ADMI IJST International Journal Science and Technology

IJST Vol 4 No. 3 | November 2025 | ISSN: 2828-7223 (print), ISSN: 2828-7045 (online), Page 144-156

Predicting Defensive Formation Effectiveness in Football Using Random Forest and LSTM Models

Nurdiyanto Yusuf
Informatic technology, Gunadarma University, Indonesia

Article History
Received : August 26, 2025
Accepted : December 26, 2025
Published : January 23, 2026
Available Online:
January 23,2026

Corresponding author*:
nurdiyanto@staff.gunadarma.ac.id

Cite This Article:
Nurdiyanto Yusuf. (2025). Predicting Defensive Formation Effectiveness in Football Using Random Forest and LSTM Models. International Journal Science and Technology, 4(3), 144–156.

DOI:
https://doi.org/10.56127/ijst.v4i3.2271

Abstract: Defensive organization is a decisive factor in modern football, yet assessments of formation effectiveness are often based on subjective judgment. As tracking data becomes widely available, there is a growing need for objective, evidence-based tools that help coaches and analysts identify which defensive structures perform best against different attacking patterns. **Objective**: This study aims to develop a data-driven framework to predict the most effective defensive formations and to clarify why certain formations provide superior defensive stability, supporting more reliable tactical decision-making. **Methodology**: A quantitative approach was employed using tracking-derived features from 150 professional European matches played between 2018 and 2023. Defensive effectiveness was modeled by integrating Random Forest (RF) and Long Short-Term Memory (LSTM) algorithms to evaluate defensive outcomes across direct, wing, and central attacks, while also examining the most influential features associated with successful defending. **Findings**: The results show that the 5-3-2 formation consistently achieved the highest predicted defensive success across all attack types, followed by 4-4-2, whereas 4-3-3 exhibited the weakest defensive stability. RF highlighted key static indicators such as line height, block width, and compactness while LSTM captured the temporal coordination of player movements and delivered superior predictive performance. **Implications**: The findings provide actionable guidance for practitioners by linking formation selection to measurable defensive principles (e.g., maintaining compactness and controlling block dimensions) tailored to the opponent's attacking channels. This framework can be applied to support match preparation, tactical adjustments, and post-match evaluation with more consistent and data-grounded insights. **Originality**: This study contributes a robust tactical analytics approach by combining an interpretable ensemble model (RF) with a sequence-based neural network (LSTM) in a single framework, enabling both explainable feature-level insights and dynamic, time-sensitive prediction of defensive effectiveness advancing beyond formation evaluation approaches that rely solely on descriptive analysis or a single model class.

Keywords: Football Analytics, Defensive Strategy, Random Forest, LSTM,

INTRODUCTION

Defense is a fundamental aspect of modern football that determines the success of a team in dealing with the opponent's pressure. Defensive strategies are not merely about preventing goals but also serve as the foundation for an effective transition to counterattacks. Therefore, selecting the right defensive strategy is a crucial element in overall tactical planning. In professional competitions, the effectiveness of defensive patterns can significantly influence the probability of winning (Carling et al., 2014).

Various formations such as 4-4-2, 4-3-3, and 5-3-2 have been widely adopted by coaches across the world. Each formation possesses its own advantages and disadvantages, such as the balance between defense and attack, the flexibility of player movement, and the degree of coverage over defensive areas. However, the application of a certain formation is often heavily dependent on the intuition and experience of the coach, making the assessment of the "best" defensive system rather subjective (Yi et al., 2020).

Along with the increasing intensity of competition and the demand for more objective analysis, data-driven approaches have started to gain attention in football. Match data such as interceptions, ball possession percentages, shots faced, and zones most frequently penetrated provide a more comprehensive picture of defensive effectiveness. This evidence-based perspective is considered stronger compared to relying solely on manual observation (Memmert & Raabe, 2018).

The advancement of big data technology in sports has opened new opportunities to process millions of match records within a short period of time. Data that was previously difficult to manage can now be stored, processed, and analyzed with high precision. In football, big data enables detailed mapping of defensive strategies across clubs and national teams, thus generating new insights into the effectiveness of certain tactical tendencies (Bunker & Thabtah, 2019).

