Stacked Ensemble-Based Framework for Predicting Market Value and Tactical Fit in Football Transfers

Shubh Harde
Artificial Intelligence and Data Science
shubhharde158@gmail.com

Vedant Bhawnani
Artificial Intelligence and Data Science
vedantbhawnani@gmail.com

Shruti Savant
Professor, Artificial Intelligence and Data Science
shrutisavant@gmail.com

***Abstract*—The transfer markets in football are a dynamic and pivotal period which entails a state of volatility where teams and clubs strategize and aim to strengthen their teams. This is more often than not, preceded by in-depth analysis of data related to hundreds of promising and prospective players to find the perfect replacement for the outgoing player. This analysis currently includes labour intensive tasks of manual video reviews and scouts going to games to watch a prospect play. This paper introduces the Football Player Replacement Finder, a novel approach to reduce the complexity and time required for scouting and acquiring impactful talents by using advanced machine learning models and automated data scraping pipelines. Our system employs supervised models for gauging the performance and price of football players along with clustering techniques for player profiling, enabling stat-by-stat comparison of players. By integrating advanced metrics along with appealing visualisations, our system empowers decision-makers to streamline their scouting process and uncover valuable talents effectively.**

***Index Terms*—Football analytics, Player replacement, Machine learning in sports, Player performance prediction, Transfer market strategy, Clustering for player profiling, Football scouting automation**

I. Introduction

The transfer window months of January and July each year are one of the most critical time periods for any football club looking to strengthen their squad. In this period, the clubs focus heavily on identifying potential signings, while keeping a close eye on the departing players. Each club has to identify suitable replacements while ensuring that they do not overpay for a player as adhering to the budget is a key requirement for any club. Scouting players and ensuring that the scouted players are within the budget, is a monumental task that requires a deep analysis of a wide range of factors, such as player statistics, market conditions such as inflation and volatility, the team effect on team dynamic of incorporating a new player, and any ego clashes that may arise in the dressing room among players. Traditionally, this process includes extremely laborious scouting methods, involving going through extensive video footage, gathering subjective reports and insights, and manually comparing hard statistics of players. These methods have always been time consuming and with the exponential increase in the amount of data and advanced metrics, it is only getting increasingly difficult to keep up with the demands of modern football. The recent advancements in football analytics have introduced multiple data-approaches to player evaluation. Platforms such as FBref [1] and Sofifa [2] offer vast amounts of data and player statistics and attributes, which have become an important tool for the recruiters. Despite the availability of these insights, the process of finding replacements remains labour-intensive. Traditional methods of scouting do not generally incorporate data driven approaches while handling larger and more complex datasets in the modern era of players scouting. This paper proposes an advanced system of player recruitment and replacement process by automating data collection, analysis and comparison. The proposed system uses machine learning models, including supervised learning and clustering algorithms to predict player performance and identify statistically similar players. By making use of similarity metrics such as Euclidean distance, cosine similarity or the Pearson Correlation, the system enhances the accuracy of player comparisons, ensuring a more relatable match between the outgoing player and the potential replacement of the outgoing player, The integration of automated data scraping pipelines allows for faster and more effective comparison of players based on on-field metrics such as expected goals, expected assists, goals, assists, passing accuracy, progressive carries and other related statistics. These metrics, combined with similarity measures, create a comprehensive view of player performance, enabling deeper and more accurate analysis, ensuring that the players are suitable not only for the immediate requirements of the club, but also with the tactical approach of the manager and the coaching staff along with the long-term goals of the club. By reducing the reliance on manual processes, risks like human errors, and biases are eliminate that are brought into the system by scouts or managers' prejudices. Player Replacement Finder aims to enable clubs in easier and faster identification of potential players with greater accuracy. The proposed system not only enhances the precision of player comparisons models, but also offers intuitive visualisation approaches that helps clubs take faster decisions in choosing the right candidate and saving them from over spending unnecessarily. This approach has the potential to optimise transfer strategies, which indirectly also affects the team performance. The remainder of this paper is organised as follows: Section 2 reviews the current literature on football player replacement and data analytics already accomplished in this domain. Section 3 outlines the methodology used to develop the Player Replacement Finder system, including the data sources and machine learning models employed. Section 4 presents the results and validations of the system, and section 5 concludes with discussion of the system's impact on modern football transfers and potential future development.

II. Literature Review

The prediction of football players' market value has been studied widely, with researchers employing different methods, datasets and algorithms to enhance prediction accuracy. This section encapsulates a comprehensive literature review focused on predicting the market value of players through machine learning and deep learning algorithms. Few researchers have found out some parameters which play an important factor in predicting the market value of a player. Age is included as one of the important factors, with young players demanding more value due to their potential for growth. Similarly, players with high popularity tend to bring their fans with them, leading to an increase in their market value [3]. The position a player plays is also thought to be important, with attackers usually having higher market values [4]. Beginning with Carmichael & Thomas (1993), economists have used regression models to identify the determinants of transfer fees [5]. Since then several researchers have followed a similar trend, with many researchers following regression-based models to forecast the player transfer fees. Mustafa A and Al-Asadi used data from FIFA as true values and used different linear and non-linear models to predict the market price of a player [6].

