Spatio-Temporal Football Match Outcome Prediction

DSP391m - Final Report

Dinh Ngoc An[1], Phan Van Phuc[2], and Nguyen Minh Dat[3]

[1,2,3]Department of Information Technology, FPT University

06th July 2024

1 Introduction and Objective

The core team for this project consists of three AI experts from FPT University, each bringing a unique perspective and skillset:

Team Member	Role	Email Address
Phan Van Phuc	AI Engineer	phucpvse170209@fpt.edu.vn
Nguyen Minh Dat	Data Engineer	datnmse170570@fpt.edu.vn
Dinh Ngoc An	Data Scientist	andnse171386@fpt.edu.vn

Table 1: Team Members, Roles, and Email Addresses

Our mission with this project is to gather data from various open-source soccer analytics websites and perform complex interactions with our collected data to turn it into usable data. We then aim to gather insights into the numerical records of each soccer event and build a machine-learning model to predict some events given these records or more specifically are goal prediction and goal position prediction.

The data processing, analysis, and modeling methodology presented in this report are derived from the code and data available in the GitHub repository: Github Link

2 Data Processing

2.1 Data Collection & Preparation

The primary data source for this project was a collection of match data, supplemented with data from Wyscount. To facilitate effective machine learning and avoid data leakage, the dataset was strategically divided into training and testing subsets.

- **Training Set:** Matches from domestic leagues in England, France, Germany, Italy, and Spain form the training set. This diverse range of leagues provides a comprehensive representation of team styles, player abilities, and tactical approaches across different footballing cultures.
- **Test Set:** To evaluate model performance on unseen data and different competitive settings, the test set consists exclusively of matches from major international tournaments, such as the European Championship and World Cup. These tournaments feature a unique blend of national team dynamics, heightened stakes, and distinct playing styles, offering a challenging benchmark for our models.

The dataset's structure includes a wide array of features, including unique identifiers for events and sub-events, tag IDs for various in-game occurrences, comprehensive player profiles, and finally detailed match information (e.g., location, date, outcome)

Figure 1 provides an overview of the types of events captured in the dataset. A more comprehensive description of the dataset is available in Appendix B.

2.2 Data Cleaning and Preprocessing

The data provided by Wyscout is stored in a JSON format similar to a Relational Database with a foreign key connecting the two. It is inappropriate to develop a model using the data current format. Our objective is to merge three datasets into a single comprehensive dataset that includes essential features contributing to the events. Furthermore, the data must be refined into a more appropriate form such as a CSV file to facilitate the model creation process.

The dataset is subsequently enriched by incorporating contextual information pertaining to the matches and the event initiating player. This enrichment process includes data on the player's side affiliation within the match (home or away), as well as their fundamental anthropometric characteristics, specifically their dominant foot, height, and weight.

As part of data preprocessing, missing values were identified and addressed such as unidentified player having id value of 0, and unlabeled events with no tag. Importantly, some football goal data points were

Event	SubEvent	Label (event – subevent)
1	10	**Duel** – Air duel
1	11	**Duel** – Ground attacking duel
1	12	**Duel** – Ground defending duel
1	13	**Duel** – Ground loose ball duel
2	20	**Foul** – Foul
2	21	**Foul** – Hand foul
2	22	**Foul** – Late card foul
2	23	**Foul** – Out of game foul
2	24	**Foul** – Protest
2	25	**Foul** – Simulation
2	26	**Foul** – Time lost foul
2	27	**Foul** – Violent Foul
3	30	**Free Kick** – Corner
3	31	**Free Kick** – Free Kick
3	32	**Free Kick** – Free kick cross
3	33	**Free Kick** – Free kick shot
3	34	**Free Kick** – Goal kick
3	35	**Free Kick** – Penalty
3	36	**Free Kick** – Throw in

Figure 1: The figure provides examples of events and sub-events within the dataset. Events typically represent specific player actions, while sub-events describe the outcomes or details of those actions.

recognized as outliers. To preserve the dataset's authenticity and ensure the model's ability to handle real-world variability, these outlier values were not discarded but were intentionally included and accounted for during model training.

In the data cleaning and preprocessing phase, the dataset was tailored for the two distinct prediction tasks:

- **For Task 1** (goal prediction), which focuses on predicting whether a given event will result in a goal. To achieve this, the dataset was filtered to events where a goal-scoring opportunity existed: Free Kick, Save attempt, and Shot. Subsequently, tag information was extracted to identify instances where the goal tag was present, forming the basis for feature creation.
- **For Task 2** (goal position prediction), the objective was to predict the precise location within the goal where a successful shot would land. This required a more focused dataset, so preprocessing centered on events explicitly tagged as Goal. Within these goal events, relevant sub-events (e.g., shot type, body part used) were extracted to provide detailed context about the goal-scoring action. This targeted data selection ensures that the model's training and evaluation are specifically tailored to the nuances of goal positioning, ultimately enhancing its predictive power for this specific task.

In addition, the position feature format is changed from two pairs of coordinates to each coordinate value belonging to a separate feature.

2.3 Feature Engineering

In order to improve the Machine Learning models beyond basic coordinates, we need to design and engineer other spatial features based on such coordinates. They allow for deeper analysis as well as model serving.

2.3.1 Shot Distance

This metric measures the distance a shot is from the goal position. To do so, we need some information:

- **Football field dimension:** As different football fields have varying dimensions, we have taken the average of most football fields, and come up with the average dimension of 105×68.

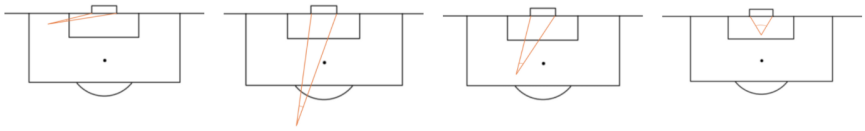


Figure 2: Visualization of shot angles various conditions. Image courtesy of Ian Dragulet.

- **Coordinates translation:** The dataset coordinates are described as "percentage of coordinates" (think of it has how much of the x, y-axis is the ball positioned), we have to translate them to real-life coordinates by scale them by 100 and multiply by the "average dimension".
- **Position of goal-post:** The goal-post position is 100×50, meaning the goal post is 50% the y-axis (meaning the middle) and 100% the x-axis (meaning at either end of the field).

With this information, we can formulate the shot distance as:

$$\text{ShotDistance}(x, y) = \sqrt{\left(\frac{(100 - x) \cdot 105}{100}\right)^2 + \left(\frac{(50 - y) \cdot 68}{100}\right)^2} \tag{1}$$

where x, y are coordinates (in percentage) of the starting position of ball.

