Project Title:
Graph It Like Beckham - Exploring Football Through Knowledge Graphs

Name: Mahmoud Abdussalem SAKKA, 11803058

Representations

(LO1) Understand and apply **Knowledge Graph Embeddings**	☐ I showed **basic** proficiency ☒ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

(LO2) Understand and apply **logical knowledge** in KGs	☒ I showed **basic** proficiency ☐ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

(LO3) Understand and apply **Graph Neural Networks**	☐ I showed **basic** proficiency ☒ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

(LO4) Compare different Knowledge Graph **data models** from the database, semantic web, machine learning and data science communities.	☐ I showed **basic** proficiency ☐ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

Systems

(LO5) Design and implement **architectures** of a Knowledge Graph	☒ I showed **basic** proficiency ☐ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

(LO6) Describe and apply **scalable reasoning** methods in Knowledge Graphs	☐ I showed **basic** proficiency ☐ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

(LO7) Apply a system to **create** a Knowledge Graph	☐ I showed **basic** proficiency ☒ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

(LO8) Apply a system to **evolve** a Knowledge Graph	☐ I showed **basic** proficiency ☐ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

Applications

(LO9) Describe and design **real-world applications** of Knowledge Graphs	☐ I showed **basic** proficiency ☒ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

(LO10) Describe **financial Knowledge Graph** applications	☒ I showed **basic** proficiency ☐ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

(LO11) Apply a system to provide **services** through a Knowledge Graph	☒ I showed **basic** proficiency ☐ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

(LO12) Describe the **connections** between Knowledge Graphs (KGs), Machine Learning (ML) and Artificial Intelligence (AI)	☒ I showed **basic** proficiency ☐ **exceeded** basic proficiency
<1-2 Sentences how you addressed this LO in your portfolio>	<Page(s) in the portfolio where this is discussed>

Additional Information

HAS NO EFFECT ON MARKING!

(please fill it out honestly, even if it is less than what is suggested in the ECTS breakdown – you are not judged on time spent!)

How many hours did you spend on your **mini-project**? (the ECTS breakdown suggests **40** hours for this)	55-60 hours

* please exclude any hours you spent on parts reused from other courses

How many hours did you spend on your **portfolio document preparation (this PDF)**? (the ECTS breakdown suggests **15** hours for this)	20 hours

* please exclude any hours you spent on parts reused from other courses

Please indicate if you have **reused** parts of the **mini-project** from other courses	☐ I reused some parts: 0% of the mini-project
-	

Please indicate if you have **reused** parts of the **portfolio document** from other courses	☐ I reused some parts: 0% of this document
-	

Declaration

I have marked all parts generated by Generative AI (e.g., ChatGPT) and given any prompt I used either in a footnote or in an appendix making clear which parts are generated by which prompts or similar.	☒ I confirm this

Here follows your portfolio (report), structured as you wish!

A possible structure (but by no means a suggestion, use whatever you like) is:

1. Introduction
2. Background
3. Method
4. Results
5. Conclusion

IN PARAGRAPHS WHERE YOU EXPLICITLY DISCUSS A PARTICULAR LEARNING OUTCOME (ESPECIALLY THOSE MENTIONED IN YOUR COVER PAGES), PLEASE USE REFERENCES IN THE FORM OF (LO1) TO INDICATE THAT.

An example:

"For creating my Knowledge Graph (LO7) I used a combination of logical rules encoding the domain knowledge of XXX and a Knowledge Graph embedding that infers edges between entities not explicitly given in the source data. […]

Let me discuss the Knowledge Graph embedding (LO1) used for that in more detail […]"

You do not have do that everywhere (it is not a competition on how often you mention a particular learning outcome), but especially at places that you explicitly reference in your cover pages, it helps you (and us) to see more easily which point you want to emphasize!

Introduction

According to FIFA, the global authority overseeing association football and its variations such as beach soccer and futsal, there are approximately five billion football fans around the world[1]. Of these, an estimated 1.5 billion viewers tuned in to watch the final match of the 2022 FIFA World Cup[2], making it one of the most watched events globally and highlighting football's position as the world's most popular sport. Consequently, FIFA and its various adjacent national governing bodies are also very motivated to serve this lively market by expanding both the frequency and variety of competitions. These events are subsequently auctioned off to various (often subscription-based) broadcasters, such as Sky or DAZN, who are willing to pay increasingly higher sums for the broadcasting rights of national and international leagues, cup competitions, and, in some cases, even clubs. Additionally, rising sponsorship deals, ticket prices and the instrumentalization of football for sports washing by various states and politicians, are contributing to the growing influx of money in the sport, which is then invested in player transfers, salaries, and stadium infrastructure. Ultimately, football has evolved into a multi-billion-dollar industry that involves not only governing bodies, clubs, corporations and fans but also entire nations, as exemplified by PSG's ownership structure or the political maneuvering and scandals behind World Cup hosting decisions. These developments also come with drawbacks, such as the widely debated risks associated with the heightened workload, resulting from the increased number of competitions and games particularly for elite players. One of the most pressing concerns is the increased likelihood of injuries, which can have far-reaching consequences not only affecting players' careers but also impacting team performance and the overall quality of football competitions.

