Intelligent Analysis of Sports Data in the Tasks of Forming Effective Sports Teams

Oleh Zaritskyi[1,†], Danyil Pylypovych[1,*,†] and Ihor Miroshnychenko[1,†]

[1]*Taras Shevchenko National University of Kyiv, 64/13, Volodymyrska Street, Kyiv, 01601*

Abstract

The article discusses topical issues of applying data mining methods in the tasks of forming effective sports teams. The methods of decision trees, linear regression, and the SVM model, as well as the ensemble method for analyzing the correspondence of player characteristics to their position, which allows for identifying the key factors that determine the best place for a player on the field, are considered. The possibilities of artificial intelligence in team coaches' decision-making tasks to build a positional strategy are analyzed, which will significantly increase the team's overall efficiency and inform its composition for a specific model of the opponent's game. The article provides recommendations for processing sports data arrays and establishing suitable models to optimize their efficiency and productivity. A comparative analysis of the real and predicted ratings of specific players has confirmed the high accuracy of the developed models, with MAE and MRSE estimates within 1-3% of the actual values. The authors have proposed for the first time a scientific approach specifically to the tasks of forming effective team compositions, taking into account statistical data, and not only predicting game results, as in most existing studies. The authors also provide practical recommendations for tuning models in terms of their effectiveness, specifically in the tasks of analyzing sports data sets, which can be used as a starting point in research on statistical data of team sports. The analysis of the developed models, identification of the best model for such tasks, and interpretation of the results obtained can become the basis for building automated analysis and forecasting systems in sports teams for the respective analytical departments of the teams. The authors also envisage additional research on the correlation between predicted effective teams and match results, which could provide additional indirect evidence of the effectiveness of the proposed forecasting methods.

Keywords

Machine learning, sports teams, decision trees, artificial intelligence, game strategy

1. Introduction

Correct positioning of players is a fundamental basis for success in team sports. It allows for the maximization of the individual strengths of each athlete, creating a balanced system where everyone plays a clearly defined role. In football, for example, positioning determines the tactical scheme of the game, and in basketball, it allows you to effectively distribute tasks between players of different builds and skills. Incorrect placement of athletes can lead to an imbalance in the team, ineffective use of individual players' talents, and, as a result, a deterioration in overall results. High-level coaches pay special attention to positional strategy, often adapting it to a specific opponent to achieve maximum efficiency.

As an example, the authors consider football teams, due to the maximum variability of positions in this sport. In football, each position on the field requires a unique set of characteristics that determine the effectiveness of the player [1]. For example, attackers must have high speed and the ability to complete attacks, while midfielders need good passing technique and field vision. Defenders, in turn, must have high indicators of selection and physical strength, and goalkeepers must have quick reactions and hand skills.

Correct positioning of players is critical for successful team play. Incorrect use of a player can lead to a decrease in his effectiveness and negatively affect the team's results. Traditionally, coaches make decisions about player positions based on their own experience and observations [2]. However,

Information Technology and Implementation (IT&I-2025), November 20-21, 2025, Kyiv, Ukraine

[*]Corresponding author.

[†] These authors contributed equally.

oleh.zaritskyi@knu.ua (O. Zaritskyi); lokdi18@knu.ua (D. Pylypovych); ihor.miroshnychenko@knu.ua (I. Miroshnychenko)

0000-0002-6116-4426 (O. Zaritskyi); 0009-0009-7424-2065 (D. Pylypovych); 0000-0002-1307-7889 (I. Miroshnychenko)

modern data analysis methods allow automating this process and making it more objective, using mathematical models and machine learning algorithms.

Using classification algorithms for decision-making, such as decision trees, can help determine which player characteristics are most important for each position. This allows not only to improve the distribution of players, but also to identify potential changes in their roles on the field and to build a team for a specific opponent's playing style [3].

Machine learning opens up new possibilities for analyzing sports data, allowing you to find patterns that are difficult to detect using traditional methods. One approach is to use classification and regression algorithms [4] to help predict which position a player will perform best. Machine learning algorithms can process large amounts of data, including player statistics, physical characteristics, technical skills, and even performance history, and use this data to build models that automatically determine the suitability of a player for a particular position [5, 6].

Decision trees are one of the most effective methods for this task, as they provide a clear interpretation of the decisions made. Decision trees can reveal which characteristics are most important for each position, and how they have changed over the years. This not only helps clubs in choosing tactics and building their squads, but also individual players in understanding their own strengths and weaknesses.

Decision trees are one of the most popular methods in machine learning due to their simplicity, interpretability, and efficiency in handling large data sets with a relatively small number of variables. For football position analysis, this method is an ideal choice for several reasons [7, 8, 9]:

1. Ease of interpretation – the decision tree builds a hierarchical structure of decisions, which allows you to clearly understand which characteristics of the player are most significant for determining his position.
2. Flexibility in working with different types of data – the method works well with numerical and categorical data, which allows you to use both physical indicators and the player's rating.
3. Determination of the most important characteristics – the decision tree model automatically selects the most relevant variables, which helps to understand which factors most affect the distribution of players.
4. Robustness to noise in the data – the method works well even in the presence of some incorrect or missing values.
5. Visualization – the results of the model could be easily presented in the form of a tree, which makes it accessible for understanding even by non-specialists in the field of machine learning and, accordingly, the administrations of the teams making decisions.

