Gradient or Not? Predicting Football Action Sequences Using Boosting vs Neural Networks

Michał Zaręba
Adam Mickiewicz University
Faculty of Mathematics and Computer Science
Poznań, Poland *michal.zareba@amu.edu.pl*

Tomasz Piłka
Adam Mickiewicz University
Faculty of Mathematics and Computer Science
Poznań, Poland *tomasz.pilka@amu.edu.pl*

Tomasz Górecki
Adam Mickiewicz University
Faculty of Mathematics and Computer Science
Poznań, Poland *tomasz.gorecki@amu.edu.pl*

Abstract

This research compares gradient boosting methods and neural network architectures for predicting football action sequences. Using detailed event annotations and spatial-temporal positional data, we evaluate the models' ability to forecast goal-scoring opportunities several actions in advance. Through feature engineering and ensemble strategies, our results reveal key contextual and spatial factors that influence goal probabilities. Ensemble models combining CatBoost, LightGBM, and XGBoost outperform individual models, achieving an F1 Score of 0.707 and PR AUC of 0.734. These findings can provide valuable insights for real-time match analysis and player evaluation.

Keywords: Football Analytics, Action Sequence Prediction, Gradient Boosting, Neural Networks, Spatial-Temporal Analysis, Ensemble Methods.

1. Introduction

The analysis and prediction of team behavior in football have gained substantial importance due to recent advancements in data collection and machine learning techniques. Accurate forecasting of future team actions offers significant strategic value: it supports coaches in decision-making, enhances tactical planning, and provides insights that can directly influence match outcomes. The emergence of detailed event-based and positional datasets, such as those offered by StatsBomb (including their 360-degree spatial tracking data), has enabled a more comprehensive understanding of in-game tactical elements [9]. This work leverages these rich data sources to predict a team's upcoming actions by modeling sequential dependencies in match events. Specifically, we aim to forecast the subsequent nine team actions based on contextual information from the four preceding events. Our objective is to assess how such predictive models can aid teams in managing tactics proactively, improving in-game decision-making, and anticipating opponents' strategies.

We investigate state-of-the-art sequential modeling techniques to achieve this, focusing on Long Short-Term Memory (LSTM) networks and Transformer-based architectures. Furthermore, we evaluate the impact of combining event-level annotations with spatial-temporal insights derived from StatsBomb 360 data. Particular attention is paid to real-world challenges,

such as missing or incomplete data, and how to mitigate their impact in practical prediction scenarios.

2. Literature Review

Predicting football actions using data-driven approaches has garnered increasing attention over the last decade. Initial studies employed probabilistic models, such as Markov chains and Hidden Markov Models (HMMs), to predict the following action in a sequence based on historical patterns [13]. However, these methods often fell short when dealing with the dynamic and non-linear nature of football gameplay. The introduction of deep learning significantly improved the predictive performance of sequential models. Recurrent Neural Networks (RNNs) and their gated variants, LSTM [8] and GRU [5], have proven effective in capturing short- and medium-term temporal dependencies in football sequences. More recently, Transformer models [16] have emerged as a powerful alternative, utilizing self-attention mechanisms to capture long-range dependencies without the limitations of recurrence. Several studies have successfully applied LSTM and GRU architectures to model team tactics and player decision-making over time [2], [11]. Transformers further enhanced these capabilities by better modeling context over longer sequences, leading to improvements in accuracy and interpretability [16], [18]. Recent works also emphasize the importance of integrating spatial context into sequential prediction. Combining event data with positional tracking, as available through StatsBomb 360[1] datasets, leads to more accurate representations of game state and significantly improves predictive accuracy [6], [17]. Despite these advancements, few studies have explicitly compared multiple modeling approaches, particularly ensemble techniques, within a unified framework. Additionally, limited work exists that thoroughly integrates detailed spatial-temporal context with advanced feature engineering to maximize predictive accuracy. Furthermore, the literature has often overlooked the explainability and interpretability of predictive models, which is crucial for practical adoption by coaches and analysts in professional football.

In summary, existing literature highlights two key factors for effective football action prediction: advanced sequence modeling architectures and the integration of rich spatial-temporal context. However, challenges remain, particularly in handling incomplete, sparse, and noisy data, which is typical in real-world sports analytics.