They employed 9 parameters, such as international reputation, weak foot column, etc. Stanojevic and Gyarmati presented their research on statistical measures, and obtained data from sports analyst company InStat, and Transfermarkt [7]. The research, following a more traditional approach, aimed to estimate the market value of 12,858 players based on player performance metrics. Clustering techniques were deployed to analyze player performance data, and the model was built on 45 predictors, with their results outperforming widely used transfermarkt.com market value estimates, paving way for more precise predictions with the help of data analytics and finer data [8]. Muller et al. [3] presented a data-driven approach to overcome the limitations of crowd-sourcing. The researchers used the data from top 5 European leagues. They created a dataset using the attributes such as player characteristics - age,position, nationality. Muller et al. took a unique approach by including data from Wikipedia, Facebook and Google metrics at that time. They employed a linear regression model, and results achieved were within the scope of crowdsourced estimates . Dobson et al. [9] explored the effects of player metrics on transfer fees and discovered that transfer fees are volatile across segments even in a single competition. More recently, Depken II & Globan use linear regression to identify that English clubs pay a premium in the transfer market, compared to clubs from other European countries.[10] Yigit et al. presented an innovative approach to assessing the player values. They leveraged a wide range of player attributes, including on-field performance metrics, demographic data, and market factors, to predict player market values. The dataset comprised football players from major leagues. 5316 players from 11 major leagues across Europe and South America were considered. Data from the football manager simulation game was collected and merged with the transfer value from Transfermarkt. The most resultant values were in accordance with the current market values. [11] Behravan et al. took a distinctive approach to predict the market values of a player, by employing Particle Swarm Optimization. The data collected was from the FIFA 20 dataset, and the value of a player in the dataset was considered the true value. The players were divided into 4 clusters based on positions using an automatic clustering algorithm. According to the authors, the RMSE and MAE for their method are 2,819,286 and 711,029,413, respectively, while the results by Muller[6] were 5,793,474 and 3,241,733. These results indicate that their methods had a significant advantage over other methods.[12] Ian et. al [13] investigated the use of machine learning to estimate transfer fees, utilizing data from sofifa.com and transfermarkt.com. They trained both linear regression and XGBoost models on a range of performance metrics, including those from Instat and GIM performance ratings. Their findings indicated that the XGBoost model outperformed the linear regression model in predicting transfer fees. This research highlights the potential of machine learning to inform transfer decisions, addressing the question of "what is the expected fee of a player given their past performance?" The authors suggest further work to assess the "reasonableness" of transfer fees based on post-transfer performance, potentially leveraging the same data sources and machine learning techniques.

III. Methodology

The proposed system uses advanced on-field player performance metrics, domestic and continental performance of football clubs, players' perceived market value, player injury data and domestic competition metrics like standings and top performers. The data is collected from FBref and Transfermarkt using automated web scraping scripts. The collected data is put through a data processing pipeline which cleans and transforms the data to match the needs of the system. The preprocessed data is used by machine learning models for calculating consolidated metrics for comparing players and teams, determining similar players, formation fit analysis and market price prediction. Figure 1 shows the proposed methodology.

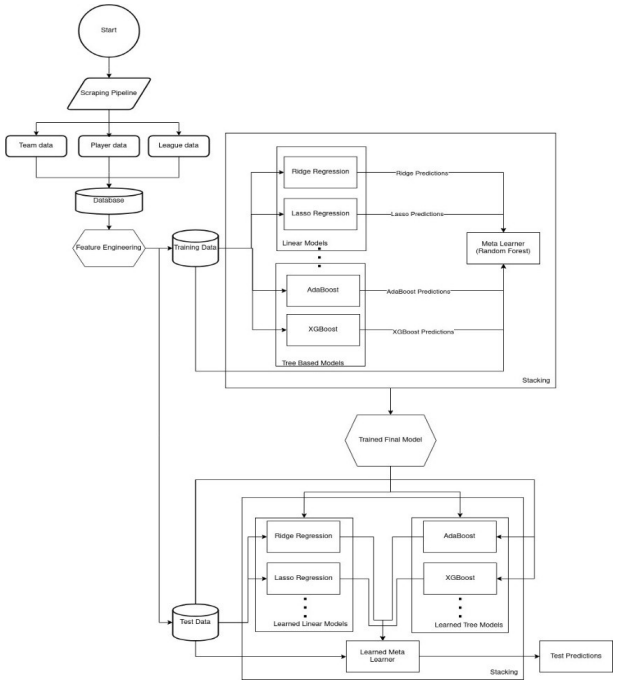

Fig. 1. *Proposed Methodology*

A. Data Pipeline

The data pipeline serves as the backbone of the system, ensuring that the data is systematically collected, cleaned and stored for efficient use for accurate analysis. The pipeline automates the extraction and preparation of data to be used in machine learning models.

B. Data Collection

The first step in the data pxipeline is collection of data from external sources. The data is collected primarily from Fbref and Transfermarkt. Fbref provides data ranging from basic statistics for players like nationality, height, to advanced on-field metrics such as expected goals and expected assists covering domestic leagues over forty countries. Transfermarkt gives detailed data about player transfers, including transfer fees, contract duration and clubs involved. Utilizing data from both the sources has offered a comprehensive perspective to the system. Multiple automated custom web scrapers were developed to systematically extract data related to players, teams, and the top five European domestic leagues. The scrapers are as follows:

1) League Standings: The system incorporates a web scraper tailored to retrieve the standings table of the top five domestic competitions in Europe allowing automated collection of team level data such as position, points, wins, losses and goal scoring statistics. Algorithm 1 shows the algorithm for the League Standings Scraper.