Linear regression is a simple model for predicting a numerical value (regression) that tries to find the best line describing the relationship between independent variables (features) and the target variable. It is used to predict a quantitative (continuous) target variable based on one or more independent variables. The model has the following form [10]:

$$\hat{y} = w_0 + w_1x_1 + w_2x_2 + \cdots + w_nx_n$$

where: $\hat{y}$ is the predicted value, w is the free term (intercept), w_1, w_2, ..., w_n are the coefficients (weights) of the model.

The model is trained by minimizing the mean square error (MSE) between the predictions and the actual values:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

The main advantages of the model are simplicity and speed of training, easy interpretation - each weight shows the influence of a feature, and it works well with linear dependencies. However, the model also has disadvantages, such as sensitivity to multicollinearity and outliers, and is not suitable for complex (non-linear) dependencies.

SVM is a powerful algorithm for classification and regression tasks that searches for a hyperplane (or boundary) that best separates classes with a maximum margin. For nonlinear problems, kernels are used to transform the feature space into a higher dimension. It is used for classification (mostly) and regression, especially when the boundaries between classes are complex [11].

SVM searches for the optimal hyperplane that separates classes as much as possible. In the case of a nonlinear boundary, a kernel trick (e.g., RBF kernel) is used to transform the input features into a higher dimension. For classification tasks, an optimization problem with constraints is solved:

$$\min_{w,b}\frac{1}{2}\|w\|^2 \ under\ the\ condition\ y_i(w\cdot x_i+b)\geq 1$$

The main advantages are efficiency with a high number of features, the ability to work with nonlinear dependencies, and resistance to overfitting (especially with the right choice of regularization). However, the model has its drawbacks in the form of slowness with large datasets, the need for fine-tuning of parameters (e.g., C, gamma), and is less interpretable, especially when using kernels.

Ensemble methods combine the predictions of several weaker models (e.g., decision trees) to create a more powerful model. Designed to improve accuracy by combining multiple models (base models or “weak learners”). Types of ensembles:

- Bagging (Bootstrap Aggregating), e.g., Random Forest - trains many trees on different subsamples of data and combines the results (voting or average).
- Boosting, e.g., Gradient Boosting, XGBoost – trains models sequentially, each subsequent one focuses on the errors of the previous one.
- Voting/Stacking - combining predictions of several different models (SVM, decision tree, logistic regression, etc.)

The advantages of such models are high accuracy, the ability to work well with large amounts of data and complex dependencies, and they are less prone to overfitting than individual trees. However, such models have the disadvantages of being difficult to interpret, slower to learn (especially Boosting), and can be resource-intensive [12].

This study uses decision trees, linear regression, support vector machine, and an ensemble voting type method to analyze the correspondence between player characteristics and position, which allows for the identification of the key factors that determine the best position of a player on the field.

A review of previous studies shows a wide range of applications of machine learning in sports analytics. One study [13] focused on predicting sports injuries using deep learning, which used the permutation entropy method to detect hidden patterns in athletes' physiological data. The achieved accuracy of 92% demonstrates significant potential for improving sports medicine.

Another study [14] analyzes the ability to predict football match results using machine learning algorithms such as LightGBM and AdaBoost. It found that the overall prediction accuracy is about 52.8%, and predicting draws remains a particularly challenging task.

Authors of the study [15] focus on automated player performance assessment, which uses classification methods to identify key characteristics of football players. It was found that the algorithms can partially imitate expert assessments, but have limited accuracy in predicting the outcome of a match at 63.4%.

Paper [16] is devoted to machine learning approaches in sports analysis, in particular, their strengths and weaknesses in the analysis of sports events. It is determined that one of the key factors in improving the accuracy of forecasts is the availability of open data.

In the field of physical education, the use of XGBoost in combination with reinforcement learning to personalize training has been studied [17]. A standardized data collection system has been introduced, which allows increasing the efficiency of the educational process by 46.42%.

As the analysis of research in the field of sports analytics has shown, most of the works are devoted to the prediction of match results. Despite the significant contribution of previous research in the field of sports analytics, none of them focus directly on determining the key characteristics of football players depending on their position on the field. The study of the authors of this article puts this aspect in the spotlight, examining the characteristics of football players over several years (FIFA15–21). This approach allows us to trace trends in the requirements for different positions.

To analyze the importance of characteristics, the above methods are used to help identify the factors that most strongly influence the rating of players at each position. The authors first considered and proposed an approach to automatically determine the optimal positions of players based on their attributes, which can be useful for coaches and analysts when forming team compositions and developing game strategies.

Thus, the authors' research further developed traditional sports data analysis, which was expanded with modern machine learning methods and offers a mathematically based approach to assessing player positions when forming balanced team lineups from the perspective of their maximum effectiveness.

The goal of the work is to improve the quality of team formation, taking into account the characteristics of each player, in order to increase the effectiveness of matches by maximizing the use of all the strengths of specific players by applying methods of Data Mining.

