ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

FOOTBALL PLAYER FEATURE LEARNING USING GRAPHS FOR RECOMMENDATION AND RETRIEVAL

M.Sc. THESIS

Öznur İlayda YILMAZ

Department of Computer Engineering

Computer Engineering Programme

JANUARY 2023

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

FOOTBALL PLAYER FEATURE LEARNING USING GRAPHS FOR RECOMMENDATION AND RETRIEVAL

M.Sc. THESIS

Öznur İlayda YILMAZ
(504201501)

Department of Computer Engineering

Computer Engineering Programme

Thesis Advisor: Prof. Dr. Şule GÜNDÜZ ÖĞÜDÜCÜ

JANUARY 2023

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ÖNERİ VE ÇIKARIM İÇİN ÇİZGE KULLANARAK FUTBOLCU ÖZELLİKLERİ ÖĞRENİMİ

YÜKSEK LİSANS TEZİ

Öznur İlayda YILMAZ
(504201501)

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Şule GÜNDÜZ ÖĞÜDÜCÜ

OCAK 2023

Öznur İlayda YILMAZ, a M.Sc. student of ITU Graduate School student ID 504201501 successfully defended the thesis entitled "FOOTBALL PLAYER FEATURE LEARNING USING GRAPHS FOR RECOMMENDATION AND RETRIEVAL", which he/she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Prof. Dr. Şule GÜNDÜZ ÖĞÜDÜCÜ** Istanbul Technical University	....
Jury Members :	**Prof. Dr. Behçet Uğur TÖREYİN** Istanbul Technical University	....
	Asst. Prof. Günce Keziban ORMAN Galatasaray University	....
		....

Date of Submission : **30 December 2022**
Date of Defense : **27 January 2023**

To my beloved Grandmother and Grandfather in Heaven,

FOREWORD

First and foremost, I would like to praise and thank God for providing me with strength and blessings throughout my life.

Furthermore, I would like to express my heartfelt gratitude to my advisor, Prof. Dr. Şule Gündüz Öğüdücü, for her unwavering support, never-ending patience, and invaluable advice.

I would like to express my deepest gratitude to my mother and father for always being there for me and standing by my side through thick and thin. I am extremely grateful to my younger brother for being the best brother anyone could ask for.

I would also like to thank my wonderful friends Bensu Özden, Deniz Doğan, Hilal Yazar, Kıymet Çelebi, Kıymet Kaya, Melike Deniz İnan, Şule Giritlioğlu, Yaren Yılmaz and Yasemin Keskin for always bringing a smile to my face even when it seemed impossible.

I would also like to gratefully acknowledge Google DeepMind for supporting me throughout my master's education.

January 2023

Öznur İlayda YILMAZ
Computer Engineer

TABLE OF CONTENTS

ABBREVIATIONS

ANN	: Artificial Neural Network
BPNN	: Back-Propagation Neural Network
BCE	: Binary Cross-Entropy
BFS	: Breadth-First Search
DFS	: Depth-First Search
XGBoost	: Extreme Gradient Boosting
KNN	: K-Nearest Neighbors
MLR	: Multiple Linear Regression
NMF	: Non-Negative Matrix Factorization
ReLU	: Rectified Linear Unit
UEFA	: Union of European Football Associations

SYMBOLS

$\mathbf{P_{t,\,i,\,m}}$	: Player feature vector of player *i* from team *t* in match *m*
$\mathbf{S_{t,\,i,\,j}}$	: Substituted pair of players *i* and *j* from team *t*
y	: Output of subtitution pair: 0 or 1
PR	: PageRank centrality
CC	: Closeness centrality
BC	: Betweenness centrality
DC	: Degree centrality
M	: Number of matches
m	: match index
p	: player index
$\mathbf{E_{p,m}}$	: Player embedding of player *p* for match *m*
$\mathbf{E_{p}}$	: Averaged player embedding of player *p*

LIST OF TABLES

LIST OF FIGURES

Page

FOOTBALL PLAYER FEATURE LEARNING
USING GRAPHS FOR RECOMMENDATION AND RETRIEVAL

SUMMARY

Football analytics is a field that has grown tremendously over the years as technology for capturing data in sporting events has improved. During the transfer season, scouts and football managers must conduct numerous analyses in order to build strong and successful teams. In order to succeed and win matches, managers' in-game decisions are critical.