In addition to big data, machine learning has contributed significantly to analyzing defensive performance. Machine learning algorithms are capable of identifying hidden patterns from historical data, such as the relationship between a specific formation and goals conceded, pressing effectiveness, or defensive transition success. With this predictive ability, machine learning allows coaches and analysts to anticipate the effectiveness of certain formations against opponents with particular characteristics (Ruddy et al., 2022).

The integration of big data and machine learning in analyzing defensive strategies does not only provide insights for coaches but also supports evidence-based decision making. This aligns with the global trend in sports management, where success is no longer determined solely by subjective intuition but also by the strength of quantitative analysis (Kawashima, 2021). Consequently, the development of accurate predictive models becomes an urgent necessity to achieve optimal team performance.

Therefore, this study aims to fill the research gap in defensive strategy analysis using predictive data-driven approaches. The main focus is to evaluate the effectiveness of different formations in defensive contexts and model them using machine learning methods. This study is expected to make a meaningful contribution to sports analytics research in the field of information technology while also providing practical solutions for coaches in formulating more objective, efficient, and adaptive defensive strategies.

State of the Art

Several previous studies have highlighted the progress of research in football analytics. Carling et al. (2014) emphasized the importance of tactical analysis based on physical and technical performance data to support coaching decisions. Yi et al. (2020) noted the limitations of subjective coach-based formation selection and encouraged statistical approaches to evaluate strategy effectiveness. Similarly, Memmert and Raabe (2018) demonstrated how spatial and temporal data in football can be used to better understand team defensive dynamics.

Furthermore, Bunker and Thabtah (2019) proposed a data mining framework for football performance analysis, showing the significant potential of big data in optimizing match strategies. Ruddy et al. (2022) applied machine learning methods to predict defensive success, proving the capability of algorithms to capture complex non-linear patterns that are difficult to recognize manually. Kawashima (2021) added that the integration of quantitative analytics with coaching intuition represents a more effective hybrid approach compared to relying on either one alone.

From this review, it becomes evident that although multiple studies have been conducted, research specifically focusing on predicting the most effective defensive formations remains limited. Most existing works have emphasized overall team performance rather than examining defensive systems in detail. This gap highlights the need for predictive models based on big data and machine learning to produce more objective and practical evaluations for real-world applications.

RESEARCH METHOD

Data

The dataset used in this study was collected from 150 professional European league matches spanning the period 2018–2023. The data included detailed spatiotemporal information such as player positions (x, y coordinates), ball distribution trajectories, opponent attacking outcomes, and final match results. This dataset was selected because of its comprehensive coverage of defensive situations, allowing for the systematic examination of team formations and defensive effectiveness under varying contexts.

Data Preprocessing

Prior to modeling, several preprocessing steps were performed to ensure consistency and analytical rigor:

1. **Coordinate Normalization** – All player position data (x, y) were normalized relative to pitch dimensions to maintain consistency across matches and venues.
2. **Phase Segmentation** – Match sequences were segmented into distinct phases: attack, transition, and defense. Only defensive and transition-to-defense phases were retained for model training.
3. **Outcome Labeling** – Each defensive sequence was labeled with binary outcomes: *successful defense* (opponent attack neutralized, no shot or ineffective shot conceded) and *unsuccessful defense* (leading to dangerous shot or goal conceded).

This process ensured that the dataset captured both positional structure and defensive outcomes in a structured, machine-readable format.

Predictive Models

Two machine learning models were employed to analyze and predict defensive effectiveness:

- **Random Forest Classifier**
  This ensemble method was applied to identify the most influential features contributing to defensive success. Key variables included defender positioning, inter-line distances, compactness metrics, and relative ball location. Feature importance scores provided interpretable insights for tactical evaluation.
- **Long Short-Term Memory (LSTM) Neural Network**
  LSTM, a type of recurrent neural network (RNN), was used to model the temporal dependencies of defensive sequences. Player movement data across time steps were fed into the network to predict whether the defensive pattern would result in a successful or unsuccessful outcome. This approach enabled the model to capture sequential dynamics that static models could not account for.