2. Exploratory data analysis of sports team datasets

Working with sports team datasets has several important features:

- Multidimensionality of data - in sports, diverse information is collected: physical indicators of players, tactical parameters, match statistics, player ratings, GPS tracker data, etc.
- The need for real-time analysis - many decisions are made directly during the game or training based on operational data.
- Seasonality and cyclicity - the data have a pronounced periodicity (pre-season training, regular season, playoffs), which affects their interpretation.
- Contextuality - statistics should be considered in the context of the opponent, weather conditions, injuries, and the team's playing style.
- Interdependence of indicators - individual player data is inextricably linked to team results.
- Use of predictive analytics - to predict results, optimize the lineup, and prevent injuries.
- The need for visualization - complex datasets often require a visual representation for coaches and players.

Modern sports teams often have separate analytics departments that work with specialized tools to collect, process, and analyze this data to make strategic decisions.

2.1. Dataset characteristics

The analysis uses the FIFA 21 Complete Player Dataset, which contains detailed information about football players in the FIFA 21 game. This dataset was created based on official statistics and player ratings provided by the game developers EA Sports [18].

The dataset presents a large number of parameters, including the player's overall rating, his individual characteristics, the positions he can play in, and the ratings for each of these positions. This dataset is popular in football analytics, as it contains structured information about real-world player attributes and could be used for various studies in the field of sports data analysis.

This dataset allows for in-depth analysis of the relationship between physical, technical, and tactical characteristics (independent variables, Table 1) of players and their performance in different positions (dependent or target variables, Table 2). In our study, the dataset has been used to build machine-learning models that will help identify key characteristics that influence a player’s position choice on the field.

These characteristics (Table 1) have been used to build a decision tree model that allows us to determine which factors have the greatest impact on a player’s performance at each position. The player's ratings at different positions are used as target variables. They are presented as separate variables (Table 2).

Table 1
Independent variables or physical, technical, and tactical characteristics

Characteristic	Description
Pace	Shows how fast a player can move around the field
Shooting	Assesses the accuracy and power of shots on goal
Passing	Characterizes the accuracy and range of passes
Dribbling	Determines the player's ability to keep the ball and beat opponents
Defending	Reflects the player's ability to win the ball and defend in a positional manner
Physicality	Includes the player's stamina, strength and balance

Table 2
Target variables or player's ratings at different positions

Variable	Position	Description
ST_rating	Strike	the player's rating as a central forward
CF_rating	Center Forward	the player's skill rating in this role
RW_rating LW_rating	Wing Forwards	the player's level of play on the right and left winger positions
CAM_rating	Attacking Midfielder	the player's rating as a playmaker
CM_rating	Central Midfielder	the player's ability to act as a link between the defense and the attack
CDM_rating	Defensive Midfielder	the player's level of play in the defensive midfield zone
CB_rating	Center Defender	the player's ability to defend in the center of defense
RB_rating, LB_rating	Wing Defenders	the player's effectiveness on the right and left flanks of defense

2.2. Data preprocessing

Before building the model, data preprocessing was performed, taking into account the specifics of sports team datasets to ensure their correctness and quality [19]. The main steps of this stage included:

1. Removing missing values – all records containing missing values in key characteristics (speed, shooting, passing, dribbling, defense, fitness) were removed, as they could affect the accuracy of the model.
2. Converting ratings to numeric format – the ratings of players at each position in the original data were stored as text values (e.g., "90+3"), which could create problems when using them in the analysis. To eliminate this drawback, the "+" symbol and all additional values were removed, leaving only the numeric part.
3. Selecting relevant variables – from all available player characteristics, only those used in the model were selected: speed, shooting, passing, dribbling, defense, fitness, as well as player ratings at different positions.

In the original dataset, player positions are represented as text values, which can contain multiple position options for a single player (e.g., "CM, CDM"). To correctly use this data in the model, the following steps were performed:

1. Primary position allocation – if a player had multiple positions, the first one listed was used, as it is the primary one in the game.
2. Position conversion into categories – positions were grouped into more generalized categories:

- Forwards: ST, CF, LW, RW.
- Midfielders: CAM, CM, CDM, LM, RM.

- Defenders: CB, LB, RB.

3. Conversion into factor variables – for use in machine learning algorithms, positions were coded as factor variables.

This conversion made the model more generalized and convenient for analyzing player characteristics in terms of primary positions [20].

Since the chosen decision tree method is not sensitive to the scale of the data, normalization was not a mandatory step. Decision trees work by comparing and splitting data using thresholds. Key reasons for decision trees being scale-insensitive:

- Decision trees make decisions based on comparing values (greater than/less than), not their absolute values.
- When building a tree, the algorithm looks for optimal split points for each feature, regardless of its scale.
- The metrics used to choose the best split (e.g., entropy, Gini index) are independent of the scale of the data.

This property makes decision trees particularly useful when working with heterogeneous data in sports analytics, where metrics can have different units of measurement and value ranges. In this study, all numerical characteristics such as speed, shooting, passing, dribbling, defense, and fitness were kept in a comparable range (0 to 100), which made additional normalization unnecessary. Therefore, all numerical values were used in their original form without scaling.

3. Building a models

3.1. Model training and validation

After defining the target and independent variables, the decision tree model was trained. The rpart() function was used to analyze the relationships between a player's characteristics and his ranking at a certain position. The rpart() function from the rpart package in R is used to build decision trees. It is used for classification and regression problems using the CART (Classification and Regression Trees) algorithm.