2.3.2 Shot Angle

The other metric is shot angle, which measures the angle from the ball to the two sides of the goal post (see Figure 2). Calculation of shot angle is divided into many steps:

- **Step 1:** Translate proportional coordinates to meters.

$$x' = \frac{x \cdot 105}{100} \qquad y' = \frac{y \cdot 68}{100}$$

- **Step 2:** Distance from ball to 2 goal post $\text{post}_1, \text{post}_2$ is calculated using position of ball and width of goal post (set to $w = 7.32m$), then applying Pythagoras theorem.

$$d_1 = d(\text{Ball}, \text{Post}_1) = \sqrt{(105 - x')^2 + \left(34 + \frac{w}{2} - y'\right)^2}$$

$$d_2 = d(\text{Ball}, \text{Post}_2) = \sqrt{(105 - x')^2 + \left(34 - \frac{w}{2} - y'\right)^2}$$

- **Step 3:** Calculate the shot angle θ using the Law of Cosine to the triangle made up of the ball and two goal posts. The angle can be converted to degrees for better interpretation.

$$\cos(\theta) = \frac{d_1^2 + d_2^2 - w^2}{2 \cdot d_1 \cdot d_2} \iff \text{ShotAngle} = \theta = \arccos \frac{d_1^2 + d_2^2 - w^2}{2 \cdot d_1 \cdot d_2}$$

$$\theta_{\text{degree}} = \frac{\theta \cdot 180}{\pi}$$

For numerical stability, we constrain the cosine step within range $[0, 1]$ using `min, max` arguments.

3 Exploratory Data Analysis (EDA)

3.1 Basic Statistics

In general, based on the events data set, a soccer match consists of an average of $1,682 \pm 101$ events, with an inter-time between two consecutive events of 3.59 ± 7.42 seconds. There are on average 59 ± 29 events

observed for a player in a match, one every 78.78 ± 105.64 seconds, confirming that soccer players are typically in ball possession for less than two minutes.

Following data preparation, the final training set contains $3,071,395$ events, while the testing set contains $179,899$ events. After filtering to be more collaborate with tasks, the number of samples training and evaluation for each task include:

- **Task 1:** Training sample is 18382 and evaluation sample is 1196
- **Task 2:** Training sample is 1616 and evaluation sample is 88

3.2 Visualization

We can see that the majority of a football match consists of passing and dueling with both having a sum of over 70%, highlighting a significant class imbalance between normal passes and other events (Figure 3) of the total event, while shots are rarely made during a match. Moreover, most matches have an event count ranging from 1600 to 1800 unique events (Figure 4).

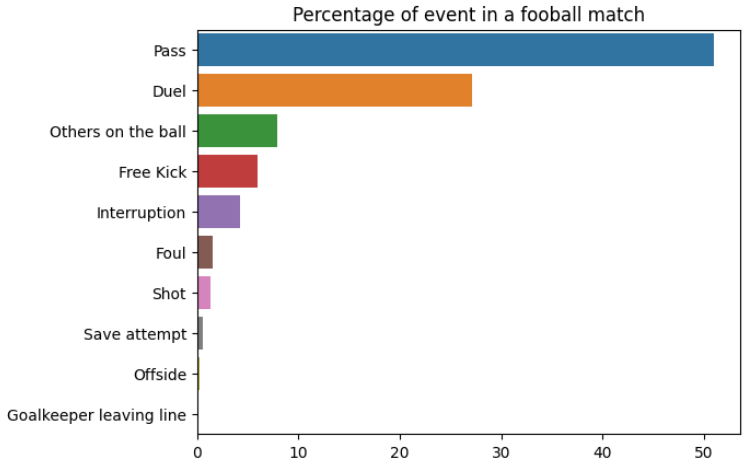


Figure 3: The figure illustrates the distribution of events within a typical football match.

To gain a deeper understanding of our dataset and ensure effective model training, we conducted a visual analysis of the class distribution within our features. This allowed us to identify which features play the most significant role in shaping our data and, consequently, which features would be most impactful in our modeling process You can have more information in Figure 5, Figure 6.

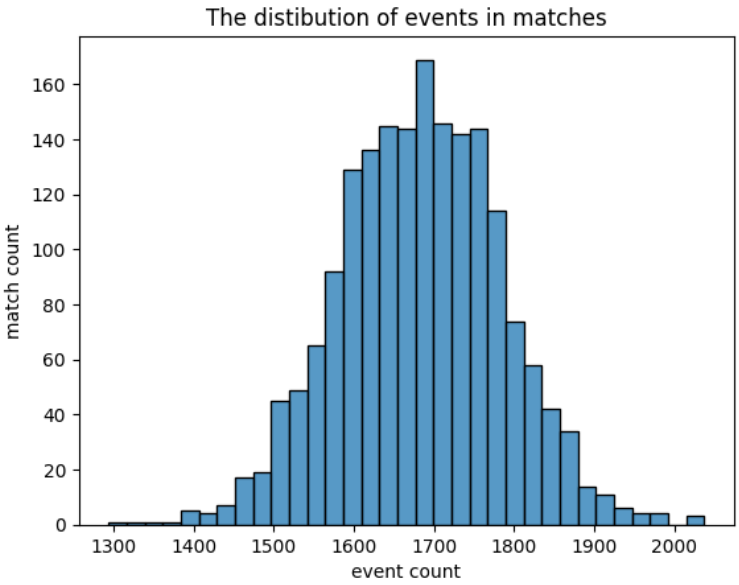
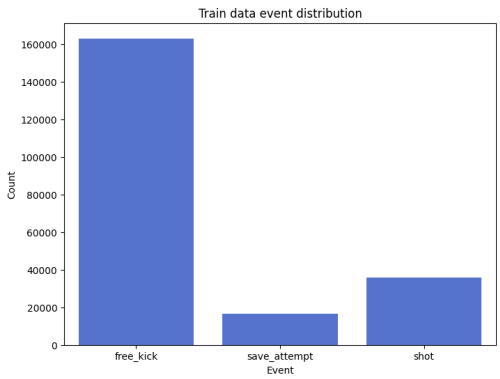
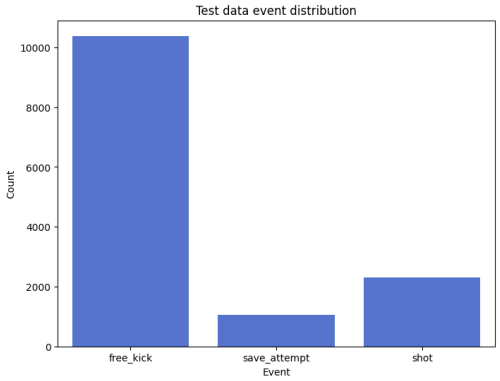