2) Squad Stats Scraper: The automated squad stats scraper extracts team level data from Europe's top five leagues. The extracted data includes a vast set of tables which cover various performance dimensions such as standard stats (matches played, goals, xG), goalkeeping and advanced goalkeeping (save rates, post shot xG, etc.), shooting (shot volume, conversion), passing and pass types (progressive passes, pass lengths), goal and shot creation (SCAs, GCAs), defensive actions (tackles, interceptions), possession (carries, take-ons), playing time (minutes played, starts), and miscellaneous metrics (fouls, aerial duels, cards). This scraper enables the system to work with detailed and comprehensive statistical squad-level

Algorithm 1 League Standings Scraper

procedure PATHDECIDER($league$)
    $folder\_path \leftarrow$ "assets
Data
textbackslash" + $league$
    if folder_path exists **then**
        return $folder\_path$
    else
        Create directory at $folder\_path$
        return $folder\_path$
    end if
end procedure
procedure LEAGUESTANDINGSCSV($league$)
    $link \leftarrow$ lookup $league$ in $league\_links$
    $df \leftarrow$ read HTML tables from $link$
    Drop columns "Notes" and "Attendance" from $df[0]$
    $folder\_path \leftarrow$ PATHDECIDER($league$)
    $csv\_path \leftarrow$ join $folder\_path$ and "$league$_Standings.csv"
    Save $df[0]$ as CSV to csv_path without index
end procedure

performance evaluation. Algorithm 2 shows the algorithm for the Squad Stats Scraper.

3) Top Performers Scraper: The Top Performers Scraper extracts data related to the top performing players from Europe's top five leagues. The data includes information about the player with the most goals (number of goals, team name, player image link), the player with the most assists provided (number of assists provided, team name, player image link) and the goal keeper with the most clean sheets kept (number of clean sheets, team name, player image link). Algorithm 3 shows the algorithm for the Top Performers Scraper.

4) Player Stats Scraper: The Player Stats Scraper extracts player performance data. The data includes a wide range of performance metrics distributed in various categories. These include Standard Stats (appearances, minutes played, goals, and assists), Shooting (number of shots, shots accuracy, xG, xGoT), Passing (total passes, completion percentages, key passes), Pass Types (e.g. long passes, through balls, switches), Goal and Shot Creation metrics that capture both direct and indirect contributions to scoring opportunities, and Defensive Actions (tackles, interceptions, blocks). Possession-related (carries, touches, dribbles) as well as playing time (minutes per appearance, starting/substitution patterns) are also retrieved. This collection of scraper data allows the system to perform a detailed analysis of the contribution of a player throughout the season. Algorithm 4 shows the algorithm for the Player Stats Scraper.

5) TransferMarkt Scraper: Using the Transfermarkt scraper, the system is served with crucial off-field player metrics such as market value, injury status, and contract length, which are essential metrics consumed by the machine learning models for predicting the market prices of the players. Algorithm 5 shows the algorithm for the Transfermarkt Scraper.

C. Data Preprocessing

Data preprocessing is an integral part of the methodology as it provides the system with clean and accurate data. The data preprocessing pipeline was implemented as a multistep approach to ensure the reliability and usability of the data. First, data from Fbref.com and Transfermarkt.com were consolidated based on the player last name and age. In instances where the player last names were the same, the positions of players were taken into account for correct aggregation. Secondly, several values were missing or zero by default (e.g., xG for most goalkeepers). To tackle such discrepancies, any row with

Algorithm 2 Squad Stats Scraper

Input: URL of a team page from FBref
Output: Dictionary of DataFrames for multiple squad stats categories
procedure TEAMSCRAPER(url)
    Set `self.url` to given url
end procedure
procedure GET_DFS
    Call `get_all_team_stats_from_URL(self.url)`
    return Dictionary of DataFrames
end procedure
procedure GET_ALL_TEAM_STATS_FROM_URL(url)
    tables ← `_gel_all_team_tables(url)`
    if tables == -1 **then**
        return -1
    end if
    Initialize empty dictionary `dfs`
    for all table in tables **do**
        df ← `_get_dataframe(table)`
        category ← table caption text before colon
        Add df to `dfs[category]`
    end for
    return `dfs`
end procedure
procedure _GEL_ALL_TEAM_TABLES(url)
    res ← Send HTTP GET request to url with timeout
    if ReadTimeout occurs **then**
        Sleep and retry
    end if
    soup ← Parse response with BeautifulSoup
    tables ← Find tables with class "stats_table"
    if tables is empty **then**
        return -1
    end if
    return tables
end procedure
procedure _GET_DATAFRAME(table)
    Read table HTML using pandas
    df ← First table from list
    Drop last column (e.g., Rk)
    Flatten headers if multi-index
    Reset index
    Drop columns: Age, Squad, Country, Comp, LgRank (if present)
    Convert all columns to numeric where possible
    return df
end procedure
procedure GET_LEAGUE_LEADERS
    res ← GET request to self.url
    soup ← Remove HTML comments and parse
    Find `div` with id "div_leaders"
    if div not found **then**
        return -1
    end if
    leaders ← All tables with class "columns"
    names ← Captions of leader tables
    Parse HTML tables using pandas
    Rename columns to Rank, Player, Value
    return Dictionary of leaders with names as keys
end procedure

Algorithm 3 Top Performers Scraper

Require: League name
Ensure: Dictionary of top scorer, top assister, and most clean sheets with images and teams

procedure FETCH_TOP_PERFORMERS(league)
    Initialize empty dictionary `best_performers`
    url ← `league_links[league]`
    response ← HTTP GET request to `url` with headers
    if response status code ≠ 200 **then**
        Print "Failed to fetch the page" and **return**
    end if
    soup ← Parse HTML with BeautifulSoup
    meta_block ← Find div with class "meta"
    paras ← Find all paragraph tags in `meta_block`
    for all paragraph p in `paras` **do**
        if p contains `<strong>`, `<a>`, and `<span>` **then**
            player ← Extract from `<a>`
            value ← Extract from `<strong>`
            team ← Extract from `<span>`
            if `"Most Goals"` in p **then**
                image ← IMAGE_GETTER(player)
                `best_performers["Top Scorer"]` ← [player, value, team, image]
            else if `"Most Assists"` in p **then**
                image ← IMAGE_GETTER(player)
                `best_performers["Top Assister"]` ← [player, value, team, image]
            else if `"Most Clean Sheets"` in p **then**
                image ← IMAGE_GETTER(player)
                `best_performers["Clean Sheets"]` ← [player, value, team, image]
            end if
        end if
    end for
    return `best_performers`
end procedure