The initial project-idea described in the one pager aimed to develop a GNN-model using player interactions, game schedules and historical injury data in order to predict the likelihood of injuries and injury types. However, this approach had to be adapted, as the search for consistent and usable data revealed that the search, compilation and cleansing of the data would go way beyond the scope and intended timeframe of this project. To implement the Graph Neural Network described in the one-pager a dataset containing player data from the FIFA video game series (later rebranded as EA Sports FC) produced by EA was used. Since the dataset does not include information on injuries, but does provide detailed player attributes such as name, age, position, and realistic ratings ranging from 0 to 99 for defending, dribbling, attacking, overall rating, and potential among many others, the approach was adapted. Instead of injury prediction, the focus shifted to forecasting a player's potential based on these attributes and their development across different iterations of the game. To highlight other applications of such a football knowledge graph, Knowledge Graph Embeddings (KGEs) were generated by training a TransE model on the knowledge graph. These embeddings were then used for a weighted similarity calculation, by combining the cosine similarity between players' embeddings with additional player characteristics. This similarity calculation allows for the use of the graph to find players with similar profiles, facilitating potential use cases such as scouting recommendations relevant for decisions regarding squad building and player transfers.

[1] https://publications.fifa.com/en/vision-report-2021/the-football-landscape/ (02.03.2025, 18:50)
[2] https://inside.fifa.com/tournament-organisation/world-cup-2022-in-numbers (02.03.2025, 18:50)

The FIFA/EA Sports FC Franchise

The FIFA video game series, developed and published by EA Sports, stands as one of the largest franchises in the industry. Known for its official licensing agreements with FIFA and several football leagues, the game offers a highly realistic simulation of real-life football. In 2022, EA Sports and FIFA ended their licensing agreement, making FIFA 23 the final game released under the FIFA name. Following this change, EA introduced the EA Sports FC series. Though only the branding changed, the rating system and gameplay remained largely consistent with previous editions.

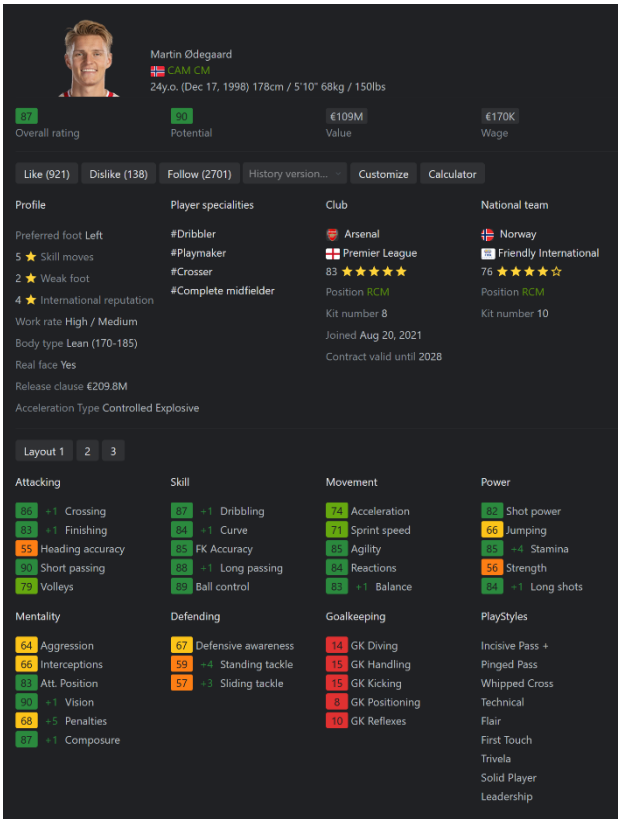

Figure 1 EA FC 24 ratings for Arsenal captain Martin Odegaard (source: sofifa; 03.03.2025, 16:12)

In the game, player ratings are determined through a system that combines input from EA Sports employees, external contributors, and volunteer FIFA Scouts. These experts continually update a detailed database of specific attributes for each player. Numeric attributes are rated on a scale from 0 to 99, while factors like skill moves, weak foot ability, and international reputation are rated on a 0 to 5-star scale. To compute the overall rating, ratings for Pace, Shooting, Passing, Dribbling, Defending, and Physical, among others, are weighted using positional coefficients based on a player's on-field position. Additionally, a player's international reputation is factored in to ensure the final rating accurately reflects their real-world counterpart's performance.[3] In addition to the stats mentioned, qualitative information such as player specialty, body type, and work rate is also provided. Greatly relevant for the in-game career mode, a potential rating is also included, which estimates how the player might develop, along with current player value and salary estimations. Figure 1 provides an example of the rating system in action, showing a section of the ratings for Arsenal FC captain Martin Ødegaard in EA Sports FC 24.