3. Data Description and Preprocessing

This study utilizes two complementary datasets from Hudl StatsBomb[2], enabling a comprehensive analysis of both on-the-ball and off-the-ball actions in professional football matches. The first dataset contains *event-level data*, which includes detailed annotations of every on-the-ball action during a game. These annotations capture essential information, including the timing, outcome, and spatial location of each action. A summary of the attributes available in the event-level dataset is presented in Table 1. These features serve as the primary input for sequential models that predict the following team actions.

The second dataset consists of *360-degree camera frames*, which capture full-pitch snapshots of player positioning during selected events. This data enriches the spatial context by providing the absolute locations of all visible players on the pitch at specific moments. Table 2 summarizes the features extracted from the 360-degree frames. These contextual features are especially valuable for modeling off-the-ball behavior and defensive pressure, which are not directly observable from event data alone.

The full dataset was compiled in collaboration with the Polish football club KKS Lech

[1] `https://statsbomb.com/what-we-do/soccer-data/360-2/`, Accessed: 26-March-2025.
[2] `https://statsbomb.com/`, Accessed: 26-March-2025.

Table 1. Overview of event attributes.

Attribute	Description
Timestamp	Time of the event in match clock (minutes and seconds from kick-off)
Player and team ids	Unique IDs linking the action to specific players and teams
Event type	Type of action (e.g., pass, shot, dribble, duel)
Event outcome	Result of the event (e.g., success, failure, turnover)
Player position	(x, y) coordinates of the player at the time of the event
Ball position	(x, y) coordinates of the ball at the time of the event
Body part	Body part used for the action (e.g., left foot, right foot, head)

Table 2. Overview of attributes in the 360-degree tracking dataset.

Attribute	Description
Player positions	Absolute (x, y) coord. of all players visible in the frame
Timestamp and event linkage	Temp. alignment of the frame with the corresp. event
Distance to nearest defender	Euclidean distance to the closest opposing player
Number of defenders on goal side	Count of opp. positioned between the actor and the goal
Visible opponents	Number of visible opposing players within the frame
Visible teammates	Number of visible teammates within the frame

Poznań and encompasses all 306 matches of the 2023/2024 Polish Ekstraklasa season. The dataset comprises a total of 574,251 event records. Initially, models predicting goal-scoring within the subsequent $n \in \{3, 5, 7, 9\}$ actions were tested to assess whether extending the action sequence window would still yield robust model performance. Ultimately, after consulting with the coaching staff, who emphasized the importance of capturing extended tactical sequences, a prediction window of nine subsequent actions was selected. It is important to note that not all event records are paired with corresponding 360-degree frames due to the selective nature of spatial tracking. Consequently, positional context is available only for a subset of actions, which introduces challenges related to missing data addressed in later sections of this work.

3.1. Data Cleaning

Due to the selective availability of 360-degree frames, many features exhibit structural missingness - i.e., attributes defined only for specific event types (e.g., `shot_statsbomb_xg` applies only to shots). To manage this, columns with over 90% missing values were removed unless deemed contextually important. The remaining missing data were handled using zero imputation and accompanying binary indicators, allowing models to distinguish between true zeros and absent data. This method preserves semantic meaning, supports interpretability, and is compatible with both tree-based and neural models that require complete inputs.

3.2. Categorical Feature Handling

Gradient boosting models natively support categorical variables without preprocessing. For neural networks, we applied one-hot encoding, which—despite increasing dimensionality—is straightforward and effective for low- to medium-cardinality features. While embeddings can be advantageous for high-cardinality data, one-hot encoding proved empirically sufficient for our setup, simplifying the modeling pipeline.

3.3. Feature Engineering

An extensive feature engineering process was applied to fully leverage the dataset's potential, thereby enhancing the contextual understanding of each game situation. Several categories of characteristics were created, including the game context (e.g., current score, who is winning), the temporal context (e.g., time remaining in half), and indicators of the dynamic state. The most notable engineered features include:

- **Goal Prediction Targets.** Binary indicators signal whether the player's team scores or concedes a goal within the next 3, 5, 7, or 9 actions. These serve as forward-looking targets for sequence-based prediction.
- **Distance to Goal.** A numerical feature measuring the Euclidean distance from the event's starting location to the opponent's goal. It provides a spatial proxy for scoring likelihood.
- **Team Momentum.** Features tracking goals scored and conceded in the last 10 minutes, reflecting match dynamics and form.
- **Danger Zone Indicator.** A binary flag indicating if the event occurred in the central final quarter of the pitch, a high-risk area tied to goal probability.
- **Possession Change.** An indicator signaling whether the possession changed compared to the previous event – useful for modeling transitions and counterattacks.
- **Progressive Actions.** Features measuring ball advancement toward the opponent's goal and classifying it as progressive based on zone-specific thresholds.
- **Attacking Pressure.** A numeric feature counting the number of attacking third actions made by the same team in the next 60 seconds, capturing sustained offensive sequences.
- **Action Frequency.** Counts of total and attacking actions by the team in the last 60 seconds, measuring recent activity intensity and buildup play.
- **Critical Time Context.** Indicators for whether an event occurs in the final 5 minutes of the half, when tactical behaviors and scoring likelihood often shift significantly.
- **Team Performance Context.** Real-time scoreline tracking, including goals for and against, goal difference, and flags for winning, losing, or tied status.