procedure IMAGE_GETTER(player_name)
    normalized_name ← lower-case, hyphenated player_name
    search_url ← "https://fbref.com/en/search/search.fcgi?search=" + player_name"
    response ← HTTP GET request to `search_url`
    soup ← Parse HTML with BeautifulSoup
    if "Players from Leagues Covered" in soup text **then**
        search_results ← Find options in select tag
        for all option in search_results **do**
            if name in option matches player_name **then**
                player_url ← href of matching option
                response ← HTTP GET request to `player_url`
                soup ← Re-parse HTML
                break
            end if
        end for
    end if
    meta_div ← Find div with id "meta"
    media ← Find div with class "media-item" in `meta_div`
    if media contains image **then**
        return image URL
    else
        return `None`
    end if
end procedure

Algorithm 4 Player Stats Scraper

Input: `url` (FBref player URL), `stat` (optional)
Output: Processed player statistics as a DataFrame or dictionary of DataFrames

function AVAILABLESTATS
    return predefined list of stat categories
end function
function GETPLAYERSTATSFROMURL(url, stat)
    if stat not in AvailableStats() **then**
        Raise error
    end if
    table, rowCount ← EXTRACTTABLEFROMURL(url, stat)
    df ← FORMATDATAFRAME(table, rowCount, playerID)
    return df
end function
function GETALLPLAYERSTATSFROMURL(url)
    tables ← EXTRACTALLTABLES(url)
    if tables not found **then**
        return -1
    end if
    for all table in tables **do**
        if table's category in AvailableStats() **then**
            df ← FORMATDATAFRAME(table, rowCount, playerID)
            Add df to result dictionary
        end if
    end for
    return dictionary of DataFrames
end function
function FORMATDATAFRAME(table, rowCount, playerID)
    Parse HTML table to pandas DataFrame
    Drop unnecessary columns and rows
    Clean and convert data types
    Add playerID to DataFrame
    return cleaned DataFrame
end function

Algorithm 5 Transfrmarkt Scraper

Input: `club_id`, optional `season_id`
Format Transfermarkt URL using `club_id` and `season_id`
Request and load the webpage
if `season_id` is not provided **then**
    Extract it from the webpage
end if
Determine if club is for current or past season
for each player on the page **do**
    Extract:
    ID, Name, Position, Date of Birth, Age
    Nationality, Current Club, Height, Preferred Foot
    Joined On, Signed From, Contract Expiry
    Market Value, Status
end for
Store each player's data in a dictionary
Return dictionary with `club_id`, list of player dictionaries

TABLE I
GOAL KEEPERS

Name	Shot Stopping	EGP	CAC	SKA	Distribution
Alisson	2.20	-0.01	0.33	1.87	26.26
G. Donnarumma	2.52	-0.03	0.43	0.55	25.21
J. Oblak	2.42	-0.08	0.61	0.81	20.73
M. Maignan	2.51	-0.17	0.74	1.89	35.85
T. Courtois	1.91	-0.06	0.43	0.57	29.52

more than eight-five percent missing values was dropped, ensuring robustness of the dataset. In the remaining dataset, numerical values were imputed using the median where the absence of values was insignificant. As the final step, the data on age and market values was put through the statistical technique called as winsorization to limit the impact of outliers in the data. It was found that winsorization had an impact on the final results and normalizing the data for players too old or too young aided in standardizing the data.

1) Feature Engineering: As part of the feature engineering process, the dataset was divided into ten classes - goalkeepers (GK), left backs (LB), right backs (RB), center backs (CB), center defensive midfielders (CDM), center midfielders (CM), center attacking midfielders (CAM), left wingers (LW), right wingers (RW), and center forwards (CF) . For each class, customized metrics were developed using the preprocessed data. The metrics were as follows:

1. Goalkeepers (GK): The following metrics quantify the on-field performances for goalkeepers.

a. Shots Saved: The shot stopping ability of goalkeepers per 90 minutes was calculated using the formula:

$$\text{Shot Stopping} = \frac{\text{Saves}}{\text{90s}} \tag{1}$$

b. Expected Goals Prevention (EGP): This custom metric quantifies the goals prevention performance of goalkeepers per 90 minutes played.

$$\text{Expected Goals Prevention} = \frac{PSxG + GA}{90s} \tag{2}$$

c. Cross and Aerial Control (CAC): Shows how well the goalkeeper performs at catching or punching crosses coming into the 16 yard box per 90 minutes played.

$$\text{Crosses Stopped} = \frac{\text{Stp}}{\text{90s}} \tag{3}$$

d. Sweeper Keeper Activity (SKA): Quantifies the goalkeeper's ability to perform sweeping actions outside the 16 yard box per 90 minutes played.

$$\text{Sweeping Ability} = \frac{\text{\#OPA}}{\text{90s}} \tag{4}$$

e. Distribution Ability: Shows how capable the goalkeeper is distributing the ball with their feet. Calculated per 90 minutes played.

$$\text{Distribution Ability} = \frac{Cmp + KP + FinalThird}{90s} \tag{5}$$

2. Center Backs (CB): The following metrics quantify the on-field performances for center backs. Table II displays a subset of the center backs dataset.

a. Defensive Actions: Custom metric showing the center back's defensive contribution on the field per 90 minutes played.

$$\text{Defensive Contribution} = \frac{Tkl + Int + Blocks + Clr + Recov}{90s} \tag{6}$$

b. Aerial Ability: Quantifies the aerial solidity of the center back per 90 minutes played.

