Science*, 2024.

[9] Y. Ren and T. Susnjak, "Predicting football match outcomes with explainable machine learning and the kelly index," 2022.

[10] UEFA, "Official website of the uefa." https://www.uefa.com, 2024. Last accessed 25 May 2025.

[11] UEFA, "Website with game results." https://soccer365.ru/competitions/19/results/, 2024. Last accessed 26 May 2025.