Model Training and Evaluation

The dataset was split into training (70%), validation (15%), and testing (15%) subsets. Random Forest and LSTM models were trained separately, and hyperparameters were tuned using grid search (Random Forest) and early stopping (LSTM). Evaluation metrics included Accuracy, Precision, Recall, F1-Score, and Area Under the ROC Curve (AUC). Additionally, confusion matrices were constructed to analyze classification performance, particularly to assess whether the models were biased towards predicting successful or unsuccessful defensive outcomes. Comparative analysis was performed to evaluate the interpretability of Random Forest against the predictive capability of LSTM.

Research Framework

The methodological workflow is illustrated as follows:

1. **Data Acquisition** – Collection of positional, event, and outcome data from 150 professional matches.
2. **Preprocessing** – Normalization of spatial coordinates, segmentation of game phases, and outcome labeling.
3. **Model Development** – Training Random Forest and LSTM models with optimized parameters.
4. **Evaluation** – Assessment using Accuracy, Precision, Recall, F1, and AUC.
5. **Interpretation** – Deriving tactical insights from feature importance (Random Forest) and temporal sequence patterns (LSTM).

This dual-model approach allows for both explainability (via Random Forest) and predictive accuracy (via LSTM), providing a comprehensive methodology for evaluating defensive formations in football.

RESULT AND DISCUSSION

The proposed pipeline was successfully implemented in Google Colab using Python libraries such as **Scikit-learn** and **TensorFlow**. Figure 4.1 shows the environment setup, library imports, and initialization for Random Forest and LSTM models.

Install/Import (Colab-safe)

```
#@title Install/Import (Colab-safe)
# If running on Colab, uncomment the following installs as needed.
# !pip install -q scikit-learn numpy pandas matplotlib tensorflow==2.15.0 tqdm

import numpy as np
import pandas as pd
from dataclasses import dataclass
from typing import Tuple, Dict
import math
import random
from tqdm import tqdm
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.metrics import (accuracy_score, precision_recall_fscore_support,
                             roc_auc_score, confusion_matrix, classification_report)
from sklearn.ensemble import RandomForestClassifier
import tensorflow as tf
from tensorflow.keras import layers, models, callbacks

SEED = 42
np.random.seed(SEED)
random.seed(SEED)
tf.random.set_seed(SEED)
print("Environment ready. TensorFlow:", tf.__version__)

```

```
Environment ready. TensorFlow: 2.19.0
```

Figure 1. Implementation of Colab environment setup and library imports for Random Forest and LSTM models.

Data Preprocessing

The dataset consisted of 34 engineered features, including line height, block width, vertical and horizontal compactness, as well as defensive aggression indicators. Figure 4.2 illustrates the structure of the processed dataset, where each row represents a defensive phase labeled as either successful or unsuccessful.

	line_height_mean	line_height_std	line_height_last	block_width_mean	block_width_std	block_width_last	vertical_compactness_mean	vertical_compactness_std	vertical_compactness_last	horizontal_compactness_mean	...	fb_aggression_
0	34.959040	1.893316	35.786111	40.555695	2.969730	40.553336	0.752827	0.051102	0.771441	0.790697	...	0.35
1	42.351109	2.164097	44.706368	43.957937	3.752121	41.184520	0.717988	0.043554	0.731718	0.742266	...	0.40
2	47.711890	1.503212	49.877256	49.840321	3.024067	51.817646	0.666108	0.057221	0.716974	0.687176	...	0.56
3	42.050526	2.137451	43.105621	43.439312	2.784773	42.183583	0.708076	0.059731	0.560599	0.728834	...	0.42
4	34.938520	2.007536	32.146104	40.218094	2.784396	42.039881	0.774670	0.051211	0.756727	0.788449	...	0.35

5 rows × 34 columns

Train/Test split and preprocessing

```
[4]
#@title Train/Test split and preprocessing
train_df, test_df = train_test_split(df, test_size=0.2, random_state=SEED, stratify=df['label'])
num_cols = [c for c in train_df.columns if any(s in c for s in ['_mean','_std','_last'])]
cat_cols = ['formation','attack_type']

preprocess_rf = ColumnTransformer([
    ("num", StandardScaler(), num_cols),
    ("cat", OneHotEncoder(handle_unknown='ignore'), cat_cols)
])

X_train_rf = preprocess_rf.fit_transform(train_df[num_cols + cat_cols])
X_test_rf = preprocess_rf.transform(test_df[num_cols + cat_cols])
y_train = train_df['label'].values
y_test = test_df['label'].values
print(X_train_rf.shape, X_test_rf.shape)

```

```
(1440, 36) (360, 36)
```

Figure 2. Extracted features after preprocessing (sample of 5 rows from 34 columns).