Figure 4: Distribution of events in matches

The purpose of visualizing the correlation between features in a dataset related to two tasks (potentially shot prediction and goal prediction) in which visualize detailed in Table 7 and Table 8. The purpose of figures is to understand the relationships between these features and how they might influence each other. This can provide valuable insights for model development, feature selection, and understanding the underlying factors affecting the target variables. You can have more information in

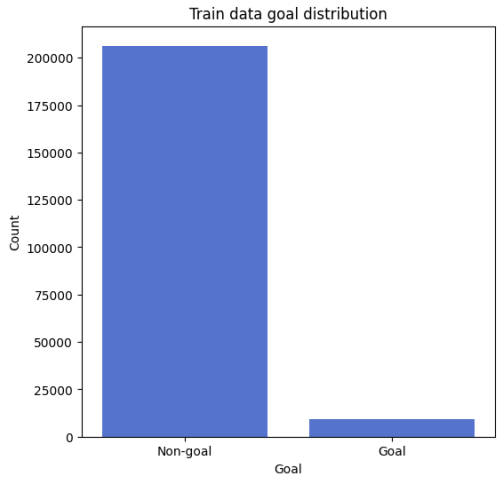


(a) Train data event count

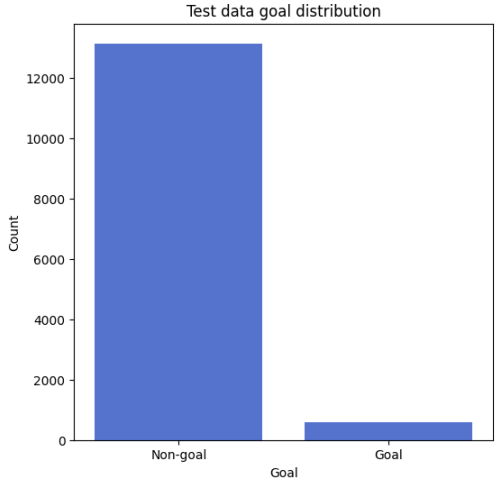


(b) Test data event count

Figure 5: In examining the distribution of the target event within the train and test datasets, we observe a predominance of "free kick" events. This imbalance could potentially introduce bias into our predictive models. However, given the limited data available, we've opted to maintain the existing distribution while ensuring a balanced ratio between the train and test sets to mitigate the effects of this imbalance.



(a) Train data goal count



(b) Test data goal count

Figure 6: The distribution of "Goal" events within the train and test data reveals a significant imbalance between goal and non-goal instances. To address this, we implemented undersampling of the majority (non-goal) label, ensuring an equal number of goal and non-goal samples for more efficient model training.

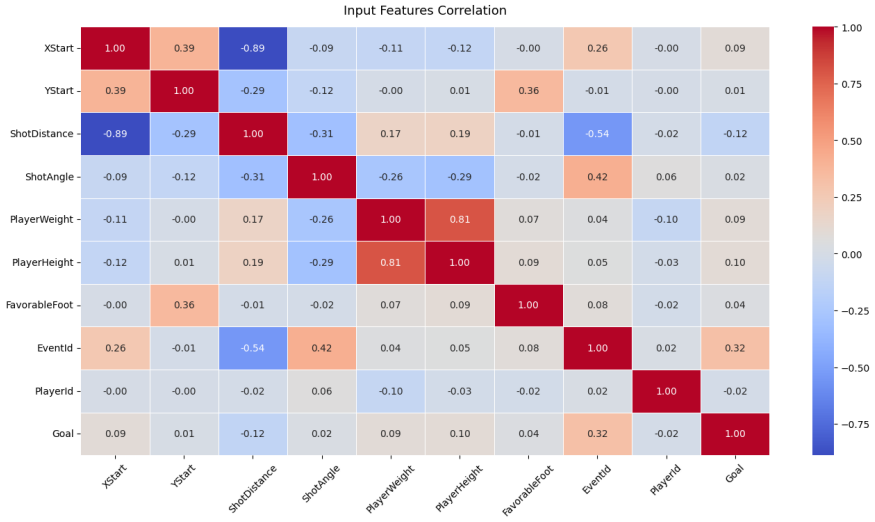


Figure 7: The correlation heatmap for Task 1 reveals that shot location (`XStart`, `YStart`) moderately influences goal likelihood, while `ShotDistance` strongly negatively impacts it. Player-specific attributes show weak or no correlation with goal outcomes, except for `PlayerId`, which exhibits a strong positive correlation, highlighting individual player skill as a key factor in shot success.

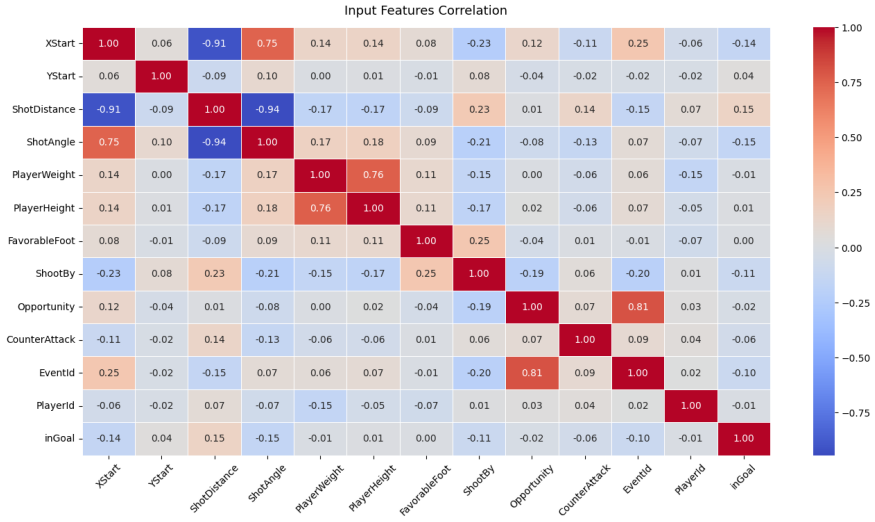


Figure 8: The correlation heatmap for Task 2 reveals interesting insights into the relationships between features and the target variable (`Goal`). Shot starting coordinates (`XStart`, `YStart`) exhibit a moderate positive correlation with each other and a weak correlation with 'Goal', suggesting that shot location influences goal likelihood but not as strongly as other factors. Conversely, `ShotDistance` displays a moderate negative correlation with 'Goal', indicating that shots taken from farther distances are somewhat less likely to result in a goal, although the relationship isn't as strong as in Task 1. Interestingly, player-specific attributes like `PlayerWeight`, `PlayerHeight`, and `FavorableFoot` show weak or no correlations with goal outcomes, similar to Task 1. The most notable positive correlations are observed between `CounterAttack` and `Goal`, suggesting that shots taken during counterattacks have a higher probability of success, and between `EventId` and `PlayerId`, indicating that certain players might be more involved in specific game events leading to scoring opportunities. Additionally, there's a noteworthy negative correlation between `PlayerId` and `EventId`, suggesting that some players might be associated with specific event types that are less likely to lead to goals.