Together, these features provide a rich temporal, spatial, and tactical context, enabling predictive models to capture the complexities of football match dynamics better.

3.4. Context Expansion and Feature Selection

To provide the model with richer temporal context, we extended the dataset by including features from the previous $n \in \{1, 2, 3, 4, 5\}$ events for each action. After empirical evidence, thereby optimizing model performance and reducing the past three events, which consistently yielded the best performance across our models. This approach allowed us to capture the immediate game flow and tactical buildup without introducing excessive noise or dimensionality. As a result of these transformations, the feature space expanded significantly. We applied a feature selection pipeline to retain only the 30 most informative features, thereby optimizing model performance and reducing overfitting. The selection process uses the Random Forest feature importance, which captures model-driven relevance and statistical dependency with the target. Random Forest importance reflects how useful a feature is in a non-linear decision-making process [3]. Figure 1 illustrates the ranked importance scores for the top selected features, highlighting their contribution to the model's predictive power. By eliminating redundancy, we create a compact, diverse, and informative set of features tailored for robust model training.

3.5. Feature Scaling

Although gradient boosting algorithms are inherently robust to feature scaling, neural networks are susceptible to the magnitude of input values. To ensure stable and efficient training, all numerical features were normalized using Min-Max scaling to the range of $[0, 1]$. This normalization is critical for several reasons. First, it helps prevent the dominance of features with more extensive numeric ranges, allowing the network to treat all inputs more equally. Second, it accelerates convergence during training by maintaining gradient stability, particularly when using activation functions such as ReLU or sigmoid. Lastly, it reduces the risk of exploding or vanishing gradients, common issues in deep networks when inputs vary widely in scale. By applying Min-Max scaling, we ensure that the neural network receives well-conditioned inputs, improving learning dynamics and overall model performance.

3.6. Final Dataset

The final dataset comprises 30 selected features using the feature selection pipeline described in earlier sections. These features represent a mix of spatial, temporal, tactical, and contextual information, carefully curated to maximize predictive performance while minimizing redundancy. The dataset contains 574,251 rows, each corresponding to a single in-game event. It was split into three subsets using Python's `scikit-learn`'s [15] `train_test_split()` function with the following proportions:

- **Training set (70%)**: 401,975 rows,
- **Validation set (10%)**: 57,425 rows,
- **Test set (20%)**: 114,851 rows.

The training set exhibits a significant class imbalance, with the target distribution as follows:

- **Negative class (no goal in next nine events)**: 619,859 instances (**99.23%**).
- **Positive class (goal in next nine events)**: 4,835 instances (**0.77%**).

This results in a class imbalance ratio of approximately 128:1, posing a considerable challenge for standard classification models. The target variable used is `team_scores_in_next_9_events`, a binary indicator that denotes whether the player's team scored a goal within the following nine actions following the current event. This forward-looking label trains models to predict rare but critical goal-scoring opportunities. We also experimented with multiple values for the prediction window size (i.e., the number of future events considered), such as $n = 3$, 5, and 7. Models trained with smaller windows performed slightly better in precision and F1 score, as shorter-term dependencies are more straightforward to model. However, after more profound analysis, we observed that football events can occur rapidly, and valuable actions like build-ups and counterattacks often unfold over longer sequences. We used a larger window of 9 events to better capture these dynamics, even at the cost of marginally reduced model performance. This trade-off enables the model to identify more nuanced tactical patterns that span longer play sequences, thereby aligning it more closely with the real-world context of football.