The model parameters (Fig.1) were adjusted with the "control" argument to provide an optimal balance between accuracy and complexity: "cp" = 0.001 (complexity parameter) allows for deeper trees and prevents overtraining, "maxdepth" = 10 defines the maximum depth and "minsplit" = 5 provides a sufficient number of observations for node splitting [22].

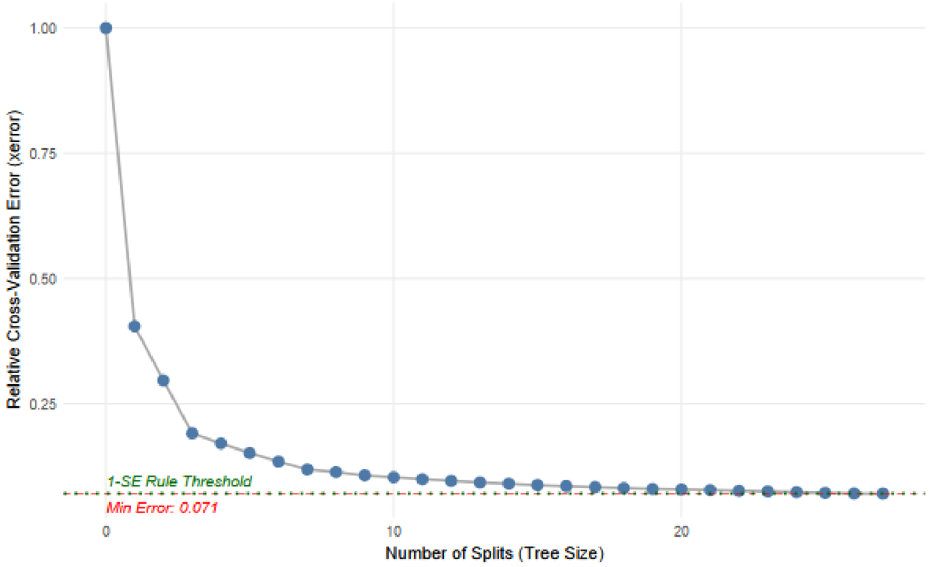

Figure 1: Selecting the optimal tree size using the complexity parameter

A linear regression model was also trained using the lm() function and a support vector method using the svm() function. For these methods, a formula was used where we defined the dependent and independent variables. Obviously, the independent variable will be the position rating, and the dependent variables are the characteristics under consideration. For the support vector method, normalization was also performed to speed up the model execution, using the scale function parameter. After running all three models, we also perform the ensemble voting method. The soft voting method was used because hard voting does not work with regression problems.

Information was obtained about the linear regression models and the support vector method (Tab. 2, 3). The output of the linear regression model was demonstrated on the example of the fullback position. The model explains the target variable very well ($R^2 > 0.98$), and all features are significant. The main contribution to the forecast is made by defending (0.52) and passing (0.197), and shooting has a negative, but very small effect - this may indicate a correlation with other variables.

The output of the support vector method model showed that the SVM model for regression has an RBF kernel, i.e., a nonlinear relationship between the features and the target variable is expected. The settings (cost = 1, gamma ≈ 1/6, epsilon = 0.1) are typical or chosen by default - perhaps they can be optimized through cross-validation. The number of support vectors 4071 is quite a lot, but with a large set, as in our case, this is normal [23, 24].

Table 3
Detailed information about the linear model

Characteristic	Beta	95% CI	p-value
Pace	0.15	0.15, 0.15	<0.001
Shooting	-0.02	-0.02, -0.02	<0.001
Passing	0.20	0.20, 0.20	<0.001
Dribbling	0.10	0.10, 0.11	<0.001
Defending	0.52	0.52, 0.52	<0.001
Physic	0.08	0.08, 0.09	<0.001

Table 4
Detailed information about the SVM model

Parameter	Value	Description
Kernel	Radial (RBF)	Allows capturing non-linear patterns
Type	eps-regression	Standard SVR formulation
Cost (C)	1	Penalty for misclassification
Epsilon (ε)	0.1	Margin of tolerance for errors
Gamma (γ)	0.17	Kernel coefficient

The model was trained on a training dataset that was 80% of the total dataset, using independent variables (player characteristics) to predict player ratings at a given position. Accuracy was assessed using mean absolute error (MAE) and root mean square error (RMSE), which quantified the difference between predicted and actual player ratings. In addition, a feature importance analysis was performed to determine which factors had the greatest impact on player performance at each position [25, 26].

3.2. Model accuracy assessment

After building the above models, an analysis of the importance of the characteristics that affect the player's rating was conducted for each position. For this, information on the importance of variables was used, which is automatically calculated by the "rpart" function in the "variable.importance" parameter for the decision tree, for linear regression, the model coefficients are extracted from each player's position, and then the absolute value of the coefficients is taken, and for the support vector method, the DALEX library and the model_parts() function are used, which implements the feature clipping method.

For each built model, it was checked whether it contained important variables (model$variable.importance). If the decision tree did not have significant branches, such a case was ignored. Data on the importance of the characteristics were converted into a tabular format (data.frame), where the name of the characteristic and the value of its influence were stored.

In linear regression, the weight (coefficient) shows how much the target variable changes when the characteristic changes by 1 (other things being equal). The larger the absolute value of the coefficient, the stronger the influence of the characteristic on the forecast. This method is straightforward and interpretable — in a linear model, importance = |coefficient|.