4 Experiments

This section outlines the methodological approach that will be employed to achieve the project's objectives. The focus will be on model selection, data split, and feature engineering

4.1 Model Selection

The prediction of football goals has seen a progression in modeling techniques, evolving from classical machine learning to more complex deep learning approaches. This project will explore both paradigms, evaluating their strengths and weaknesses in the context of structured football data.

Traditional machine learning models remain a valuable tool, particularly when dealing with structured data like match statistics, team rankings, and player information. We will investigate the following models:

- **Logistic Regression [3]:** Logistic regression is a linear model that estimates the probability of an event occurring based on a set of independent variables. It employs a logistic function to map the linear combination of features to a probability value between 0 and 1, representing the likelihood of a particular outcome (win, lose, or draw). Its advantages lie in its simplicity, interpretability (coefficients can be easily understood), and computational efficiency. However, it may not capture complex non-linear relationships in the data. Despite this, its straightforwardness and often surprisingly good performance make it a valuable baseline model for our project.

- **Random Forest [2]:** Random forest is an ensemble learning method that constructs a multitude of decision trees during training and outputs the mode (for classification) or means (for regression) prediction of the individual trees. Its key strengths are robustness to overfitting, ability to handle non-linear relationships, and effective feature importance estimation. However, it can be computationally expensive for large datasets and lacks the inherent interpretability of simpler models. Given the potential for complex interactions between features in football goal prediction, the random forest's ability to model non-linearity makes it a promising candidate for our project.

- **XGBoost [4]:** In the context of football goal prediction, XGBoost's ability to handle structured data with various feature types (e.g., numerical statistics, and categorical team information) makes it a promising candidate. However, its sensitivity to hyperparameters and potential for overemphasizing certain features require careful tuning and regularization.

- **LSTM (Long Short-Term Memory) [5]:** LSTM, a variant of recurrent neural networks, is designed to address the vanishing gradient problem common in traditional RNNs, making it particularly well-suited for analyzing sequences with long-term dependencies. In football, a goal is not only an isolated event but is influenced by the flow of play, team tactics, and individual player performances over time. LSTM's ability to maintain memory of past events allows it to capture the temporal dynamics of a match, recognizing patterns that unfold over the course of the game. Furthermore, its flexibility in handling sequential input of varying lengths (different match durations) and incorporating diverse data types (e.g., player actions, team formations, game statistics) makes it a versatile model for goal prediction. However, the computational complexity of LSTMs and potential for overfitting on limited datasets pose challenges that necessitate careful model design and regularization techniques.

These machine-learning models are often favored for their interpretability. Feature importance analysis can reveal the factors most influential in predicting goal outcomes, providing valuable insights into the game.

4.2 Feature Selection

For task 1 as goal prediction. One important consideration for developing models is to pick the right set of features. After testing and engineering so far, we have chosen the following features:

- `XStart`: Starting x-coordinate of the ball.

- `YStart`: Starting y-coordinate of the ball.

- `ShotDistance`: Shot distance of the ball from `(XStart, YStart)` position.
- `ShotAngle`: Shot angle of the ball from `(XStart, YStart)` position.
- `PlayerWeight`: Weight of player (in kilogram).
- `PlayerHeight`: Height of player (in centimeters).
- `FavorableFoot`: The side with which the player likes to kick the ball (either `right` or `left`).
- `EventId`: Event when the ball movement was recorded (goal, pass, etc.)
- `PlayerId`: Identification of player.

For numerical features (coordinates, height, weight and distance, angle), we applied normalization in the form of `StandardScaler` (see Equation 2) to make feature ranges uniform, by subtracting mean μ and scaling by standard deviation σ, plus ϵ to avoid zero division.

$$x_{\text{scaled}} = \frac{x - \mu}{\sigma + \epsilon} \tag{2}$$

For Task 2, the prediction of goal position, the feature set expands upon the foundation laid in Task 1. While the core input features from Task 1 are retained, three additional features are introduced to enhance the model's ability to discern goal positions:

- `ShootBy`: This categorical feature identifies the body part used to score the goal (e.g., head, left foot, right foot).
- `Opportunity`: This binary feature indicates whether the goal was scored from an open play situation or a set piece (e.g., free kick, corner).
- `CounterAttack`: This binary feature signifies whether the goal was scored during a counterattack.

In addition to these new features, the target variable for Task 2 is `inGoal`, which represents the specific position within the goal where the ball crossed the line.

5 Evaluation

In this section, we experiment with various Machine Learning models by performing a hyperparameter search and comparing their metrics against one another.

5.1 Model Results

5.1.1 Task 1 - Goal Prediction

In evaluating the three models on the goal prediction task shown in Table 2, Random Forest, XGBoost, and LSTM consistently outperformed Logistic Regression. The XGBoost models achieved substantially higher accuracy and F1 scores, demonstrating the superior ability to discern nuanced patterns in the data and effectively handle potential non-linear relationships between features and the target variable. While Random Forest, XGBoost, and LSTM performed comparably well, the latter exhibited a slight edge in overall accuracy and F1 S1 (predicting goal events), suggesting a marginally better capability to identify positive instances.

A variety of machine learning models will be explored to identify the most suitable approach for predicting football goals. Given the classification nature of the kick (goal, not goal), the following models are proposed:

Model Name	F1S0	F1S1	Accuracy
Logistic Regression	0.838	0.841	0.839
Random Forest	0.888	0.896	0.892
XGBoost	**0.893**	**0.904**	**0.899**
LSTM	0.882	0.897	0.890

Table 2: Model evaluation for the task of predicting whether an event results in a goal involving multiple metrics. Accuracy, the overall proportion of correct predictions, was calculated across all tasks. Additionally, F1 scores were used to assess the model's performance specifically on goal prediction. F1S0 measures the harmonic mean of precision and recall for the negative class (non-goal events), while F1S1 does the same for the positive class (goal events).