4. Predictive Models

4.1. Gradient Boosting Methods

In this research, we utilized three popular gradient boosting models: XGBoost [4], CatBoost [7], and LightGBM [10], due to their robust performance on tabular data, native handling of categorical variables, and configurable class weighting for imbalanced classification. Hyperparameter optimization was performed using the Optuna framework [1], running 100 trials for

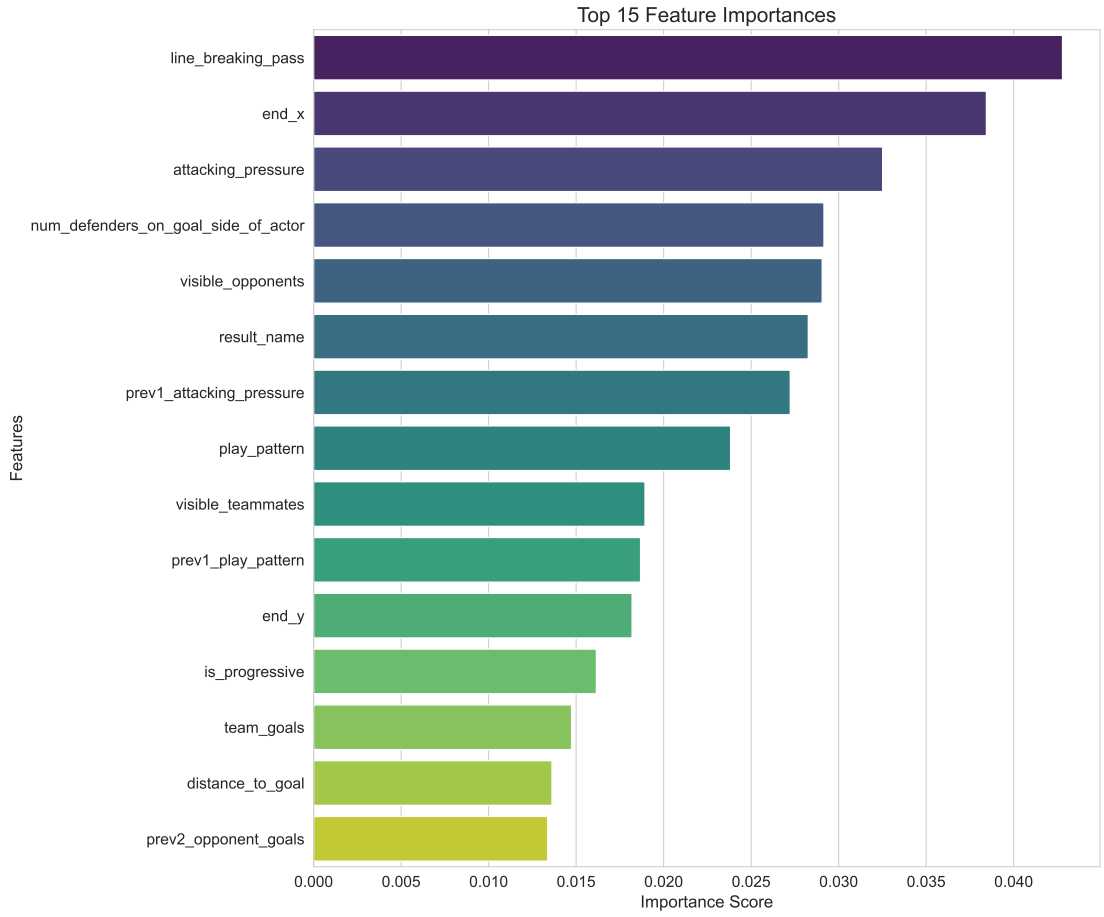

Fig. 1. Feature importance scores derived from the trained Random Forest model.

each model. Key search dimensions included learning rate (0.01–0.3, log-uniform), tree depth (ranging from 2 to 12), number of estimators (50–500), subsampling and column sampling ratios (0.5–1.0), regularization terms (reg_alpha, reg_lambda, and CatBoost's l2_leaf_reg), and additional model-specific parameters such as grow_policy (CatBoost), boosting_type (LightGBM), and gamma (XGBoost). Early stopping on validation splits was applied during tuning to prevent overfitting. Final models were trained on the full training set, and probability thresholds were optimized to maximize the F1 score. Evaluation metrics—including accuracy, precision, recall, F1, ROC AUC, and PR AUC—demonstrated the models' effectiveness in capturing rare goal-scoring events, validating their suitability for high-dimensional, imbalanced sports data.