The feature clipping method means that each feature is shuffled randomly in turn, and the mean square error (RMSE) of the model result is estimated. If the shuffle of a feature greatly increases the error, then it is important. This is a black-box method that works even for complex/nonlinear models (like SVM with an RBF kernel).

Each entry was accompanied by information about the player position for which the models were built. All the results obtained were combined into a common dataset for further analysis. The importance of the features was presented in the form of a bar chart (Fig. 4), which allows you to easily compare the importance of each feature for different positions and each model. For this, the ggplot2 library [27] was used, which displayed the variables and their importance, grouping the data by player positions.

To provide a detailed overview of model performance across different field positions, a comparison was conducted between predicted and actual ratings using the MAE and RMSE metrics for each method. The results for all positions and models are summarized in the following tables. This breakdown allows a side-by-side evaluation of how each model performs under different positional requirements and highlights the stability and strengths of each approach.

The accuracy of the model for different positions (Tab. 5) indicates relatively small deviations (up to 3%) of the predicted values from the test values, which confirms good generalization.

Table 5
Comparative Analysis of Model Performance across Player Positions

	Decision Tree		Linear Regression		SVM (RBF)		Ensemble	
Position	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
ST	1.94	2.45	1.30	1.65	1.10	1.41	1.23	1.56
CF	1.87	2.37	1.07	1.38	1.00	1.30	1.13	1.45
RW/LW	1.91	2.43	0.85	1.11	0.81	1.06	1.01	1.28
CAM	1.80	2.30	1.02	1.31	0.97	1.26	1.09	1.40
CM	1.92	2.44	1.16	1.47	1.11	1.42	1.23	1.56
CDM	2.02	2.55	1.03	1.31	0.98	1.27	1.15	1.46
CB	**1.76**	**2.24**	**0.82**	**1.04**	**0.73**	**0.95**	**0.89**	**1.13**
RB/LB	2.07	2.61	0.99	1.28	0.97	1.26	1.13	1.43

MAE indicates the average absolute difference between predicted and actual ratings. For example, for ST (forwards) of the decision tree model, MAE ≈ 1.94 means that the predicted rating differs from the actual by an average of 1.94 points (for a 100-point scale). RMSE gives more weight to large errors because it uses squared deviations. RMSE is always slightly larger than MAE, because it is more sensitive to large deviations. The support vector model coped best with this task, even better than the ensemble method, which indicates that if the parameters are tuned in the best way, this model will be even better than the others, but it still shows better accuracy results. The accuracy of the model is quite high, because MAE ≈ 0.73–1.11 means that the error in the rating prediction is only about 1 point. The best prediction for central defenders (CB, MAE = 0.73) - the model most accurately determines the rating for this position. The worst prediction is for central midfielders (CM, MAE = 1.11). Possible reasons: greater influence of specific characteristics that the model does not take into account, or more significant differences between players in this position. The overall error is uniform for most positions (≈0.8–1.0 MAE), indicating the stability of the model in prediction.

3.3. Algorithm of Intelligent Team Formation and Feedback Loop

To implement an effective system for intelligent team formation using machine learning methods, it is necessary to structure the process into clear, iterative stages. This enables not only the initial construction of the team based on player attributes, but also the refinement of the system based on match results and tactical feedback. The proposed algorithm consists of the following steps (Fig. 2):

Gathering structured and relevant information about players, including their physical, technical, and tactical characteristics, as well as historical performance data. This step forms the foundation for any further analytical modeling.

Training predictive models (e.g., decision tree, linear regression, SVM) to forecast the player's effectiveness at different positions based on collected characteristics. Each model is trained on historical data and tuned for accuracy.

Models are validated using metrics such as Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). The best-performing model is selected based on generalization ability and accuracy across positions.

Using the predicted ratings and selected model, a team composition is formed that best fits the intended tactical scheme (e.g., 4-2-3-1, 4-1-4-1). The model selects players with the highest predicted performance for each role.

The formed team participates in a game or simulation. The effectiveness of the team is observed in real conditions, allowing the assessment of how well the predicted strengths align with actual outcomes.

After the match, the actual performance of the team and individual players is analyzed. Discrepancies between predicted and real effectiveness are noted, and misalignments in role suitability are identified.

Based on post-game evaluation, the model's parameters can be adjusted. This includes refining feature importance, tuning hyperparameters, and possibly updating the dataset with new observations, thus creating a feedback loop that increases model precision over time.

Such an iterative cycle ensures not only high initial accuracy but also the ability to adapt to changing team dynamics, individual player development, and evolving strategies. The feedback loop (from Step 6 to Step 2) is a key mechanism for continuous improvement.

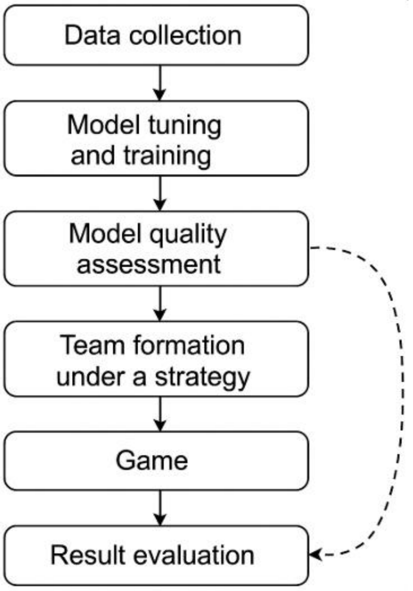

Figure 2: Algorithm for intelligent team formation and feedback loop

4. Analysis of results

Only decision trees have visualization as such, which is a plus for this model, but what is the point of this visualization if it is more inaccurate than others and has different results, so the visualization output for the CB position was demonstrated, and the importance of characteristics will be best seen on the general graph for the three models.