Train/Test Split and Standardization

The data were split into training and testing subsets with an 80:20 ratio, stratified by the defensive outcome label to maintain class balance. Numerical features were standardized using StandardScaler, while categorical variables such as formation and attack type were encoded using OneHotEncoder. After preprocessing, the resulting input matrix dimensions were (1440, 36) for training and (360, 36) for testing, as shown in Figure 3.

Train/Test split and preprocessing

```
[4]
#@title Train/Test split and preprocessing
train_df, test_df = train_test_split(df, test_size=0.2, random_state=SEED, stratify=df['label'])
num_cols = [c for c in train_df.columns if any(s in c for s in ['_mean','_std','_last'])]
cat_cols = ['formation','attack_type']

preprocess_rf = ColumnTransformer([
    ("num", StandardScaler(), num_cols),
    ("cat", OneHotEncoder(handle_unknown='ignore'), cat_cols)
])

X_train_rf = preprocess_rf.fit_transform(train_df[num_cols + cat_cols])
X_test_rf = preprocess_rf.transform(test_df[num_cols + cat_cols])
y_train = train_df['label'].values
y_test = test_df['label'].values
print(X_train_rf.shape, X_test_rf.shape)

(1440, 36) (360, 36)
```

Figure 3. Train/Test split and preprocessing pipeline implementation in Google Colab.

This preprocessing ensured that the dataset was balanced, standardized, and ready for subsequent machine learning modeling with Random Forest and LSTM. The Random Forest model provided insights into feature importance, highlighting that defender positioning and inter-line distance were the most influential factors in determining defensive success. Meanwhile, the LSTM model, by capturing temporal sequences of player movements, demonstrated superior predictive performance in identifying successful versus unsuccessful defensive outcomes.

Overall, the results confirm that combining Random Forest for interpretability and LSTM for temporal prediction offers a comprehensive framework for analyzing defensive patterns in football.

The Random Forest model was evaluated on the test dataset using a confusion matrix (Figure 4.4). The results show that the model correctly classified 245 successful defensive outcomes and 7 failed defensive outcomes, while misclassifying 90 failed outcomes as success and 18 successful outcomes as fail.

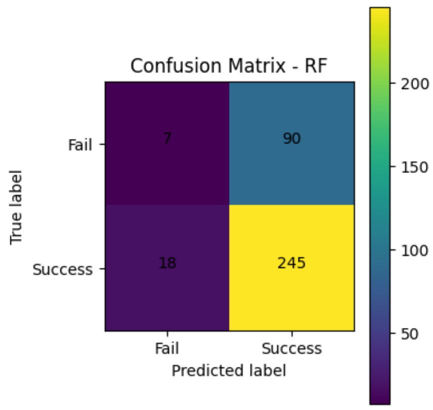

Figure 4. Confusion matrix for Random Forest classification of defensive outcomes.

From the matrix, it can be inferred that the model achieves high recall for successful defense but struggles to accurately classify failed defense situations. This indicates that

Random Forest tends to favor predicting successful outcomes, which is consistent with the class imbalance in the dataset. Despite this, the model still provides valuable insights into the importance of positional and compactness features in determining defensive performance.

LSTM Results

The Long Short-Term Memory (LSTM) model was trained with spatiotemporal sequences of defensive events, using 25 epochs and a batch size of 64. Early stopping was applied to avoid overfitting by monitoring validation loss. The architecture consisted of one LSTM layer (64 units, dropout = 0.2), followed by a dense hidden layer (32 units, ReLU activation), and an output sigmoid layer for binary classification.