Data Preprocessing

Initial Data Cleaning

As described in the introduction, the selected dataset contains information on players and teams from FIFA 15 up to EA Sports FC 24. The data was scraped from Sofifa.com, a publicly accessible platform that compiles player ratings and detailed information about the teams featured in the video game. This analysis focuses exclusively on male players and teams. The database consists of two CSV files: the player file, containing 180,021 observations across 109 variables, and the team file, comprising 6,947 observations with 54 columns. Spanning from the 2015 to the 2024 iteration of the game, the dataset represents players and teams' multiple times, with each instance reflecting a different stage of their career and development. The dataset's columns include unique identifiers, names, and detailed ratings, as outlined in the introduction. Additionally, it features qualitative

[3] https://www.goal.com/en/news/fifa-player-ratings-explained-how-are-the-card-number--stats-decided/1hszd2fgr7wgf1n2b2yjdpgynu (03.03.2025, 15:37)

attributes such as work rate and preferred foot, as well as financial and contractual details, including transfer market values, salary estimates, and measures of prestige and reputation.

Several variables, such as jersey numbers and release clauses, were removed either due to their limited analytical value or, in the case of release clauses, because they contained unreliable information that did not reflect real-world data. The variable indicating whether a player plays for a national team was also excluded, as the game features only a limited selection of national teams. Consequently, many real-life national team players would be misclassified as non-national team players, introducing inconsistencies. Certain categorical attributes were transformed for better usability. For instance, the club_position column, originally containing specific role labels such as "RES" (reserve) and "SUB" (substitute), was recoded into a numerical indicator distinguishing between starters, substitutes, and reserves. However, this approach has limitations: players who are injured or unavailable at the time of the game's release may not appear as starters in the dataset, even if they are regular starters in real life. A more refined metric, such as the ratio of starting eleven appearances to total games or minutes played per match exceeding a certain threshold, would provide a more accurate measure of a player's role within the team. Similarly, the club_loaned_from variable, which indicates whether a player was on loan, was converted into a binary on_loan variable. Originally, the player_positions column contained multiple position labels for players capable of playing in different roles. To simplify analysis, this variable was aggregated into broader positional categories. The first-listed position was assumed to be a player's primary role. Goalkeepers (GK) remained distinct, while strikers (ST, CF) were grouped under ATT (attack). Wide players, including wingers and wingbacks (RW, LW, RWB, LWB), were categorized separately as WING due to their unique tactical function, particularly in systems used by coaches such as Antonio Conte or Ruben Amorim, where wingbacks play a more offensive role. Midfielders, spanning central and defensive roles (CM, CAM, CDM, LM, RM), were unified under MID, while defenders (CB, RB, LB) were categorized as DEF.

In some cases, teams from one country compete in leagues based in another, requiring adjustments to ensure consistency. Instead of assigning clubs to their country of origin, the dataset was modified so that a team's nationality aligned with the league in which it competed. For example, AS Monaco, despite being a club from Monaco, competes in France's Ligue 1 and was thus assigned "France" as its nationality. Similarly, Welsh teams playing in England's Championship and League Two were reassigned to England, while Canadian teams in Major League Soccer were attributed to the United States. Beyond these adjustments, certain leagues shared identical names across different countries, creating ambiguity. One notable example was the "Premier League," which referred to both the English and Russian top divisions. To resolve this, league names were adjusted for clarity, with the Russian Premier League explicitly renamed as "Russian Premier League" to distinguish it from its English counterpart. Finally, leagues with an insufficient number of teams were excluded from the analysis, with the threshold set at eight teams. For instance, players and teams from the Croatian league, which does not meet this criterion, were omitted.