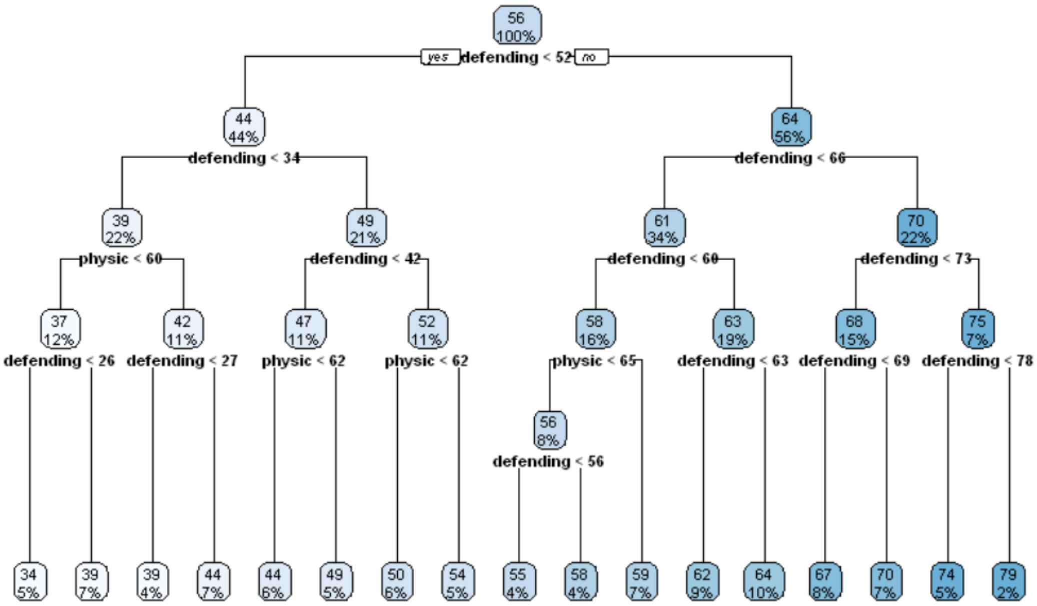

Figure 3: Decision tree for cb

As a result of the study of positions and their influence on the quality of the game (Fig. 4).