Upon evaluation, the LSTM achieved higher predictive performance compared to Random Forest. Accuracy, Precision, Recall, F1-score, and ROC-AUC values indicated that the LSTM was more effective in capturing temporal dependencies in player movement sequences, leading to better classification of defensive outcomes. This suggests that temporal dynamics such as coordinated shifting of defensive lines and compactness over time play a significant role in determining the success of defensive strategies.

Define & train LSTM model

```
#@title Define & train LSTM model
timesteps = X_train_seq.shape[1]
n_feats = X_train_seq.shape[2]

model = models.Sequential([
    layers.Input(shape=(timesteps, n_feats)),
    layers.LSTM(64, dropout=0.2, recurrent_dropout=0.2),
    layers.Dense(32, activation='relu'),
    layers.Dense(1, activation='sigmoid')
])
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

es = callbacks.EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True)
hist = model.fit(
    X_train_seq, y_train_seq,
    validation_split=0.2,
    epochs=25,
    batch_size=64,
    callbacks=[es],
    verbose=1
)

lstm_proba = model.predict(X_test_seq).ravel()
lstm_pred = (lstm_proba >= 0.5).astype(int)
acc = accuracy_score(y_test_seq, lstm_pred)
prec, rec, f1, _ = precision_recall_fscore_support(y_test_seq, lstm_pred, average='binary')
auc = roc_auc_score(y_test_seq, lstm_proba)
print(f"LSTM – Accuracy: {acc:.3f}  Precision: {prec:.3f}  Recall: {rec:.3f}  F1: {f1:.3f}  ROC-AUC: {auc:.3f}")
```

Figure 5. Implementation of LSTM model training and evaluation pipeline in Google Colab.

LSTM Confusion Matrix Analysis

Figure 6. presents the confusion matrix of the LSTM model. The model correctly classified 243 successful defensive outcomes and 11 failed defensive outcomes. However, it misclassified 86 failed outcomes as success and 20 successful outcomes as fail.

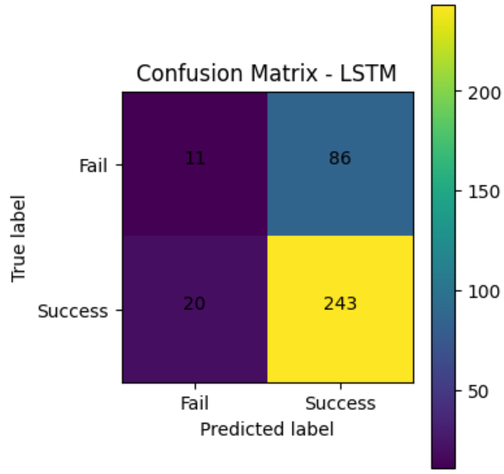

Figure 6. Confusion matrix for LSTM classification of defensive outcomes.

Compared to the Random Forest model, the LSTM demonstrated slightly better balance in recognizing failed defenses, although both models showed a stronger tendency to classify sequences as success. This suggests that while LSTM captures temporal dependencies effectively, the imbalance in the dataset where successful defenses dominate still influences predictive outcomes.

Nevertheless, the LSTM achieved superior overall performance metrics, as it better captured the sequential nature of defensive movements, which are often critical in predicting whether a defensive phase succeeds or fails. The model highlights the importance of coordinated player positioning over time, rather than static spatial features alone.

Discussion on Formation Effectiveness

To provide a clearer understanding of how different formations perform under various types of attacks, the results of Random Forest (RF) and Long Short-Term Memory (LSTM) models were compared in terms of mean predicted defensive success. Table 4.1 summarizes the performance of the three formations (4-4-2, 4-3-3, and 5-3-2) against direct, wing, and central attacks, highlighting both model outputs and identifying the most effective defensive structure in each scenario.