$$\text{Aerial Ability} = \frac{Won}{90s} \tag{7}$$

c. Passing Ability: Shows how well the center back passes the ball and progresses the ball up field per 90 minutes played

$$\text{Passing Ability} = \frac{Cmp + KP + PrgP}{90s} \tag{8}$$

d. Positioning and Defensive Awareness: Quantifies the positional awareness of the center back on field per 90 minutes played.

$$\text{Positioning and Defensive Awareness} = \frac{Blocks + Clr}{90s} \tag{9}$$

e. Disciplinary Record: Shows how disciplined the center back is across the games. Calculated per 90 minutes played.

$$\text{Discipline} = \frac{CrdY + CrdR + 2CrdY + Fouls}{90s} \tag{10}$$

TABLE II
CENTER BACKS

Name	Def. Actions	Aerial Duels	Passing	Def. Aware.	Discipline
Marquinhos	12.43	4.07	13.85	4.80	1.15
P. Cubarsí	8.11	2.72	11.35	3.37	0.70
P. Torres	8.02	2.32	9.25	3.60	0.40
V. van Dijk	11.55	2.45	8.02	5.68	0.51
W. Saliba	10.98	2.43	8.08	3.83	1.10

3. Full Backs (LB and RB): The following custom metrics have been used to quantify the performances of left backs and right backs. Table III displays a subset of the full backs dataset.

a. Defensive Duties: Quantifies how well the full back contributes defensively to the team, per 90 minutes played.

$$\text{Defensive Duties} = \frac{Def3rd, Int, Blocks, Clr, Recov}{90s} \tag{11}$$

b. Offensive Contributions: Quantifies how well the full back contributes offensively to the team, per 90 minutes played.

$$\text{Offensive Contributions} = \frac{PrgC + PrgP + KP + xA}{90s} \tag{12}$$

c. Final Third Play: This custom metric shows how well the full back makes themselves available to contribute in the final third, per 90 minutes played.

$$\text{Final Third Play} = \frac{Crs + SCA + CPA + PPA}{90s} \tag{13}$$

d. Possession Play: Quantifies how well the full back take cares of the ball on their feet, per 90 minutes played

$$\text{Possession Play} = \frac{Att3rd\_possession + TotDist}{90s} \tag{14}$$

e. Dribbling Accuracy: Measures how well the player dribbles through the opposition's press.

$$\text{Dribbling Accuracy} = \frac{Succ}{90s} \tag{15}$$

4. Central Defensive Midfielders (CDM): The following custom metrics have been used to quantify the performances of center defensive midfielders.

TABLE III
FULL BACKS

Name	Att. Contributions	Final Third	Possession	Dribbling
A. Balde	9.24	8.29	29.83	0.43
A. Robertson	10.69	9.68	32.86	0.04
D. Udogie	9.48	3.56	26.86	0.11
F. Dimarco	7.41	14.59	35.74	0.03
F. Mendy	4.61	0.71	21.42	0.03

TABLE IV
CENTER DEFENSIVE MIDFIELDERS

Name	Def. Work	Passing	Build-Up	Recoveries	Line Breaking
B. Guimarães	10.26	3.07	22.80	5.98	13.75
Casemiro	17.54	6.53	24.28	6.80	11.14
G. Xhaka	9.33	3.71	37.14	5.57	23.84
J. Neves	13.41	5.73	26.61	7.46	16.13
Y. Bissouma	12.95	7.32	21.00	6.63	10.24

TABLE V
CENTER MIDFIELDERS

Name	Passing	Dribbling	Def. Work	Chance Creation	Possession
D. Rice	11.26	4.13	9.56	4.17	54.04
F. Valverde	13.30	2.67	10.72	2.91	67.67
Pedri	18.64	4.20	12.43	4.92	75.93
Vitinha	2.19	4.66	8.66	3.16	15.71

TABLE VI
CENTRAL ATTACKING MIDFIELDERS

Name	Playmaking	BP	FTI	Goal Threat	FBE
D. Olmo	7.76	9.11	31.22	1.4	0.79
F. Wirtz	9.52	11.44	51.39	1.09	1.15
Isco	13.09	10.2	34.3	1.22	0.6
Ju. Bellingham	8.44	10.10	27.15	1.11	0.58
X. Simons	7.69	9.62	29.75	0.75	0.78

TABLE VII
WINGERS

Name	Dribbling	CAP	Goal Threat	FTI
Bukayo Saka	10.07	7.82	0.89	42.62
Ousmane Dembélé	12.51	5.56	3.07	48.25
Rodrygo	9.39	4.02	0.72	41.80
Raphinha	6.39	7.80	1.73	38.85

a. Defensive Contributions:

$$\text{Defensive Contributions} = \frac{Tkl + Int + Blocks + Clr + Recov}{90s} \tag{16}$$

b. Passing Ability:

$$\text{Passing Ability} = \frac{Cmp}{90s} \tag{17}$$

c. Build-Up Play:

$$\text{Build-Up Play} = \frac{xA + xAG + Ast + PrgDist}{90s} \tag{18}$$

d. Ball Recovery & Defensive Work:

$$\text{Ball Recovery \& Defensive Work} = \frac{Recov + Int}{90s} \tag{19}$$

e. Line Breaking Passes:

$$\text{Line Breaking Passes} = \frac{KP + PrgP + \text{1/3\_passing}}{90s} \tag{20}$$

5. Central Midfielders (CM): The following custom metrics have been used to quantify the performances of center midfielders.

a. Passing and Vision: Quantifies the how well the center midfielders passes the ball to contribute in offensive phases of the play, per 90 minutes played

$$\text{Passing} = \frac{PrgP + \text{1/3\_passing}}{90s} \tag{21}$$

b. Dribbling: Shows how well the center midfielder takes care of the ball and dribbles past opponents, per 90 minutes played.

$$\text{Ball Carrying} = \frac{Succ + PrgC + CPA}{90s} \tag{22}$$

c. Defensive Work: Explains the contribution of the center midfielder in defence, per 90 minutes played.