A substitute is a player who enters the field in place of another player during a football game. This thesis proposes a recommendation system to assist football managers in making substitution decisions. The proposed system utilizes passing networks of teams to learn feature embeddings of football players. Using the extracted player embeddings, a k-nearest neighbors model, an XGBoost model and an artificial neural network model were trained to learn the relationship between suitable player pairs that can be substituted during the match. The model recommends the most suitable football player to substitute for a given input player. The main objective of employing passing graphs to gain knowledge of key player aspects is that the features extracted are unbiased. To prove the validity of this approach, two feature sets containing player attributes assigned by the FIFA19 game were used as baselines. The experiments demonstrated that using features extracted from passing graphs for player substitute recommendations yields models with higher recommendation performance when compared to the baseline models.

Aside from decisions made during football games, managers are also responsible for decisions made off the field, such as selecting players to transfer to the team. Finding the most similar and suitable football players to fill a vacant position left on the team by another player who has left the club is essential for keeping the team dynamics together. A similar player retrieval system was developed for this thesis. The model employs egocentric passing networks calculated individually for each player, which take into account the passing relations of the nearest players as well as centrality values of players, which describe the players' roles in the network. It has been demonstrated that when combined with the baseline model's features namely player pass length, pass direction, and pass location, the proposed new features improve the performance of the player retrieval task.

The experiments showed that using passing graphs of football teams to learn player representations can be beneficial for producing better performing recommendation and retrieval models.

ÖNERİ VE ÇIKARIM İÇİN ÇİZGE KULLANARAK FUTBOLCU ÖZELLİKLERİ ÖĞRENİMİ

ÖZET

Günümüzde, spor etkinliklerinde veri toplama teknolojisinde görülen büyük gelişmeler sayesinde spor alanında makine öğrenmesi ve yapay zekanın kullanımı bir hayli yaygınlaşmıştır. Uzmanlar, maçlar sırasında toplanan veriler ile maç skoru tahmini, takım ve bireysel oyuncu performansı analizi gibi birçok alanda çalışmalar gerçekleştirmektedir. Dünya çapında en popüler spor olan futbol da dolayısıyla bu çalışmalara konu olmaktadır.

Futbol takımlarının teknik direktörlerinin maçların içinde ve dışında verdikleri stratejik kararlar başarılara direkt olarak etki etmektedir. Maç sırasında verilen kararlardan biri oyundan çıkacak futbolcunun ve onun yerine oyuna girecek olan futbolcunun belirlenmesidir. Bu aksiyonun gerçekleştirilme sebepleri oyun sırasında sakatlanan bir futbolcunun yerine yedeğin girmesi, beklenen performansın altında performans sergileyen oyuncunun değiştirilmesi ya da tamamen taktiksel hamleler olabilir. Maç skorunun korunması amacıyla ofansif bir futbolcunun çıkarılıp yerine defansif role sahip bir oyuncunun oynatılması maç sırasında teknik direktörler tarafından yapılan stratejik hamlelere bir örnek olabilir.

Bu çalışmada, teknik direktörlere maç esnasında yapacakları değişiklikler için önerilerde bulunan bir sistem sunulmaktadır. Sistem, girdi olarak aldığı futbolcuya en uygun yedek futbolcuyu önermektedir. Önerilen modelin amacı herhangi bir insani hata ya da eğilim olmadan futbolcuların özelliklerinin çıkarılması ve bu özellikler ile başarılı önerilerin yapılmasıdır. Futbol maçlarında en çok gerçekleşen aksiyon futbol oyuncularının kendi aralarında paslaşmasıdır. Bu yüzden bu çalışmada oyuncuların pas özellikleri kullanılmıştır.

Futbol takımlarının her maçı için kendi içindeki paslarını ifade eden çizgeler oluşturulur. Bu çizgelerde düğümler futbolcuları ifade etmektedir. Oyuncuların kendi içlerinde yaptıkları pas sayıları ise kenarların ağırlıklarıyla ifade edilmektedir. Paslar yönlü olduğundan, oluşturulan çizgelerin kenarlarında da yön özelliği bulunmaktadır. Her takım için pas çizgeleri oluşturulduktan sonra futbolcuların vektörler ile ifade edilip makine öğrenmesi algoritmalarında kullanılması için çizge üzerinden düğümlerin gömülümleri çıkarılmıştır. Bu işlem için Node2vec ve GraphWave isimli düğüm gömülüm algoritmaları kullanılmıştır. Bu algoritmalarla, veri setinin ilk %70'i kullanılarak her futbolcunun oynadığı her maçtan düğüm gömülümleri çıkarılıp ortalamaları alınarak futbolcuları ifade eden özellik vektörleri oluşturulmuştur.