Missing Value & NA Handling

There were various types of missing values in the dataset. For example, players listed as free agents in the FIFA game lacked both club and league affiliation. After examining individual examples, it quickly became apparent that many players who are listed as free agents in the game play for a club in real life that is not included in the game. To address this, all players without a recorded league or club were filtered out, ensuring that only those with a clear team affiliation remained in the dataset. Beyond filtering out free agents, missing values were imputed. One example for variable requiring imputation was value_eur, which represents a player's estimated market value. To facilitate the

applied imputation process, players were grouped into brackets based on their age and overall rating. Missing values were then replaced with the mean value of players in the same position category, age group, overall rating range, and league level. In rare cases where no suitable comparison group was available, the lowest recorded value within the same category was used as a fallback to prevent introducing artificial inflation. A similar imputation strategy was applied to team-level attributes such as transfer budgets and club valuations. First, domestic_prestige and international_prestige for clubs in FIFA versions 15 and 16 were inconsistent with later iterations, so their scores were imputed using the average from FIFA 17 onwards for the same clubs, leveraging the relative stability of these rankings over time. For financial metrics like transfer_budget_eur and club_worth_eur, a cascading imputation approach was implemented. Initially, missing values were replaced with the mean of clubs sharing similar international and domestic prestige, league level, and overall club rating. If gaps remained, broader groupings with fewer constraints were used to ensure all missing values were filled with the most contextually appropriate estimates.

NA values were also examined and imputed. In the case of the club_joined_date column, which was NA for players who were on loan, a simplified assumption was applied considering that most loans are finalized in the summer. If a player was on loan and lacked a club_joined_date, it was set to August 1st of the year before the corresponding FIFA version, given that the game is released in September of the preceding year (e.g., FIFA 23 was released in September 2022). Another issue involved goalkeeping attributes. Since outfield players do not have goalkeeping statistics, NA values in goalkeeping-specific attributes such as goalkeeping_speed were not true NA values but rather structurally absent data. To correct this, all NA values for non-goalkeepers were set to zero. Similarly, outfield player attributes such as pace, shooting, passing, dribbling, defending, and physic were set to zero for goalkeepers. Finally, the mentality_composure attribute contained NAs. To address this, an imputation strategy similar to the one used for value_eur was applied: missing values were replaced with the mean composure score of players within the same position category, age group, and overall rating range. If any gaps remained, they were filled using the minimum available composure score within the dataset.

Creating the League Data

The original dataset did not contain explicit league-level data, but using the available team information, a derived league dataset was constructed. For this purpose, team data was aggregated at the league level. For attributes such as team ratings, the mean values across all teams within each league were computed. Furthermore, additional league-specific attributes were derived. The overall league rating, as well as its attack, midfield, and defense ratings, were rounded to maintain consistency with the player and team datasets. Regarding league prestige, a normalized prestige score was introduced based on financial indicators. The international prestige of a league was computed using a weighted formula incorporating normalized values of international_prestige, transfer_budget_eur, and club_worth_eur. This approach accounted for both financial power and reputation, with club-worth receiving the highest weight (0,6), as it is often the most stable indicator of a league's standing overtime.

Subsampling

Due to computational constraints, a subsampling strategy was implemented to reduce the dataset size while preserving its representativeness. To achieve this, the dataset was restricted to the four most recent FIFA versions (21, 22, 23, and 24). Within this subset, players appearing in multiple FIFA editions were identified, and only those with the highest number of occurrences across these versions were retained. To maintain diversity and balance across different player characteristics, a stratified sampling approach was applied. Players were grouped based on their overall rating range,

age group, and positional category. From each unique combination, twenty players were randomly selected to ensure that all player profiles were adequately represented. This resulted in a total of 2,168 unique players and 8,672 player observations, representing 2,163 team observations across 175 league observations.

Knowledge Graph Creation and Population

To construct the knowledge graph (KG), an ontology was designed in Protege to formalize the relevant relationships. The ontology defined in Figure 2 consists of three main classes: Player, Club, and League. Players are linked to clubs through the plays_for relationship, and both players and clubs are associated with leagues using the competes_in property, enforcing a one-to-one constraint to maintain consistency with real-world football structures. Clubs also feature a rival_with relationship, modeled as a symmetric property. Each entity is enriched with data properties that reflect the relevant characteristics from the FIFA dataset, such as a player's age, position, overall rating, a club's international prestige and transfer budget, and a league's level and nationality. The ontology serves as the structural foundation of the knowledge graph, ensuring logical consistency and enabling graph-based reasoning over the encoded football data (LO5, LO2).

Once the ontology was defined, the next step was to populate the knowledge graph with the preprocessed data described in the previous chapter. This was implemented in Python using rdflib, which facilitated the transformation of structured data into RDF triples that adhere to the relationships and logical constraints defined in the ontology, thus converting the dataset into a graph. (LO2, LO7) Each entity was instantiated as an RDF node, with attributes such as overall rating, potential and skill attributes mapped to their respective data properties. The object properties (plays_for, competes_in, and rival_with) were assigned to establish relationships between the entities. Finally, the populated knowledge graph was validated by verifying that each player, club, and league had the correct relationships, and checking for orphaned nodes or inconsistencies. The resulting populated knowledge graph was saved in the form of RDF triples as a Turtle (ttl) file, which serves as the basis for the Graph Neural Network and Knowledge Graph Embeddings discussed later.