$$\text{Defensive Work} = \frac{Tkl + Int + Blocks + Clr + Recov}{90s} \tag{23}$$

d. Chance Creation: Quantifies the creative qualities of the center midfielder.

$$\text{Chance Creation} = \frac{SCA + xG + xA + xAG}{90s} \tag{24}$$

e. Possession Retention: Shows the ability of the center midfielder to retain the ball and not concede possession to the opposition, per 90 minutes played.

$$\text{Possession Retention} = \frac{Cmp + KP + \text{1/3\_passing} + Succ}{90s} \tag{25}$$

6. Central Attacking Midfielders (CAM): The following custom metrics have been used to quantify the performances of central attacking midfielders.

a. Creativity and Playmaking: Quantifies the creativity of the central attacking midfielder, per 90 mintutes played.

$$\text{Playmaking} = \frac{xA + SCA + \text{1/3\_passing}}{90s} \tag{26}$$

b. Ball Progression (BP): Numerifies the ability of the central attacking midfielder to move the ball up-field, per 90 minutes played.

$$\text{Ball Progression} = \frac{PrgP + PrgC}{90s} \tag{27}$$

c. Final Third Impact (FTI): Shows how much the central attacking midfielders impacts the game in the final (attacking) third, per 90 minutes played.

$$\text{Final Third Impact} = \frac{Att3rd\_possession + CPA + PPA}{90s} \tag{28}$$

d. Goal Threat: This metric shows the central attacking midfielder's ability to score goals, per 90 minutes played.

$$\text{Goal Threat} = \frac{xG + npxG + Gls}{90s} \tag{29}$$

e. Final Ball Efficiency (FBE): The ability of the central attacking midfielder to deliver the final ball, per 90 minutes played.

$$\text{Final Ball Efficiency} = \frac{xA + xAG + PPA}{90s} \tag{30}$$

7. Wingers (LW and RW): The following custom metrics have been used to quantify the performances of wingers. **a. Dribbling and Ball Carrying:** Shows how well the winger is at dribbling the ball past opponents, per 90 minutes played.

$$\text{Dribbling} = \frac{Succ + PrgC + CPA}{90s} \tag{31}$$

b. Crossing and Playmaking (CAP): This metric quantifies the playmaking and crossing ability of the winger, per 90 minutes played.

$$\text{Crosses and Playmaking} = \frac{xA + xAG + Crs}{90s} \tag{32}$$

c. Goal Threat (GT): This metric shows the winger's ability to score goals, per 90 minutes played.

$$\text{Goal Threat} = \frac{xG + npxG + Gls}{90s} \tag{33}$$

d. Final Third Involvement (FTI): Shows how much the winger impacts the game in the final (attacking) third, per 90 minutes played.

$$\text{Final Third Impact} = \frac{Att\_3rd\_possession + CPA + PPA}{90s} \tag{34}$$

8. Center Forwards (CF): The following custom metrics have been used to quantify the performances of center forwards.

TABLE VIII
CENTER FORWARDS

Name	GT	Ch.Conv.	LUP	Shooting Accuracy	PBP
E. Haaland	2.18	0.05	4.12	2.06	6.20
H. Kane	2.36	0.07	7.05	1.73	6.07
K. Mbappé	2.02	0.05	14.00	2.25	9.70
L. Martínez	1.35	0.03	7.54	1.32	5.56
R. Lewandowski	2.76	0.08	5.66	1.65	5.96

a. Goal Threat (GT): This metric shows the center forward's ability to score goals, per 90 minutes played.

$$\text{Goal Threat} = \frac{xG + npxG + Gls}{90s} \tag{35}$$

b. Chance Conversion : Quantifies how efficient is the center forward at converting chances, per 90 minutes played

$$\text{Chance Conversion} = \frac{G - PK + xG}{90s} \tag{36}$$

c. Link-up Play (LUP): Shows how well the center forward links up with the team through passing the ball, per 90 minutes played.

$$\text{Link-Up Play} = \frac{PrgR + xA + PPA}{90s} \tag{37}$$

d. Shooting Accuracy: Shows how accurately does the center forward shoot the ball on goal, per 90 minutes played.

$$\text{Shooting Accuracy} = \frac{SoT + Sh}{90s} \tag{38}$$

e. Penalty Box Presence (PBP): Quantifies the how present the center forward is in the penalty box, per 90 minutes played.

$$\text{Penalty Box Presence} = \frac{Att\_Pen}{90s} \tag{39}$$

D. Machine Learning Models

This section presents the proposed architecture of the machine learning model along with the parameters and rationale behind the selected models.

1) Dimensionality Reduction: To avoid overfitting, which is often introduced with high-dimensional data, PCA was performed after cleaning and feature engineering. This preprocessing reduced some dimensionality and highly correlated data, ensuring that the principal components identified by PCA were robust and effectively captured the variance.

- The Elbow Method (Scree Plot) explains the variance versus the number of components in a range (x, y).

To better understand the workings behind PCA, the mathematical formulas governing PCA are provided below.

Given a standardized dataset $\mathbf{X} \in \mathbb{R}^{n \times d}$, where n is the number of samples and d is the number of features, the covariance matrix is computed as:

$$\mathbf{C} = \frac{1}{n-1}\mathbf{X}^\top \mathbf{X} \tag{40}$$

Eigenvalue decomposition is then performed on the covariance matrix:

$$\mathbf{C}\mathbf{v}_i = \lambda_i \mathbf{v}_i \tag{41}$$

where λ_i is the eigenvalue corresponding to the eigenvector $\mathbf{v}_i$. The eigenvectors are sorted in descending order of their eigenvalues.