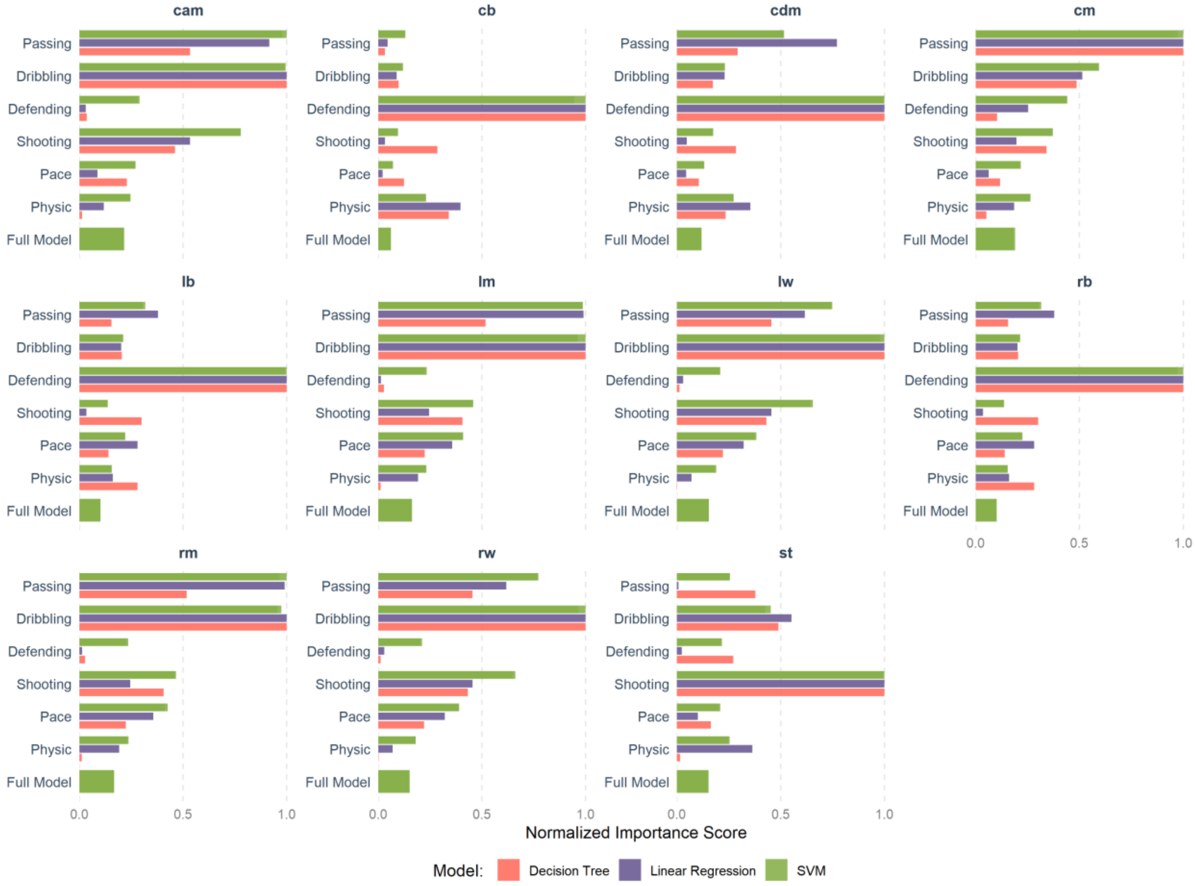

Figure 4: Comparison of Normalized Importance across different positions and models

Let's start the analysis in the order of the positions in the picture. The attacking midfielder position has 3 important characteristics: dribbling, passing, and shooting. The support vector method says that there is no characteristic that does not affect the assessment, while the decision tree says about two characteristics: defense and physics, and linear regression only defense, that they do not affect. That is, a more qualitative model found the influence of these two factors on the quality of the game in this position. That is, the support vector method model says that in this position, the player must be a universal player and have all the qualities in his arsenal.

The central defender position has only one most important characteristic, which is defense, but the physical data indicator also has a good influence. However, the physical indicator has a greater influence on the opinion of linear regression and decision trees than the support vector method, but this method revealed at least some need for passing skills, unlike other models.

Now let's consider the position of the defensive midfielder. According to all models, the most important indicator is defense. The decision tree believes that to the same degree, the indicators of physics, shooting, and passing are good indicators of importance in approximately equal degrees. Linear regression says that the indicators of speed and shooting are unnecessary, and the indicator of passing is very influential, but not the most important, and the indicator of physics also has a good influence. The support vector method says that the indicator of passing has a good influence, and the indicator of physics has a good influence; it considers all other characteristics not very influential, but it would also be nice to have them. Therefore, we can say that the defensive midfielder is the same defender, but must be able to start the attack with passes, and also be more versatile than the central defender.

The center forward position is quite interesting because the models disagree on which metric is most important. The decision tree and linear regression model say that dribbling is most important, and the support vector method says that shooting, but it should be noted that this model says that dribbling has a very strong influence, and the other two models, that shooting also has a strong influence, so to speak, said the opposite. Also, all the models said that passing is also an important metric. The other metrics are also important, but the linear regression says that defending in this position is completely unnecessary, which is quite interesting.

For the central midfielder position, all three models ranked passing ability as the most important characteristic, and dribbling as the second most important characteristic. The support vector method again emphasizes the versatility of players, and the main characteristic of the least important is defense. The linear regression model also ranked them in the same order of importance as the support vector method, but gave them a lower score, especially speed, which it considers unimportant. The decision tree, on the other hand, gave importance only to shooting, so the decision tree is quite different from other models in taking into account secondary characteristics for this position in terms of importance.

The full-back position also has the most important indicator according to the three models, and this is obviously defense. The support vector method and the linear regression model highlight passing ability, but in general, consider all other characteristics not very important; however, it is desirable to have, except for one characteristic of hitting, the linear regression model considers this indicator unnecessary. The decision tree highlights the indicators of physics and hitting, which are very different from the linear regression model. This model also says that passing and speed are not particularly important, which is also different from the other two models. It is interesting that all the indicators are so different from each other in different models, but in the dribbling indicator, there is absolute equality.

At the winger position, all models said that dribbling was the most important attribute, along with two important attributes, passing and shooting, but the support vector method gave them more importance. Basically, all models said the same about their top importance attributes, but unlike the support vector method, linear regression model, and decision tree, the models said that the physics and defense attributes were completely unnecessary, especially the decision tree.

The last position is striker, where three models say that the main characteristic is hitting, and it is also important to be able to beat opponents. The support vector method says that other characteristics are quite important to an approximately equal extent. The linear regression model says that defense and passing are completely unnecessary, and speed is not very necessary; it only highlights physics, and more than others. The decision tree considers physics unimportant, and speed is not particularly important, also speed is not particularly important, and the others are important.

5. Conclusions

The results of modeling player ratings by position largely coincide with expert assessments. Notably, speed appeared auxiliary rather than key in decision trees. For attackers, shooting and dribbling are most important; for midfielders, versatility and passing; for defenders, defensive skills.

Decision trees proved interpretable but less accurate (MAE 1.76–2.07), suitable mainly for explanatory analysis. Linear regression performed better (MAE 0.82–1.3) and is fast and simple, though less effective for nonlinear patterns. Support Vector Machine (SVM) achieved the highest accuracy (MAE 0.73–1.11), showing strong generalization and ability to model complex relations, but requires more computational resources and is less interpretable. Ensemble methods yielded moderate results and were outperformed by SVM, except for the striker position.

SVM appears most promising for assisting in role selection, especially at early stages of a player's career or when reassigning roles. For example, a player with high speed and shooting may be well-suited for a striker role, while strong passing and dribbling may indicate a midfielder.