5.1.2 Task 2 - Goal Position Prediction

Surprisingly, for the goal position prediction task demonstrated in Table 3, Logistic Regression outperformed both Random Forest, XGBoost, and LSTM in terms of accuracy. This result suggests that the relationship between features and goal position may be more linear than initially assumed, favoring the simpler Logistic Regression model. However, the tree-based ensemble methods still demonstrated comparable performance, particularly in predicting specific goal positions (as indicated by their F1 scores). This highlights the importance of considering both overall accuracy and class-specific performance when selecting a model for this type of multi-class classification problem.

Model Name	F1S1	F1S2	F1S3	F1S4	F1S5	F1S6	F1S7	F1S8	Accuracy
Logistic Regression	**0.341**	**0.324**	**0.133**	0.000	0.000	**0.333**	**0.300**	0.118	**0.239**
Random Forest	0.095	0.200	0.111	0.133	**0.333**	0.227	0.267	**0.316**	0.216
XGBoost	0.240	0.267	0.125	0.118	0.143	0.125	0.105	0.174	0.170
LSTM	0.091	0.200	0.105	**0.200**	0.000	0.267	0.000	0.273	0.170

Table 3: The table presents a comprehensive evaluation of model performance on the multi-class classification task of predicting goal positions (Tags 1-8). For each model—Logistic Regression, Random Forest, XGBoost and LSTM—the F1 score is reported for each individual tag, indicating the model's ability to balance precision and recall for specific goal positions. Additionally, the overall accuracy, representing the proportion of correctly classified instances across all tags, is provided for each model. This allows for a nuanced comparison of model performance, highlighting strengths and weaknesses across different goal position categories.

5.2 Training Procedure

The training process involves optimizing the hyperparameters of machine learning to achieve the best possible predictive performance.

In this study, the hyperparameters for each model were carefully selected to optimize performance. For the logistic regression model, the `newton-cg` solver was chosen for its efficiency in handling multiclass problems (`multinomial`). The random forest model utilized 1000 decision trees (`n_estimators`), ensuring a robust ensemble while maintaining computational efficiency through parallelization (`n_jobs`). The XGBoost model was configured with the `binary:logistic` objective for binary classification tasks and `multi:softprob` for multiclass problems, both employing a random seed for reproducibility. To capture temporal dependencies in the data, a Long Short-Term Memory (LSTM) neural network model was also included, utilizing the Rectified Linear Unit (ReLU) activation function (`activation='relu'`), trained for 50 epochs (`epochs=50`), and using a batch size of 32 (`batch_size=32`) for gradient updates. In all cases, the models were designed to balance predictive accuracy with computational resources, ultimately yielding insightful results.

To further improve performance, we employed grid search [1], a method systematically evaluating all possible combinations of hyperparameter values within a predefined grid. While thorough, grid search can be computationally expensive, especially with many hyperparameters or a large number of potential values.

5.3 Hardware

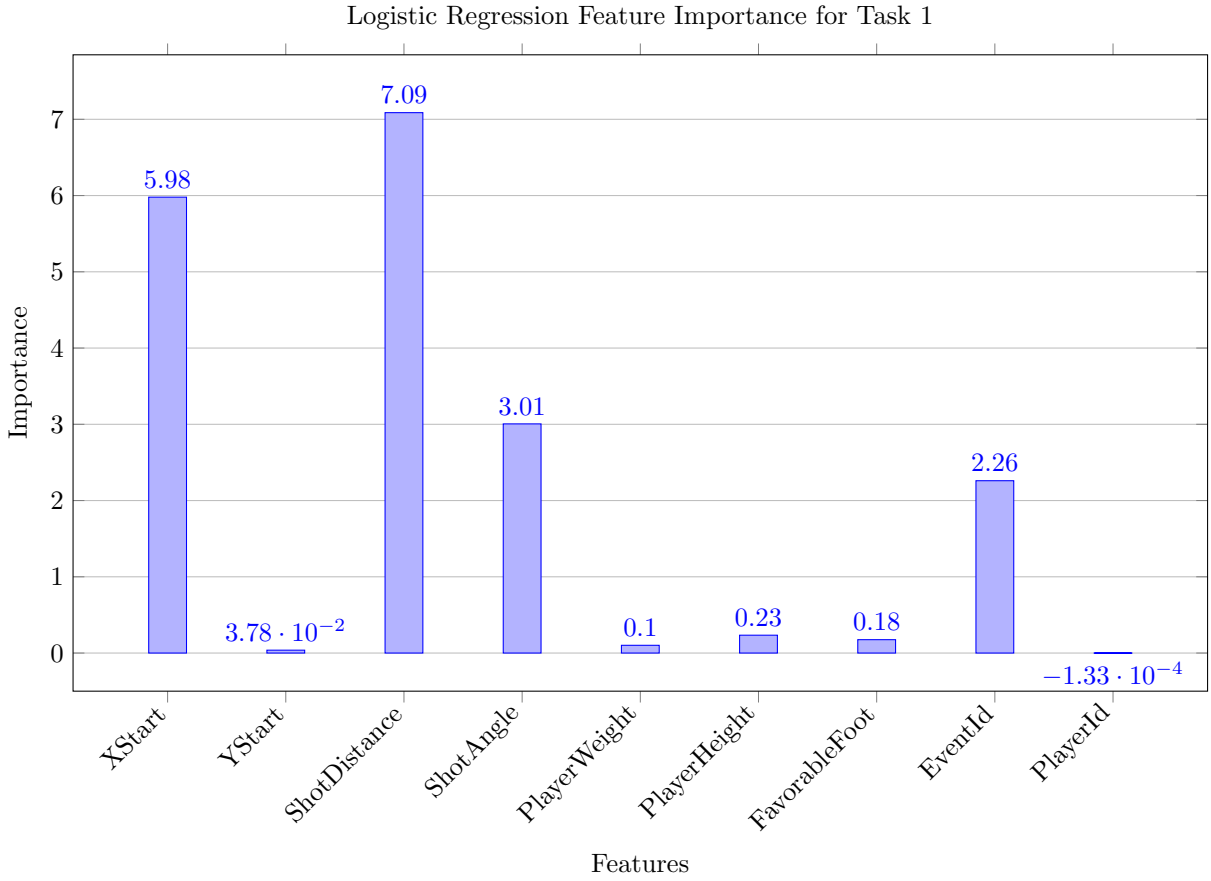
Given the diverse computational requirements of our chosen models, we will leverage a combination of hardware resources. The training of machine learning models, such as logistic regression, random forests, XGBoost, and LightGBM, will primarily utilize CPU resources due to their reliance on feature engineering and tree-based algorithms. For the deep learning models (LSTMs), which demand significant computational power for matrix operations and optimization, we will employ a GPU (P100) to accelerate training and enable efficient exploration of the hyperparameter space.