4.2. Neural Network Architecture and Optimization Strategy

For the neural network-based approach, we implemented two architectures: a fully connected Multi-Layer Perceptron (MLP) and a convolutional neural network (CNN) adapted for tabular data. The MLP, implemented in PyTorch [14], consisted of three hidden layers with 512, 256, and 128 units, respectively, each followed by ReLU activations and dropout layers with a rate of 0.4 for regularization. The final layer consisted of a single neuron with a sigmoid activation function to output the probability of a goal-related event. This architecture totaled approximately 180,000 trainable parameters. The CNN model comprised three convolutional blocks with 1D convolutional layers having 64, 128, and 256 filter,s respectively. Each block was followed by ReLU activation, batch normalization, max pooling, and dropout layers to enhance regularization. The resulting features were flattened and passed through two dense layers with

128 and 64 units, each with batch normalization and dropout, before reaching the final sigmoid-activated output layer. To address class imbalance inherent in football event data, both models used a focal loss function [12], which adjusts the loss contribution of each example based on prediction confidence. The loss is defined as:

$$\text{FL}(p_t) = -\alpha(1 - p_t)^{\gamma} \log(p_t) \tag{1}$$

where p_t is the predicted probability for the true class. The focusing parameter γ and the scaling factor α were optimized via log-uniform search over the ranges $[0.01, 1.0]$ and $[0.01, 10.0]$, respectively. Model training used 5-fold stratified cross-validation with early stopping based on PR AUC. Hyperparameters, including learning rate, batch size, dropout rates, and focal loss parameters, were optimized using the Optuna framework across 100 trials. Final models were retrained on the full training set using the average number of epochs from early stopping and evaluated on a held-out test set using F1 Score and PR AUC. Combined with focal loss and robust tuning, these architectures offer a flexible and effective solution for predicting rare events in football.

4.3. Model Ensembling Strategy

To enhance performance and robustness, we applied ensemble strategies combining CatBoost, LightGBM, and XGBoost—a practical approach for rare-event prediction. Three methods were tested: (1) a simple average of predicted probabilities, (2) a maximum probability selection to boost recall, and (3) a weighted average (40% CatBoost, 30% LightGBM, 30% XGBoost) to balance diversity and performance. Each ensemble was evaluated on a held-out test set using metrics like F1 score, PR AUC, and recall. Thresholds were optimized via the precision-recall curve. The Simple Average ensemble achieved the best F1 and PR AUC, while the Maximum Probability variant offered the highest recall. The weighted ensemble also performed competitively, confirming that ensembling improves predictive accuracy in imbalanced football data.

5. Results

5.1. Evaluation Metrics

To evaluate model performance under severe class imbalance, we focused on F1 Score, ROC AUC, and PR AUC, which better reflect model effectiveness in rare-event settings. F1 Score balances precision and recall, while PR AUC is particularly suited for imbalanced data as it emphasizes performance on the minority class. ROC AUC captures the model's overall discriminative ability. This set of metrics ensures a robust accuracy assessment and the capacity to detect goal-scoring opportunities.

5.2. Model Evaluation and Comparison

To assess and compare the performance of the trained models, we used several standard classification metrics: Precision, Recall, F1 Score, ROC AUC, and PR AUC. These metrics are particularly well-suited for our imbalanced binary classification problem, where the minority class (goal in the following nine events) is of primary interest. Table 3 summarizes the evaluation results. The best values for each metric are highlighted in bold.

The comparison shows that ensemble methods outperform individual models across most metrics. The Simple Average Ensemble achieves the best F1 Score and PR AUC, indicating strong balanced performance in precision and recall for the rare positive class. The Maximum Probability Ensemble leads in Recall, which may be particularly useful in scenarios where false negatives are costlier than false positives. Although the CNN model achieves the highest Precision, it, along with the Dense Neural Network with Focal Loss, suffers from lower recall, which

Table 3. Comparative analysis of model evaluation metrics

Model	Precision	Recall	F1 Score	ROC AUC	PR AUC
XGBoost	0.901	0.530	0.667	0.974	0.662
LightGBM	0.811	0.557	0.661	0.977	0.685
CatBoost	0.869	0.587	0.701	0.961	0.707
Neural Network (Dense, Focal Loss)	0.929	0.438	0.595	0.940	0.534
CNN Neural Network	**0.971**	0.462	0.626	0.937	0.567
Ensemble Gradient (Simple Average)	0.885	0.589	**0.707**	0.981	**0.734**
Ensemble Gradient (Triple model weigheted)	0.878	0.591	0.707	**0.981**	0.734
Ensemble Gradient (Max Probability)	0.832	**0.609**	0.703	0.980	0.727

limits their overall F1 performance. Nevertheless, they demonstrate the value of deep learning models in extracting useful representations even in highly imbalanced tabular datasets. Traditional tree-based models, such as LightGBM and CatBoost, remain competitive and benefit significantly when combined through ensemble strategies.