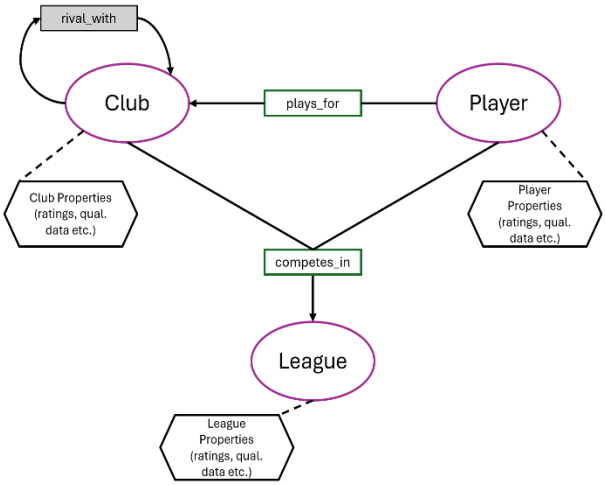

Figure 2 Knowledge Graph Structure

Graph Neural Network

The goal of implementing a Graph Neural Network (GNN) in this project is to leverage the structured relationships between players, clubs, and leagues to predict a player's potential. A player's potential in the game refers to their projected future ability, indicating how much their overall rating can improve compared to their current assigned rating when playing the game's career mode. Similarly, in real life, assessing a player's potential is crucial for scouting, transfer decisions, and player development strategies, as teams seek to identify young talents who can grow into key assets over time. (LO9)

Traditional machine learning models operate on tabular data and may fail to capture patterns arising from the interconnected nature of certain domains, such as the footballing world, where a player's development is influenced not just by their attributes but also by their environment, namely the

clubs and leagues they play for. A young player developing in a slightly weaker league and gradually moving up to better competitions may have a better opportunity to hone their skills, whereas a player who spends their entire career in a weaker team might never fully realize their potential. Constructing a Knowledge Graph (KG) and applying a GNN incorporates these complex dependencies into the learning process, enhancing the predictive power of the model beyond what is achievable with standard tabular methods. The implementation of the GNN closely follows the principles of a traditional machine learning pipeline. The first step involves constructing the KG, as described in the Knowledge Graph Creation and Population chapter. The KG is then used as input for the GNN, which is defined by selecting an appropriate architecture, i.e., the type of GNN (GCN, GAT, etc.), the number of layers, and the activation function. Once the model is initialized, it can be further refined through hyperparameter optimization, improved train-test split strategies, and architectural adjustments. This rough overview illustrates how Knowledge Graphs serve as a structured foundation for Machine Learning, enhancing the extraction of meaningful patterns that can be instrumentalized for AI-driven models and real-world applications. (LO12)

For this project, the GNN was implemented using PyTorch Geometric (PyG). The architecture follows a multi-layer Graph Convolutional Network (GCN) approach, which enables message passing between connected nodes, allowing the model to learn representations that incorporate both node-specific features and information propagated from neighboring entities. Before defining the structure and training the GNN, the KG was converted into a format suitable for PyG. Nodes (players, clubs, leagues) were mapped to numerical indices, and categorical features such as position category, preferred foot, and work rate were encoded using LabelEncoder from scikit-learn. Numerical attributes were standardized using StandardScaler, and the edges in the KG were extracted to form the adjacency matrix. (LO3)

The model consists of three GCNConv layers, applying convolution operations to transform the raw feature vectors into an initial set of node embeddings, which are then further refined by incorporating contextual information from neighboring nodes. Finally, a fully connected layer predicts player potential as a continuous value. Each GCN layer is followed by a ReLU activation function, introducing non-linearity. The model is trained using the Adam optimizer, with a mean squared error (MSE) loss function, as the objective is to perform regression, given that potential ratings are in the interval 0 to 99. The learning rate was set to 0.001, and a weight decay of 1e-3 was used to prevent overfitting. A step-based learning rate scheduler was applied, reducing the learning rate by a factor of 0.8 every 1000 steps to facilitate stable convergence. The model was trained for a maximum of 15,000 epochs, with an early stopping patience of 500 epochs. The hidden layer size was set to 32. (LO3)

Random vs. Player-Stratified Split

To evaluate the model, two different data-splitting strategies were implemented: a random split and a player-stratified split. In the random split, nodes were assigned to training, validation, and test sets without considering their underlying relationships. While this is a conventional approach, it poses a risk of data leakage since a player appearing in multiple FIFA versions could end up in different sets, leading to unreliable metrics. To mitigate this, a player-stratified split was introduced. Here, players were grouped based on their unique ID, ensuring that all instances of a given player were assigned to the same dataset split. This approach more accurately reflects real-world scenarios where predictions must be made for players who have never been seen in training, thereby providing a more realistic estimate of the model's generalization ability.