6 Results Interpretation and Visualization

Given a trained and tested model, we can explain the effects the features have on the final class output prediction.

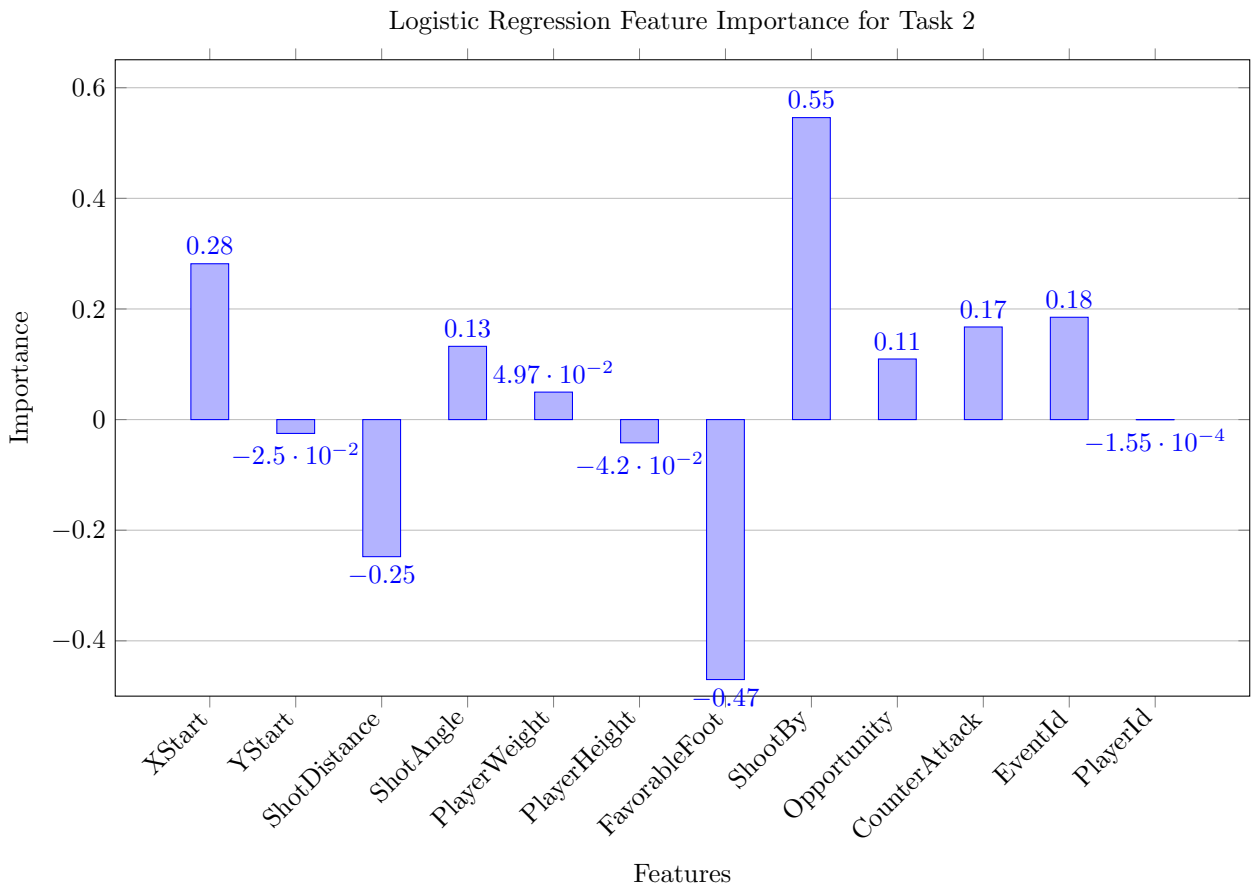
6.1 Logistic Regression

To get Logistic Regression feature importance, we simply take the weights of the coefficient w from the linear equation $y = w^T x + b$, on which the data is trained on.



From the figure visualized for task 1, it is clear that the **x-coordinate of the ball, shot distance, and shot angle** are among the most positively influential features to the prediction of the model. Meanwhile, information about the player makes little or no contribution to the model.

- Large `XStart` means the player is closer to the goal, meaning higher expected goal.
- Higher `ShotDistance, ShotAngle` means the player is closer to the actual goal