It is important to note that these models rely on predefined quantitative characteristics and do not account for tactical context, individual play style, or team cohesion. Model accuracy also varies by position – predictions for central defenders were more precise than for midfielders.

Therefore, machine learning should be seen as a complementary tool, best used in tandem with expert tactical analysis. Continued refinement of models based on match data can support more accurate lineup decisions and personalized strategies.

Future directions include improving the integration of playing style into model tuning and identifying statistical correlations between model-generated lineups and actual match outcomes. This would contribute to developing automated, intelligent systems for sports analytics. Further advancements could also reduce player acquisition costs, support talent scouting, and optimize resource allocation across infrastructure and social initiatives.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] M. Osadets, Features of tactical training of football players, Youth and the market 7 (2015) 126–130. URL: http://nbuv.gov.ua/UJRN/Mir_2015_7_26.

[2] V. Khomenko, Modern tactical constructions of the game of leading European football clubs in 2018, SI 2(12) (2024) 59–70.

[3] S. G. Pryimak, A. V. Zavorotynskyi, Decision trees and their application for classification of students of different groups of sports and pedagogical improvement, 2018.

[4] A. Fedetskyi, Method of sigmoid deviations and regression scale in modeling the technical fitness of football players, Physical education, sport and health culture in modern society 3 (35) (2016) 104–109.

[5] M. S. Martsenyuk, et al., Analysis of methods for detecting disinformation in social networks using machine learning, Cybersecurity: Education, Science, Technology 2 (22) (2023) 148–155.

[6] A. A. Astrakhantsev, et al., Investigation of the effectiveness of machine learning algorithms for traffic classification in mobile networks, Problems of Telecommunications 1 (30) (2022) 3–17.

[7] N. Guliyev, Research of methods for constructing decision trees for the implementation of the random forest algorithm in the medical field, Measuring and Computing Devices in Technological Processes 1 (2025) 36–43.

[8] A. O. Otroshchenko, Business analysis and modeling of decision-making processes in project management, 2025.

[9] A. Kolomiiets, I. Miroshnychenko, V. Ziuziun, N. Datsenko, T. Kmytiuk, Development of Project Management Models for Information Systems to Improve Website SEO Metrics, in: XI International Scientific Conference "Information Technology and Implementation" (IT&I 2024), Vol. 3909, 2024, pp. 334–345. URL: https://ceur-ws.org/Vol-3909/Paper_27.pdf.

[10] O. M. Berezky, et al., Application of the linear regression method for the analysis of quantitative characteristics of cytological images, Ukrainian Journal of Information Technology 3 (1) (2021) 73–77.

[11] O. I. Sheremet, O. V. Sadovoi, Support Vector Method (SVM), Mathematical Modeling 1 (2013) 13–17.

[12] N. Koshkina, On increasing the accuracy of JPHIDE steganogram detection, Physical and Mathematical Modeling and Information Technologies 32 (2021) 170–174.

[13] W. Bi, Y. Zhao, H. Zhao, Predicting sports injuries using machine learning: Risk factors and early warning systems, Molecular & Cellular Biomechanics 22 (2025) 335. doi:10.62617/mcb335.

[14] E. Wang, X. Yin, Y. Li, T. Wang, Sports Betting: An Application of Machine Learning to the Game Prediction, Applied and Computational Engineering 132 (2025) 104–118. doi:10.54254/2755-2721/2024.20626.

[15] L. Zhang, Y. An, Enhancing Sports Team Management Through Machine Learning, IEEE Access (2025). doi:10.1109/ACCESS.2025.3551889.

[16] A. Obradović, D. Kečo, Sports Results Prediction Model Using Machine Learning, SAR Journal - Science and Research (2024) 184–189. doi:10.18421/SAR73-03.

[17] Exploration of machine learning based on big data in sports models and physical education teaching, Molecular & Cellular Biomechanics 22 (2025) 940. doi:10.62617/mcb940.

[18] FIFA 21 complete player dataset. URL: https://www.kaggle.com/datasets/stefanoleone992/fifa-21-complete-player-dataset?resource=download&select=players_15.csv.

[19] V. B. Ilnytskyi, Development of a recommender system for audio content using machine learning methods without a teacher, 2025.

[20] S. A. Tristan, Research of methods for analyzing customer feedback on employees for the IS of a product company, 2025.

[21] V. V. Aksenov, Research of methods and means of automated team formation for work in IT projects, 2025.

[22] rpart function – Rdocumentation. URL: https://www.rdocumentation.org/packages/rpart/versions/4.1.24/topics/rpart.

[23] lm function – Rdocumentation. URL: https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/lm.

[24] svm function – RDocumentation. URL: https://www.rdocumentation.org/packages/e1071/versions/1.7-16/topics/svm.

[25] B. Fu, et al., Predictive modeling for durability characteristics of blended cement concrete utilizing machine learning algorithms, Case Studies in Construction Materials 22 (2025) e04209.

[26] Y. Cheng, et al., A robust framework for accurate land surface temperature retrieval: Integrating split-window into knowledge-guided machine learning approach, Remote Sensing of Environment 318 (2025) 114609.

[27] Create Elegant Data Visualisations Using the Grammar of Graphics • ggplot2. URL: https://ggplot2.tidyverse.org/.