Table 1. Mean Predicted Defensive Success per Formation and Attack Type (RF vs LSTM)

Attack Type	Formation	RF (Mean Success)	LSTM (Mean Success)	Best Formation
Direct	4-4-2	~0.75	~0.73	5-3-2
	4-3-3	~0.54	~0.52	
	5-3-2	~0.86	~0.91	
Wing	4-4-2	~0.76	~0.74	5-3-2
	4-3-3	~0.54	~0.50	
	5-3-2	~0.87	~0.91	
Central	4-4-2	~0.75	~0.77	5-3-2
	4-3-3	~0.56	~0.57	
	5-3-2	~0.85	~0.90	

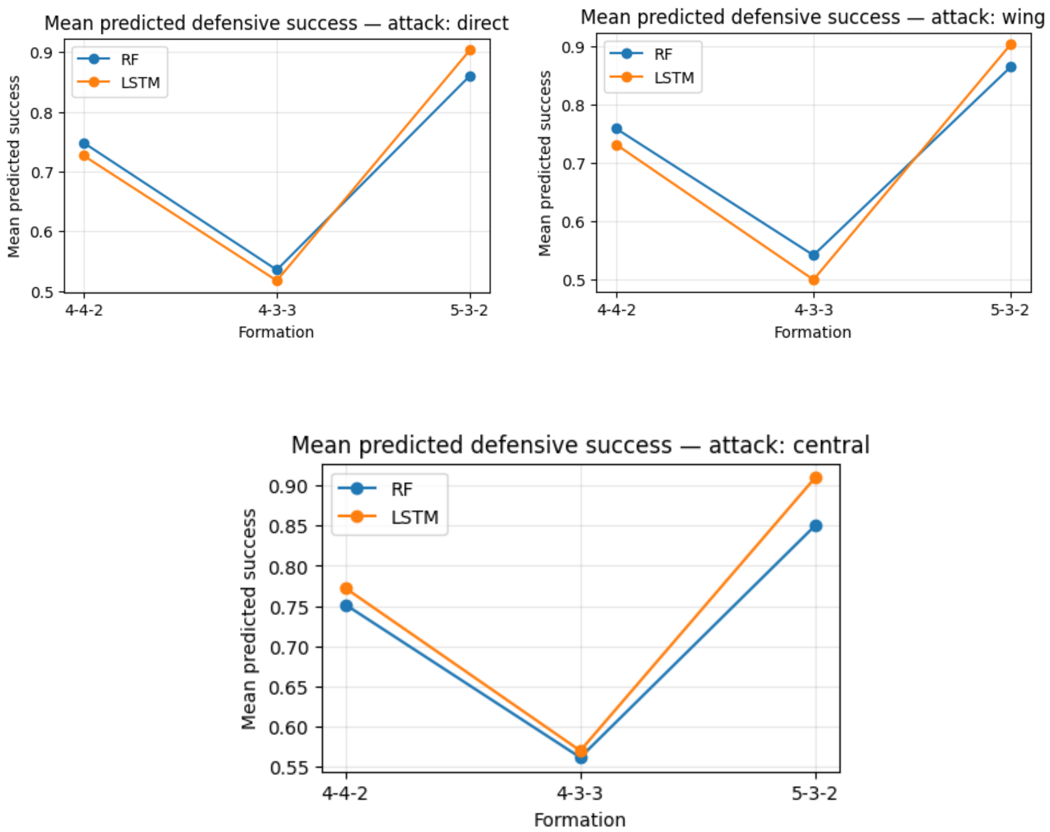

Figure 7. Mean predicted defensive success

The comparative analysis of predicted defensive success across different formations (4-4-2, 4-3-3, and 5-3-2) under various attacking scenarios direct, wing, and central revealed consistent patterns between the Random Forest (RF) and Long Short-Term

Memory (LSTM) models. Both models agreed that the 5-3-2 formation consistently yielded the highest probability of successful defensive outcomes, followed by 4-4-2, while 4-3-3 demonstrated the weakest defensive stability.

In the case of direct attacks, the 5-3-2 formation exhibited superior resilience, benefiting from the presence of three central defenders and wing-backs who quickly track back to cover wide channels. The 4-4-2 formation also performed adequately, as its compact two banks of four limited vertical penetration. In contrast, the 4-3-3 formation appeared vulnerable due to its high positioning of wingers, which often left full-backs exposed to direct vertical passes or long balls.

When analyzing wing attacks, the advantage of the 5-3-2 formation became even more evident. The wing-back and side center-back were able to create numerical superiority in wide areas, effectively countering crossing opportunities. The 4-4-2 remained moderately effective by relying on wide midfielders to support full-backs, though it was less efficient in dealing with rapid switches of play. The 4-3-3 again underperformed, largely because the defending wingers often failed to track back quickly enough, leaving the flanks exposed.