Let $\mathbf{V}_k \in \mathbb{R}^{d \times k}$ be the matrix of the top k eigenvectors. The data is then projected onto the new k-dimensional subspace as[14]:

$$\mathbf{Z} = \mathbf{X}\mathbf{V}_k \tag{42}$$

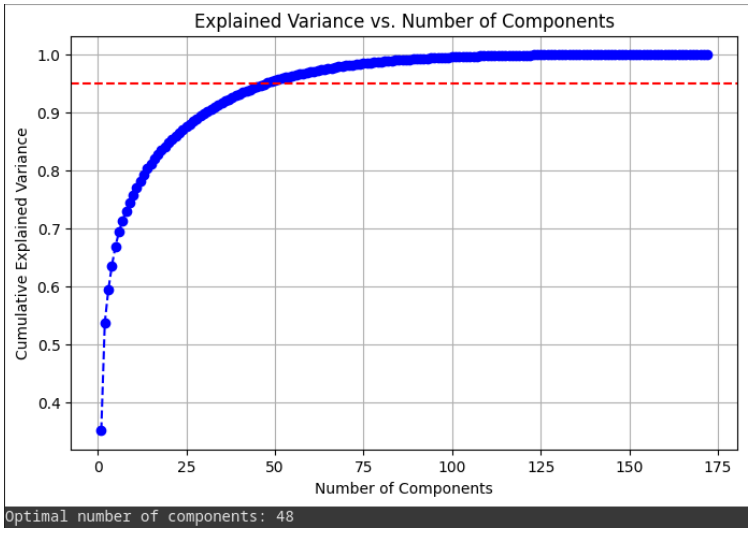

Fig. 2. PCA Scree Plot

Here, $\mathbf{Z} \in \mathbb{R}^{n \times k}$ is the transformed feature matrix in the reduced-dimensional space.

Following PCA, the transformed data are passed into an SEM, a stacked ensemble model, which performs the prediction of the market value of a player.

E. Stacked Ensemble Modeling

In recent times, multiple meta-heuristic learners and optimization algorithms have been doctored in the domain of football analytics [15]. This paper presents a novel approach among those to predict the market value of a football player, based on real data, compared to the FIFA values used by many other studies.

Stacking is an ensemble approach that uses a 2-level approach, level-0 base learners and a level-1 meta learner. The level-1 meta learner is an aggregator that receives the output from single-based learners.

a) Base Learners: The base learners used are selected to capture linear and non-linear relationships in the data. Both parametric and non-parametric models were selected.

Linear Learners

- **Ridge Regression**: Also called Tikhonov regularization, it is used in ill-posed problems, useful to mitigate the problem of multicollinearity in regression problems, caused by a high dimensionality. It is useful in this study due to large number of principal components(¿40). It employs L_2 regression to control multilinearity. Ridge regression minimizes the following loss function[16]:

$$\hat{\beta} = \arg\min_{\beta} \left\{ \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda \|\beta\|_2^2 \right\} \tag{43}$$

- **Lasso Regression**: Lasso Regression is a statistical operator that penalizes the model to prevent overfitting and enhance accuracy. It does so by shrinking some coefficients to zero, effectively excluding them from the model. It employs L_1 regression for automation feature selection.
  Lasso regression introduces an L_1 penalty to promote sparsity [17]:

$$\hat{\beta} = \arg\min_{\beta} \left\{ \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda \|\beta\|_1 \right\} \tag{44}$$

- **ElasticNet**: ElasticNet combines both, Lasso and Ridge regression, which improves its ability with regards with reconstruction. ElasticNet combines both L_1 and L_2 penalties[18]:

$$\hat{\beta} = \arg\min_{\beta} \left\{ \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2 \right\} \tag{45}$$

Tree-Based Learners

- **Random Forest Regressor**: Random forest regression is a machine learning technique that uses multiple decision trees to predict continuous values using the bootstrap averaging method on the outputs given by each tree. It is included for its ability in smoothing out the errors from other learners, in the overall stacking architecture, while effectively capturing non-linear dependencies.
- **XGBoost**: By leveraging a magnitude of decision trees, it focuses on creating a powerful predictive model by using an iterative process that focuses on minimizing errors. XGBoost is proved to be unbeatable for handling structured data, along with its ability to address class imbalance, which was crucial in our study, due to the limited data of goalkeepers compared to other positions.
- **LightGBM**: LightGBM is designed specifically for large-scaled data, and employs a leaf-wise growth strategy, opposed to the level-based growth strategy used by other tree based algorithms. This allows it to have deeper trees and better predictive ability.
- **Gradient Boosting Regressor**: Similar to XGBoost, it builds the model iteratively.
- **AdaBoost Regressor**: AdaBoost is often used in conjunction with other machine learning algorithms to improve performance. The output of multiple weak learners is combined into a weighted sum that represents the final output of the model [19]
- **Support Vector Regression (SVR)**: SVR tries to find a function that is able to accurately predict the continuous output value for any given output value. It uses both linear and non linear kernel.

1) Meta Learner: The meta learner, also known as the Level-1 learner, is the final model that receives the output from all the base models, and predicts the market value by using the stacking ensemble algorithm. This study uses the Random Forest Regressor as the meta learner. A common strategy to train a stacking model is to use a hold-out set, where the dataset is split into two parts - the first layer is trained using the first part of the dataset, and the second layer is given the second set of data. The output from the meta model is used to create a new dataset, which makes this new dataset a 3D dataset. This new dataset ensures that the model learns the target value, given the inputs from the first layer. The rationale behind using the Random Forest in both base learners and as a meta learner is to use its ability to generalize well, and smoothen the results, and helps capture the complex relationships. As a base learner, it introduces diversity, and as a meta learner, its insensitivity to multicollinearity makes it a strong candidate to be used in both the layers.