Results

The performance of the GNN was evaluated using both random split and player-stratified split strategies. The key metrics used for comparison included Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2) score.

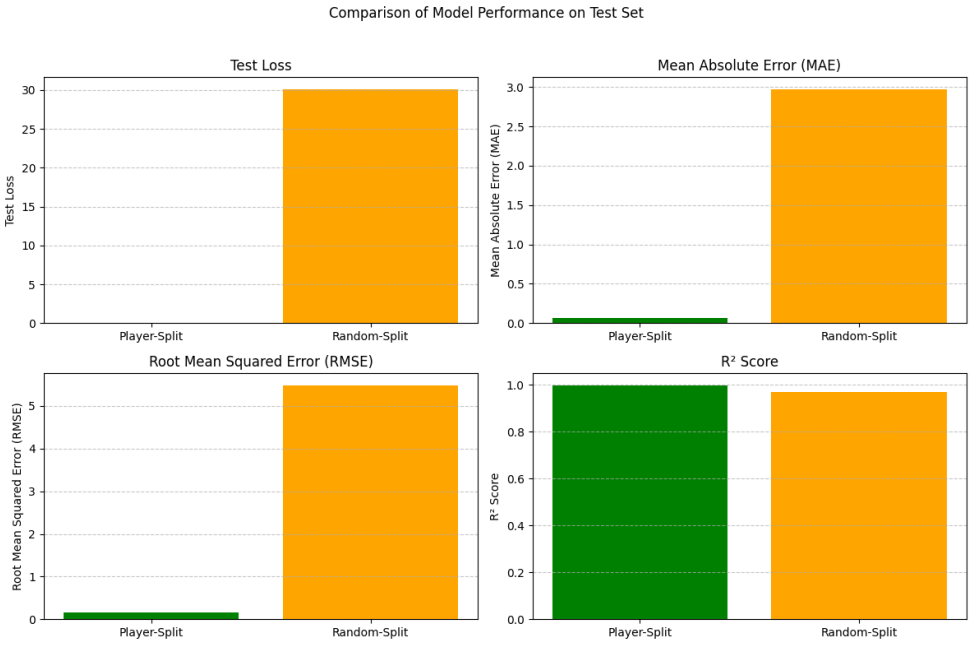

Figure 3 Model Comparison for the Random and Player-Stratified Split

The results highlight a clear difference between the random split and the player-stratified split. The random split achieved a final test RMSE of 5.49, test MAE of 2,98 and R^2 of 0.97, showing an okay performance but likely benefiting from data leakage. Additionally, while the training loss decreased rapidly, the validation loss declined more slowly and remained high. The significant gap between the two suggests that the model was overfitting, learning patterns specific to the training data rather than generalizing effectively. In contrast, the player-stratified split performed significantly better in terms of generalization, achieving a final test RMSE of 0.15, a test MAE of 0.07 and R^2 of 0.9996. Notably, the training and validation loss followed a similar trajectory throughout the training process, indicating that the model was learning meaningful patterns rather than overfitting the training data. This suggests that by ensuring no overlap between the training, validation, and test sets, the model was able to generalize well to unseen players. Early stopping was triggered after 13,209 epochs.

Split	RMSE	MAE	R^2	Num of. Epochs
Random	5.4895	2.9755	0.9689	11049
Stratified	0.1519	0.0655	0.9996	13209

Table 1 GNN Results: Random vs. Player-Stratified Split

These results highlight the importance of choosing an appropriate train-test split strategy to ensure a realistic evaluation. Given that this study serves as a proof of concept and is subject to computational constraints, hyperparameter tuning was intentionally omitted. However, further refinements, including the integration of additional contextual data such as team performance and tactical roles, could enhance the model's performance. Additionally, experimenting with alternative GNN

architectures, such as GraphSAGE or Graph Attention Networks (GAT), may lead to further improvements.