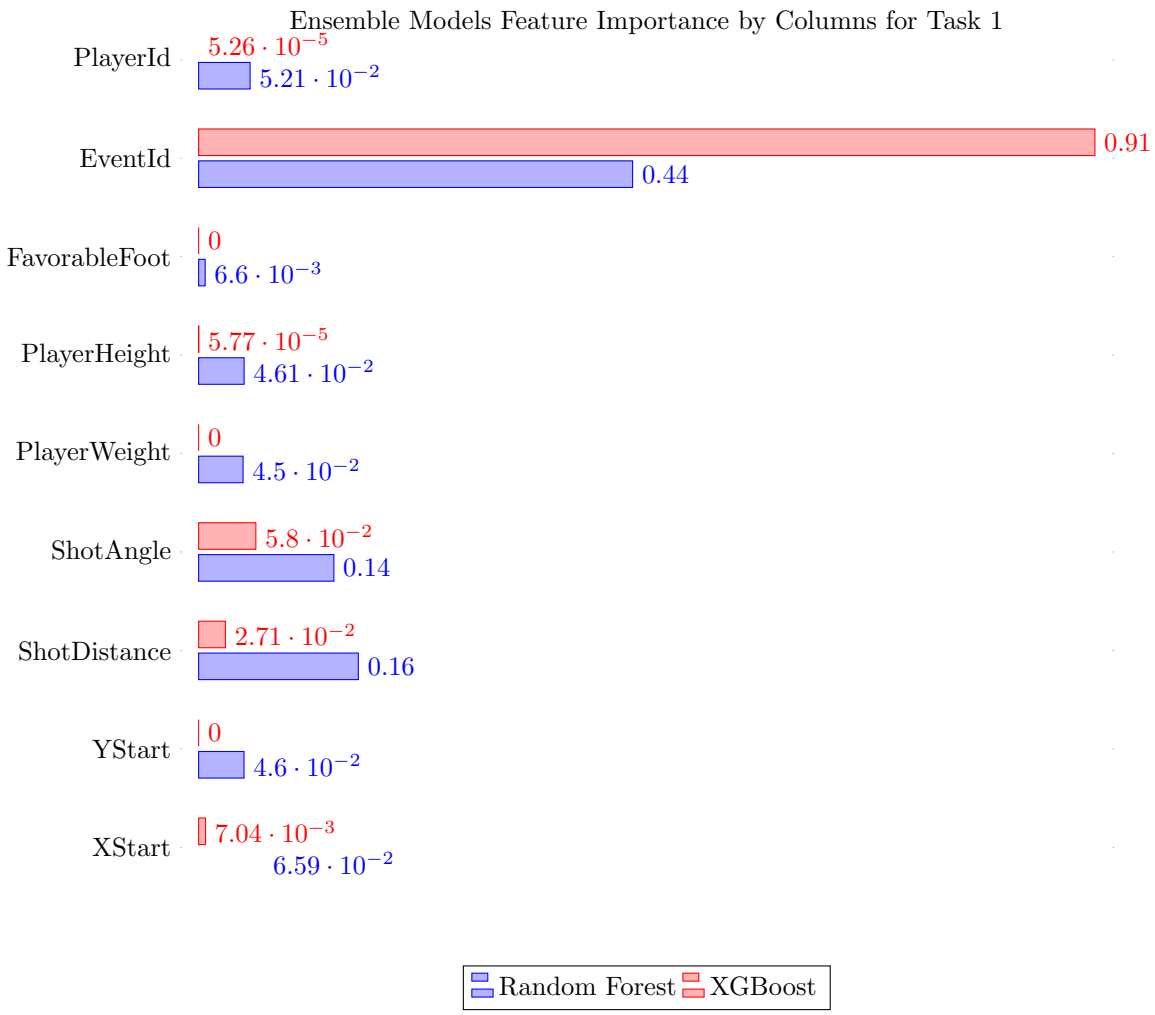


In the task 2, the feature importance analysis reveals valuable insights into the factors influencing successful shots in the model. The player taking the shot (`ShotBy`) emerges as the most critical factor, underscoring the significance of individual skill and decision-making. The initial shot coordinates (`XStart` and `YStart`) also play a key role, with `XStart` being particularly influential. Additionally, Shot Distance significantly impacts shot success, highlighting the challenge of converting long-range attempts.

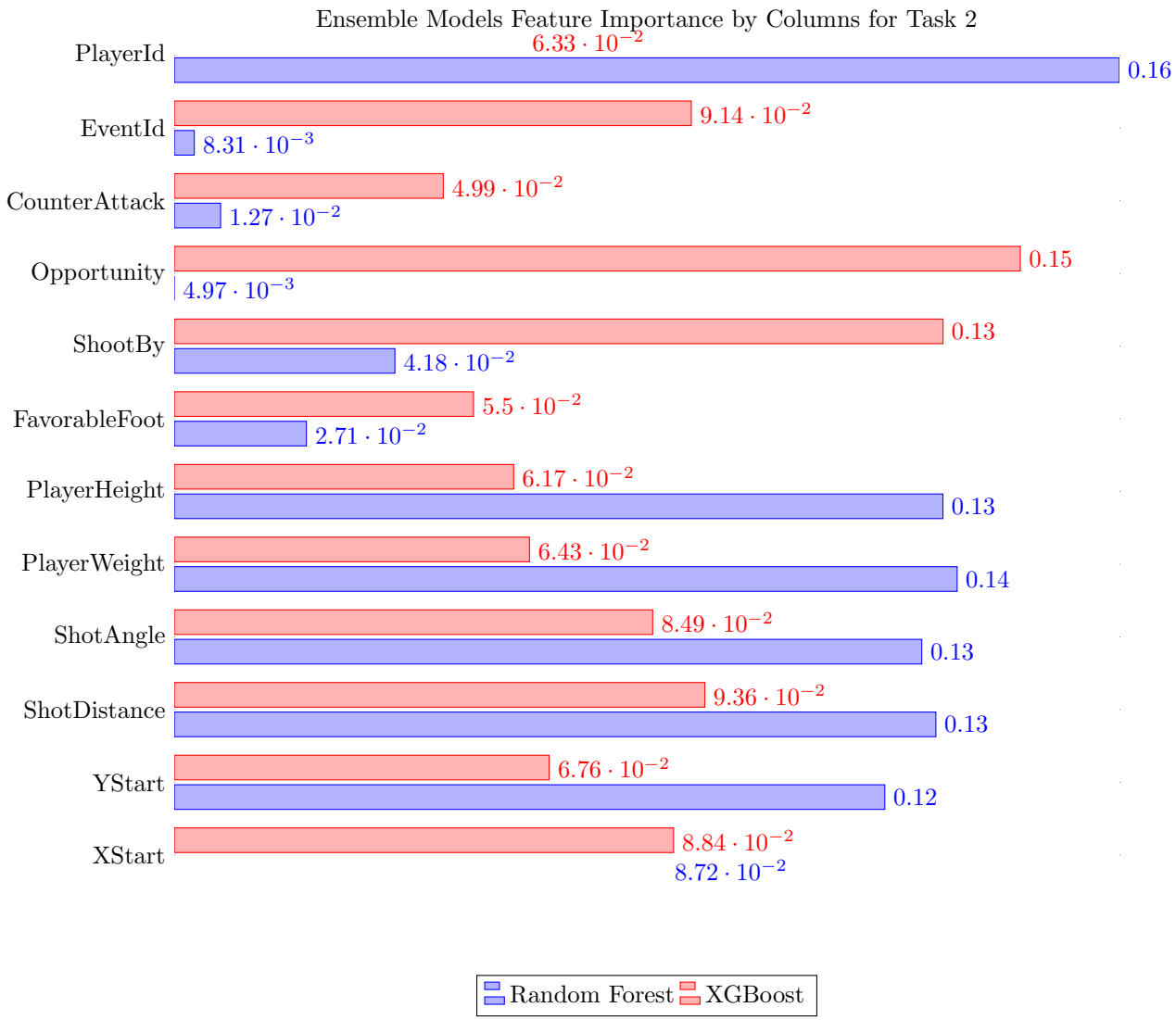
Interestingly, some features initially assumed to be important, such as player weight, height, and preferred foot, demonstrate negligible influence on the outcome. This suggests that the model has learned to prioritize other factors in its predictions. However, retaining these features might still be valuable for enhancing model variance and capturing subtle interactions that could contribute to overall performance.