6. Discussion and Future Work

Our evaluation revealed that both gradient boosting models (CatBoost, LightGBM, XGBoost) and neural networks (MLP, CNN) achieved competitive performance in predicting whether a team will score within the following nine actions. Each model type offered different trade-offs—some favoring precision, others favoring recall—highlighting the flexibility to tailor predictive systems to tactical needs. Feature importance analysis provided actionable insights into which spatial and temporal factors contribute most to goal outcomes, aligning with football intuition (e.g., presence of a danger zone, match timing). The use of focal loss and class weighting proved effective for handling severe class imbalance. Ensemble strategies, straightforward and weighted averaging, consistently outperformed individual models by combining their strengths. While CNNs lack theoretical spatial structure in tabular data, their inclusion helped probe local feature combinations heuristically. Compared to established football metrics like VAEP or xT, which evaluate isolated actions primarily based on field position, our model captures longer sequences and richer contextual data—including off-ball dynamics—enabling broader player evaluation across positions. Despite encouraging results, the study is limited to one league (Ekstraklasa). Future work should generalize the framework to other competitions and seasons. We also plan to benchmark our approach directly against xT and VAEP, incorporate anomaly detection for rare-event prediction, and revisit sequential models, such as LSTMs, with more targeted tuning. Finally, our outputs can be extended into a player evaluation metric that aggregates predicted goal impact over sequences, supporting fairer scouting across roles.

Acknowledgements

The publication and the underlying research owe their existence to the invaluable support provided by the KKS Lech Poznań club, which granted access to the datasets.

References

[1] Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M.: Optuna: A next-generation hyperparameter optimization framework. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. p. 2623–2631. KDD '19, Association for Computing Machinery, New York, NY, USA (2019)

[2] Bialkowski, A., Lucey, P., Carr, P., Yue, Y., Sridharan, S., Matthews, I.: Large-scale anal-

ysis of soccer matches using spatiotemporal tracking data. In: IEEE International Conference on Data Mining, pp. 725–730 (2014)

[3] Breiman, L.: Random forests. Machine learning 45(1), pp. 5–32 (2001)

[4] Chen, T., Guestrin, C.: XGBoost: A Scalable Tree Boosting System. In: International Conference on Knowledge Discovery and Data Mining. pp. 785–794 (2016)

[5] Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Conference on Empirical Methods in Natural Language Processing. pp. 1724–1734 (2014)

[6] Decroos, T., Bransen, L., Haaren, J.V., Davis, J.: Actions speak louder than goals: Valuing player actions in soccer. In: International Conference on Knowledge Discovery & Data Mining. pp. 1851–1861 (2019)

[7] Dorogush, A.V., Ershov, V., Gulin, A.: Catboost: gradient boosting with categorical features support (2018)

[8] Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation 9(8), pp. 1735–1780 (1997)

[9] Hudl Statsbomb: 360 Data. `https://statsbomb.com/what-we-do/soccer-data/360-2/` (2021), accessed March 20, 2025

[10] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y.: Lightgbm: a highly efficient gradient boosting decision tree. In: International Conference on Neural Information Processing Systems. p. 3149–3157 (2017)

[11] Le, H.M., Yue, Y., Carr, P., Lucey, P.: Coordinated multi-agent imitation learning. In: International Conference on Machine Learning. p. 1995–2003 (2017)

[12] Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection (2017)

[13] Ötting, M., Karlis, D.: Football tracking data: a copula-based hidden markov model for classification of tactics in football. Annals of Operations Research 325(1), pp. 167–183 (2023)

[14] Paszke, A., et al.: Pytorch: An imperative style, high-performance deep learning library (2019)

[15] Pedregosa, F., et al.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, pp. 2825–2830 (2011)

[16] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: International Conference on Neural Information Processing Systems. p. 6000–6010 (2017)

[17] Zaręba, M., Piłka, T., Górecki, T., Grzelak, B., Dyczkowski, K.: Improving the evaluation of defensive player values with advanced machine learning techniques. In: Harnessing Opportunities: Reshaping ISD in the Post-COVID-19 and Generative AI Era (ISD2024 Proceedings) (2024)

[18] Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W.: Informer: Beyond efficient transformer for long sequence time-series forecasting (2021)