Knowledge Graph Embeddings for Player Similarity Analysis

In addition to implementing a GNN for predicting player potential, Knowledge Graph Embeddings (KGEs) were used to compute a weighted player similarity score. The goal was to leverage structured relationships within the KG to identify similar players, which is an essential task in scouting and transfer decisions. In professional football, finding a like-for-like replacement for a departing player is crucial, particularly for clubs such as FC Barcelona or FC Bayern, which follow defined playing philosophy or specific tactical system. For instance, if a club sells a key midfielder, they must identify a replacement with a comparable skill set who seamlessly integrates into their system. To achieve this, a cosine similarity on KGE vectors, combined with weighted attribute-based similarity scores, was implemented to identify players with comparable profiles, even across different leagues and levels of competition. The KGE model was implemented using PyKEEN, with TriplesFactory processing the subject-predicate-object structure of the graph, converting it into a format suitable for training embedding models. Based on lessons learned from the GNN chapter, a player-stratified split was applied, dividing the dataset into training (60%), validation (20%), and test (20%) sets, ensuring that all observations of a given player remained within the same split, thus preventing data leakage. For the embedding model, TransE was selected and trained for 50 epochs using the Adam optimizer with a learning rate of 0.005 and a batch size of 512. The embeddings were then evaluated using standard link prediction metrics, including Hits@K, Mean Rank, and Mean Reciprocal Rank (MRR). Unfortunately, the results for TransE were suboptimal, with low performance across all evaluation metrics. The Hits@1, Hits@3, and Hits@10 scores indicated that the model struggled to correctly predict missing links within the graph. The MRR and Mean Rank values further confirmed that the learned embeddings lacked strong predictive power. Experiments were conducted with different embedding models, including ComplEx, RotatE, and DistMult, as well as adjustments to hyperparameters and split strategies. However, none of these variations led to notable improvements in performance, suggesting that either the graph structure lacked sufficient relational depth or that the player-club-league relationships alone were insufficient to generate meaningful embeddings. (LO1)

Nonetheless, the KGEs were used to calculate the cosine similarity of players, which was then incorporated into a weighted sum based on this similarity and key player attributes. This hybrid approach ensures that both statistical similarity, captured by embeddings, and football-relevant factors, such as position, physicality, and mentality, are taken into account. First, the embedding vector of a queried player is retrieved from the trained TransE model. Using cosine similarity, the similarity between this vector and the embeddings of all other players in the dataset is computed. To refine the results and ensure footballing relevance, a weighted similarity score containing additional player attributes is applied. Positional similarity accounts for 25% of the score, while physical similarity contributes 15% and mentality similarity makes up 10%. Work rate similarity accounts for another 10%, prioritizing players with similar work rates. Finally, the cosine similarity is weighted most heavily at 40%. After calculating the weighted similarity score, the function ranks the top candidates and returns the k most similar players. To ensure meaningful comparisons, players from different FIFA versions of the same individual are excluded, preventing multiple versions of the same player from being recommended. The variables used in this approach were chosen because they provide a good overall representation of a player. However, in a real-world application, more precise and comprehensive metrics could be used, such as detailed performance data, tactical roles, or position-specific statistics. Similarly, the weightings of the individual attributes can be understood as

hyperparameters and could be flexibly adjusted and optimized to meet the specific needs of a club or scouting team.

Calculating the weighted similarity for Martin Ødegaard (Arsenal FC, FIFA 23) results in Harvey Barnes (Leicester City, FIFA 23) as the most similar player. This makes sense, as both players are technically gifted, have comparable ratings, and share a similar work rate, as shown in Table 2. Other recommended players, such as Ismaila Sarr and Bowen, further reinforce the function's ability to identify reasonable alternatives. Despite the weak performance of the embedding model itself, the weighted similarity calculation still produced reasonable results.

Name	Age	Position	Physic	Mentality	Work Rate	Overall	Cosine Sim	Weighted Sim
Harvey Lewis Barnes	24	MID	64	78	High/Medium	80	0,9966	0.9477
Ismaïla Sarr	22	MID	66	78	High/Medium	78	0,9952	0.9451
Ricard Puig Martí	21	MID	63	77	High/Medium	76	0.9893	0.9447
Jarrod Bowen	25	MID	66	77	High/Medium	80	0,9964	0.9446
Nanitamo Ikoné	22	MID	60	79	High/Medium	79	0,9964	0.9445

Table 2 Top 5 Results for Martin Ødegaard
(Age: 23, Position: MID, Physical: 63, Mentality: 78, Work Rate: High/Medium, Overall: 84)

Restrictions, Relevance and Potential Applications

While the implemented Knowledge Graph, GNN, and KGE illustrate the potential of a football KG, it is important to emphasize that this project serves as a proof of concept. One key limitation lies in the choice of dataset. The FIFA dataset provides a simplified representation of player abilities, but it lacks essential real-world factors such as match performance data, injury history, and tactical adaptability, among others, all of which are crucial for professional scouting. Incorporating detailed match event data such as pass networks, pressing tendencies, and Expected Goals (xG) metrics, would further improve the KG's ability to represent a player's ability. As a result, while the KG effectively models player-club-league relationships, it does not fully capture more complex interactions such as team-specific playing styles or on-field dynamics.