The mathematical representation for the proposed model is given as:

$$\hat{y} = f_{\text{meta}}\left(f_1(\mathbf{Z}), f_2(\mathbf{Z}), \ldots, f_m(\mathbf{Z})\right) \tag{46}$$

a) Where::

- $\hat{y}$: Final prediction (market value)
- $f_1, f_2, \ldots, f_m$: Base learners (e.g., Ridge, Lasso, RF, XGBoost, etc.)
- $\mathbf{Z} \in \mathbb{R}^{n \times k}$: PCA-reduced feature matrix with n samples and k components
- f_{meta}: Meta learner (Random Forest Regressor)

To evaluate the performance of the final SEM, multiple regression metrics were computed on both training and test sets. The rationale behind computing both training and testing sets was to check for overfitting. The difference between the training and test sets is a great metric to understand the state of the model. The higher the difference between these, the more the model overfits or underfits. The model achieved an R^2 score of 0.9464 on the training set and 0.9457 on the test set. The other metrics are shown in table 1.

Metric	Train	Test
R^2 Score	0.9464	0.9457
Mean Squared Error (MSE)	9.73×10^{13}	9.69×10^{13}
Root Mean Squared Error (RMSE)	3,119,656.30	3,112,937.64
Mean Absolute Error (MAE)	2,199,469.47	2,259,460.70
Cross-Validation R^2 (Mean $\pm$ Std)		0.9383 ± 0.0029

TABLE IX
EVALUATION METRICS FOR THE STACKING ENSEMBLE MODEL

The cross validation R^2 score being close to the R^2 score of train and test sets proves that the model does the overfit, and the combination of chosen base learners and meta learners is a great fit for the type of data this study deals with.

IV. Conclusion and Future Work

By efficiently automating data extraction and processing of advanced player statistics from robust and vast data sources such as Fbref and Transfermarkt, the proposed system uses the extracted data to not only automate the manual and subjective process of scouting football players, but also drive the decision making process of coaches and managers. The system leverages complex machine learning models for predicting the market value predictions, along with clustering algorithms to identify players that are similar both, tactically and stylistically. The system includes role-specific metrics and performs position-aware analysis. Additionally, the usage of detailed visualizations such as radar charts and other performance graphs, makes the data analysis interpretable and actionable for coaches.

The system provides a robust framework to find replacements for football players, there are several aspects that exist for future enhancements. In the future, we would like to work on integrating match-by-match real time performance data which would offer dynamic tracking of players on the field. Additionally, building an all-inclusive analysis hub by including video analytics and sentiment analysis of players from media sources. Finally, expanding our databases to retired players, women's leagues and youth leagues would open underexplored doors.

References

[1] FBref, "Fbref.com - football stats and history," https://fbref.com, 2024, accessed: 2024-04-11.

[2] SoFIFA, "Sofifa - football player ratings, attributes and career data," https://sofifa.com/, 2024, accessed: 2024-04-11.

[3] O. Müller, A. Simons, and M. Weinmann, "Beyond crowd judgments: Data-driven estimation of market value in association football," *European Journal of Operational Research*, vol. 263, no. 2, pp. 611–624, 2017.

[4] Q. Yi, M.-A. Gómez, H. Liu, B. Gao, F. Wunderlich, and D. Memmert, "Situational and positional effects on the technical variation of players in the uefa champions league," *Frontiers in Psychology*, vol. 11, p. 1201, 2020.

[5] F. Carmichael and D. Thomas, "Bargaining in the transfer market: Theory and evidence." *Applied Economics*, vol. 25, pp. 1467–76, 12 1993.

[6] M. A. Al-Asadi and S. Tasdemır, "Predict the value of football players using fifa video game data and machine learning techniques," *IEEE Access*, vol. 10, pp. 22 631–22 645, 2022.

[7] Transfermarkt, "Transfermarkt - football transfers and market values," https://www.transfermarkt.com, 2024, accessed: 2024-04-11.

[8] R. Stanojevic and L. Gyarmati, "Towards data-driven football player assessment," in *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, 2016, pp. 167–172.

[9] S. Dobson, B. Gerrard, and S. Howe, "The determination of transfer fees in english nonleague football," *Applied Economics*, vol. 32, no. 9, pp. 1145–1152, 2000.

[10] C. A. Depken and T. Globan, "Football transfer fee premiums and europe's big five," *Southern Economic Journal*, vol. 87, no. 3, pp. 889–908, 2021.
[11] A. T. Yiğit, B. Samak, and T. Kaya, "Football player value assessment using machine learning techniques," in *Intelligent and Fuzzy Techniques in Big Data Analytics and Decision Making*, 2020, pp. 289–297.
[12] I. Behravan and S. M. Razavi, "A novel machine learning method for estimating football players' value in the transfer market," *Soft Computing*, vol. 25, no. 3, pp. 2499–2511, 2021.
[13] I. G. McHale and B. Holmes, "Estimating transfer fees of professional footballers using advanced performance metrics and machine learning," *European Journal of Operational Research*, vol. 306, no. 1, pp. 389–399, 2023.
[14] I. Jolliffe, "Principal component analysis springer verlag," 01 2002.
[15] S. Buyrukoğlu and S. Savaş, "Stacked-based ensemble machine learning model for positioning footballer," *Arabian Journal for Science and Engineering*, vol. 48, no. 3, pp. 1371–1383, 2023.
[16] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970. [Online]. Available: http://www.jstor.org/stable/1267351
[17] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996. [Online]. Available: http://www.jstor.org/stable/2346178
[18] H. Zou and T. Hastie, "Zou h, hastie t. regularization and variable selection via the elastic net. j r statist soc b. 2005;67(2):301-20," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, pp. 301 – 320, 04 2005.
[19] S. Rosset, H. Zou, and T. Hastie, "Multi-class adaboost," *Statistics and its interface*, vol. 2, 02 2006.