6.2 Ensemble Models

For ensemble-based models (XGBoost, Random Forest), it is still easy to interpret the final results using information gain (or impurity decrease) because of the decision tree structure under the hood.



Here, the outcome of an event heavily depends on the event id itself, followed by shot distance and shot angle.



The feature importance analysis of ensemble models, incorporating both Random Forest and XGBoost, reveals intriguing insights into the factors influencing successful shots in Task 2. Notably, `PlayerId` emerges as the most dominant feature for both models, underscoring the significance of individual player skill and tendencies in determining shot outcomes.

While `EventId` displays minimal impact in the Random Forest model, it gains prominence in XGBoost, suggesting that the specific game context or match situation could be a crucial factor in this model's predictions. `XStart`, indicating the initial horizontal position of the shot, consistently ranks among the top predictors for both models, emphasizing the strategic importance of shot location on the field.

Interestingly, `Shot Distance` and `Shot Angle` exhibit a relatively high importance in both models, signifying that the physical attributes of the shot itself play a significant role in determining success. Conversely, features like `Player Height`, `Player Weight`, and `FavorableFoot` demonstrate a relatively lower impact across both models, suggesting that these player-specific attributes might not be as influential as initially assumed.

7 Conclusion and Recommendations

In conclusion, we have extracted some interesting insights into how various factors affected the possibility of a goal. We see how different events, such as a foul or a free kick, can make such a difference when making

prediction. Additionally, simpler models such as Logistic Regression are able to pump out better results than ensemble models in task 2, which comes as a surprise.

In the following reports, we will be doing more visualization such as plotting all events on a football field for more clarity. More experimentation with models is also necessary as we still have Deep Learning models such as RNN to try out in the near future.

References

[1] Daniel Mesafint Belete and Manjaiah D Huchaiah. Grid search in hyperparameter optimization of machine learning models for prediction of hiv/aids test results. *International Journal of Computers and Applications*, 44(9):875–886, 2022.

[2] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE, 1995.

[3] Michael P LaValley. Logistic regression. *Circulation*, 117(18):2395–2399, 2008.

[4] Alexey Natekin and Alois Knoll. Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, 7:21, 2013.

[5] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A review of recurrent neural networks: Lstm cells and network architectures. *Neural computation*, 31(7):1235–1270, 2019.

A Example of a sample dataset


```
{
"eventId": 8,
"subEventName": "Simple pass",
"tags": [
  {
    "id": 1801
  }
],
"playerId": 15231,
"positions": [
  {
    "y": 50,
    "x": 50
  },
  {
    "y": 48,
    "x": 50
  }
],
"matchId": 2516739,
"eventName": "Pass",
"teamId": 2446,
"matchPeriod": "1H",
"eventSec": 2.409745999999984,
"subEventId": 85,
"id": 179896442
}
```

B Events Description

Table 4: Available event and sub event IDs combinations description

Event	SubEvent	Label (event - subevent)
1	10	Duel - Air duel
1	11	Duel - Ground attacking duel
1	12	Duel - Ground defending duel
1	13	Duel - Ground loose ball duel
2	20	Foul - Foul
2	21	Foul - Hand foul
2	22	Foul - Late card foul
2	23	Foul - Out of game foul
2	24	Foul - Protest
2	25	Foul - Simulation
2	26	Foul - Time lost foul
2	27	Foul - Violent foul
3	30	Free Kick - Corner
3	31	Free Kick - Free kick
3	32	Free Kick - Free kick cross
3	33	Free Kick - Free kick shot
3	34	Free Kick - Goal kick
3	35	Free Kick - Penalty
3	36	Free Kick - Throw in
4	40	Goalkeeper leaving line
5	50	Interruption - Ball out of field
5	51	Interruption - Whistle
6	60	Offside - Offside
7	70	Others on the ball - Acceleration
7	71	Others on the ball - Clearance
7	72	Others on the ball - Touch
8	80	Pass - Cross
8	81	Pass - Hand pass
8	82	Pass - Head pass
8	83	Pass - High pass
8	84	Pass - Launch
8	85	Pass - Simple pass
8	86	Pass - Smart pass
9	90	Save attempt - Reflexes
9	91	Save attempt - Save attempt
10	100	Shot - Shot

Table 5: Available Tag IDs and Descriptions

Tag	Label	Description
101	Goal	Successful goal attempt.
102	own_goal	Goal scored for the opposing team.
301	assist	Pass leading directly to a goal.
302	keyPass	Pass that creates a goal-scoring opportunity.

Continued on next page

Tag	Label	Description
1901	counter_attack	Attack initiated immediately after regaining possession.
401	Left	Action performed with the left foot.
402	Right	Action performed with the right foot.
403	head/body	Action performed with the head or other body part.
1101	direct	Shot or pass aimed directly at the goal.
1102	indirect	Shot or pass not directly aimed at the goal.
2001	dangerous_ball_lost	Loss of possession in a dangerous area.
2101	blocked	Shot or pass that is blocked by a defender.
801	high	Pass or shot that is elevated above the ground.
802	low	Pass or shot that is kept low to the ground.
1401	interception	Successfully intercepting a pass.
1501	clearance	Clearing the ball away from danger.
201	opportunity	Clear chance to score a goal.
1301	Feint	A deceptive move to fake out an opponent.
1302	missed ball	Unsuccessful attempt to control the ball.
501	free_space_r	Free space on the right side.
502	free_space_l	Free space on the left side.
503	take_on_l	Attempt to dribble past an opponent on the left.
504	take_on_r	Attempt to dribble past an opponent on the right.
1601	sliding_tackle	Defensive tackle performed by sliding.
601	anticipated	Action performed in anticipation of an opponent's move.
602	anticipation	The ability to predict and react to an opponent's actions.
1701	red_card	Player sent off for a serious foul.
1702	yellow_card	Player cautioned for a foul.
1703	second_yellow_card	Second yellow card, resulting in a red card.
1201 - 1223	gb, gbr, gc, gl, glb, gr, gt, gtl, gtr, ...	Positions where the ball can go in relation to the goal (in/out, high/low, left/center/right).
901	through	Pass that bypasses the defensive line.
1001	fairplay	Fair play action.
701	lost	Duel, tackle, or challenge lost.
702	neutral	Neutral outcome for a duel, tackle, or challenge.
703	won	Duel, tackle, or challenge won.
1801	accurate	Action performed accurately.
1802	not accurate	Action performed inaccurately.