Of course, the dataset influences the structural design of the KG. The current implementation primarily revolves around player-club-league relationships, which remain relatively shallow in terms of football-specific context. Expanding the KG to include more competitions, coaches, tactical systems, advisors, national teams, and contract details would provide a richer foundation for analysis. Additional object properties could enhance the graph's relational structure, allowing for a more nuanced understanding of player development and career trajectories. (LO5, LO7)

The KGE approach, while promising, also has its shortcomings. The model struggled with link prediction, indicating that the graph's relational depth was insufficient to generate meaningful embeddings. This suggests that either the number of relationship types was too limited, or that the data used for training lacked the necessary variety to form robust representations. The weighted similarity approach used in player comparisons was also determined heuristically, meaning that while the results were reasonable, they were not optimized. A more advanced approach could involve machine learning techniques that adjust similarity weightings based on tactical fit, or performance data, leading to more accurate scouting recommendations. (LO1)

Relevance of Football Knowledge Graphs

Despite the mentioned limitations, the project demonstrates that in the context of football analytics, a Knowledge Graph (KG) can effectively integrate player, team, and match data, enabling advanced statistical analysis and AI-driven decision-making. One of its most valuable applications is in scouting and recruitment, where clubs aim to identify suitable replacements for departing players. By incorporating additional data sources, such as match statistics, player tracking data, and tactical preferences, this or a similar KG could evolve into a comprehensive decision-support tool for clubs. (LO9, LO11) Professional football teams, such as Brentford FC, have already showcased the effectiveness of data-driven recruitment models. They focus on scouting talent from underappreciated leagues using key metrics such as Expected Goals (xG) and passing accuracy. This strategy has enabled them to compete successfully against financially superior clubs. Beyond recruitment, Brentford also applies analytics to optimize in-game tactics, refining set-piece strategies, counter-attacks, and pressing structures. [4] (LO9)

Beyond player evaluation, football KGs also offer significant financial applications. Player valuation, salary estimations, and transfer market trends can be analyzed by connecting financial data such as transfer fees, and player career trajectories within the KG structure. Even in the simple GNN case analyzed in this report, the target variable could be switched from potential to a player's potential future transfer fee. In modern football, where transfer fees continue to rise, securing the right player at the right time can be a decisive factor, particularly for smaller clubs whose financial survival depends on scouting and developing undervalued talent. By efficiently selling high-potential players and replacing them with statistically comparable but lesser-known alternatives, clubs can maintain financial sustainability while remaining competitive on the field. Furthermore, a well-structured football KG could be leveraged for media analytics and betting models, both of which, with the huge sums of money involved in soccer, are markets that are worth billions. (LO9, LO10, LO11)

Conclusion

For this project, I focused on developing a football-centric Knowledge Graph (KG), showcasing its applications in data-driven analytics. An ontology was constructed and populated, integrating player, club, and league data and providing a structured foundation for the models implemented (LO2, LO5, LO7). Two use-cases were explored: a Graph Neural Network (LO3) for potential prediction, and Knowledge Graph Embeddings (LO1) for player similarity analysis. The GNN leveraged the structured relationships within the graph to predict a player's future ability, while the KGE approach demonstrated how entity embeddings can be used to identify statistically similar players. These implementations illustrate the discussed versatility of KGs in football analytics, ranging from scouting and recruitment to financial decision-making. (LO9, LO10, LO11)

In the course of implementing these applications, the connections between Knowledge Graphs (KGs), Machine Learning (ML), and Artificial Intelligence (AI) were highlighted. The KG was populated with preprocessed data and served as a structured data representation, allowing the ML models (GNN, TransE) to leverage the relational information that traditional tabular data often lacks. In this project, the KG provided a meaningful framework for feature engineering in the GNN, enabling the model to incorporate both direct player attributes and contextual insights derived from club and league interactions. Similarly, KGEs captured latent relationships between entities similarity assessments.

[4] https://medium.com/@abbasmerchant60/moneyball-in-football-brentfords-data-driven-success-in-the-premier-league-66f5c72d8f01 (03.03.2025, 22:48)

These examples demonstrate how KGs enhance ML models by embedding domain knowledge into the learning process, making them a powerful tool for AI applications. (LO12)

While the use cases explored here are to be seen as proof-of-concept, they already illustrate the value of a football KG. With further expansion, incorporating richer match event data, tactical insights, and economic indicators among others, such a system could evolve into an essential decision-support tool for clubs, analysts, and stakeholders in the football industry.

Disclaimer

Generative AI tools were utilized throughout this project primarily for grammar and spell-checking, as well as for occasional debugging assistance. While all conceptual work, analysis, and interpretations were conducted independently, Generative AI tools were employed as a supportive tool.