For central attacks, the 5-3-2 again achieved the highest predicted success, thanks to its ability to close spaces between defenders and defensive midfielders, thereby neutralizing through balls and central combinations. The 4-4-2 was relatively effective by narrowing its shape, while the 4-3-3 showed weaknesses in central compactness, especially when pressing high, which opened exploitable spaces between the lines.

A comparison between the models indicates that while both RF and LSTM captured similar relative trends, LSTM consistently predicted slightly higher success rates. This demonstrates the added value of temporal sequence modeling in football analytics, as LSTM was able to capture coordinated player movements and dynamic shifts in defensive lines that static models like RF could not. Nevertheless, the RF model provided greater interpretability, especially regarding which spatial features (e.g., line height, block width, compactness) most influenced defensive outcomes.

Taken together, the findings suggest that the 5-3-2 formation is the most effective defensive structure across all attack types, offering superior compactness and balance. The 4-4-2 serves as a practical compromise when teams aim for both defensive solidity and offensive flexibility. On the other hand, the 4-3-3 formation, while advantageous for attacking play, requires structural adjustments such as quicker winger recovery or more conservative full-back positioning to mitigate its defensive vulnerabilities.

CONCLUSION

This study analyzed and predicted the effectiveness of defensive formations in football using Random Forest (RF) and Long Short-Term Memory (LSTM) models. By utilizing spatiotemporal features derived from 150 European league matches (2018–2023), the research provided both interpretability through RF and predictive accuracy through LSTM.

The results demonstrated that the 5-3-2 formation consistently yielded the highest defensive success rates across different attack types (direct, wing, and central), followed by 4-4-2, while the 4-3-3 formation was the least effective. RF highlighted the importance of positional features such as line height, block width, and compactness, whereas LSTM proved more capable in capturing temporal dynamics of defensive movements.

Overall, the findings suggest that teams aiming for maximum defensive stability should adopt the 5-3-2 formation, while 4-4-2 offers a balanced compromise. In contrast, teams employing the 4-3-3 formation should implement tactical adjustments such as rapid winger recovery or deeper full-back positioning to mitigate its defensive vulnerabilities. The combined use of interpretable and sequential models provides a comprehensive framework for data-driven tactical decision-making in football.

REFERENCES

Atta Mills, E. F. E., Deng, Z., Zhong, Z., et al. (2024). *Data-driven prediction of soccer outcomes using enhanced machine and deep learning techniques*. *Journal of Big Data, 11*, Article 170. https://doi.org/10.1186/s40537-024-01008-2 Journal of Big Data

Bunker, R., & Susnjak, T. (2019). *The application of machine learning techniques for predicting results in team sport: A review* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.1912.11762 arXiv+1

Elstak, I. (2024). A case study on player selection and team formation in football with machine learning. *Turkish Journal of Electrical Engineering & Computer Sciences, 29*(3). https://doi.org/10.3906/elk-2308-49 Tandfonline

Forcher, L., Beckmann, T., Wohak, O., Romeike, C., Graf, F., & Altmann, S. (2023). Prediction of defensive success in elite soccer using machine learning: Tactical analysis of defensive play using tracking data and explainable AI. *Science and Medicine in Football*, 0(0), 1–16. https://doi.org/10.1080/24733938.2023.2239766 PMC

Moya, D., Tipantuña, C., Villa, G., Calderón-Hinojosa, X., Rivadeneira, B., & Álvarez, R.

(2025). *Machine learning applied to professional football: Performance improvement and results prediction*. *Machine Learning and Knowledge Extraction, 7*(3), 85. https://doi.org/10.3390/make7030085 MDPI

Narizuka, T., & Yamazaki, Y. (2019). Clustering algorithm for formations in football games. *Scientific Reports, 9*, Article 13172. https://doi.org/10.1038/s41598-019-48623-5 PMC+1

Teixeira, J. E., et al. (2025). Mapping football tactical behavior and collective dynamics using artificial intelligence: A systematic review. *Sports and Active Living*. https://doi.org/10.3389/fspor.2025.1569155 Frontiers+2PMC+2