Geçmiş futbol maçlarına ait değişiklik bilgileri kullanılarak, daha önce maç esnasında değişiklik yapılmış futbolcu çiftleri belirlenmiştir. Geçmişte birbiri arasında değişiklik yapılmış oyuncu çiftleri pozitif örneklem olarak alınmıştır. Modellerin oyuncular

arasındaki değişiklik ilişkisini öğrenmesi için veri setine rastgele biçimde geçmişte hiç değişiklik yapılmamış futbolcu çiftleri de negatif örneklem olarak eklenmiştir. Sonuçta, oluşan veri setinde futbolcu çiftlerine ait düğüm gömülümlerinin tutulduğu vektörler ve çıktı olarak pozitif ve negatifi ifade eden 1 ve 0 değerleri bulunmaktadır. Bu veri seti kullanılarak k en yakın komşu, XGBoost ve yapay sinir ağı modelleri eğitilmiştir. Modellerin başarısı ise geçmişte hassasiyet ölçütüyle, geçmişte önerilen futbolcu çiftleri arasında başarılı değişiklikler gerçekleştirilmiş midir kontrolü yapılarak ölçülmüştür.

Önerilen modellerin başarısı, FIFA19 oyununda futbolculara atanmış olan özelliklerin değerleri kullanılarak oluşturulan temel modellerle karşılaştırılmıştır. Böyle bir veri setini kullanmaktaki amaç, bu tarz değerlerde insan bazlı hataların ya da meyillerin olabilmesi ve bu hataların çizgelerden çıkarılan futbolcu özellikleriyle aşılabileceğinin gösterilmesidir. Karşılaştırma amacıyla çeşitli deneyler yapıldığında, çizge üzerinden çıkarılan özelliklerle eğitilmiş modellerin başarısının temel modellere göre daha yüksek olduğu görülmüştür.

Bu tez kapsamında gerçekleştirilen bir diğer çalışmada ise transferlerin gerçekleştiği dönemlerde takımların karar mekanizmasına yardımcı olacak benzer futbolcu çıkarım sistemi hayata geçirilmiştir. Transfer dönemlerinde, futbolcular takım değiştirmektedirler. Bu durumda teknik ekibin takımdan ayrılan futbolcunun yerine ona en benzer özellikle başka bir oyuncuyu transfer edip, takımın dinamiğini ayakta tutması gerekmektedir.

Normal şartlarda benzer futbolcu belirleme işlemi yetenek kaşifleri ve teknik direktörler tarafından çeşitli görüntüler izlenerek yapılmaktadır ve hataya açıktır. Bu çalışma, veriye bağlı kalarak makine öğrenmesi yöntemleriyle insan hatasının olmadığı bir sistem önerek benzer futbolcuları bulmayı amaçlamaktadır.

Karşılaştırma yapılan temel modelde futbolculara ait pasların uzunluk, yön ve lokasyon bilgileri kullanılmıştır. Bu tez kapsamında yapılan çalışmadaki amaç, temel modeldeki çeşitli eksikliklerin giderilerek futbolcuların ayırt edici niteliklerinin daha iyi öğrenilmesi ve dolayısıyla daha doğru benzer futbolcu çıkarımı yapılmasıdır. Temel modeldeki niteliklere ek olarak futbolcuların maçlardaki kişisel ego çizgeleri çıkarılmıştır. Ego çizgelerin kullanılmasındaki amaç oyuncuların en yakın düğümlerdeki oyuncularla olan pas alışkanlığının yakalanabilmesidir. Node2vec algoritması kullanılarak futbolcu gömülümleri elde edilmiştir. Oyuncu bazlı ego çizgelerden çıkarılan gömülümlerin yanı sıra, pas çizgelerin tamamı kullanılarak her futbolcu için çeşitli merkezilik değerleri hesaplanmıştır. PageRank, Yakınlık, Aradalık ve Derece merkezilik değerleri hesaplanmıştır. Tüm nitelikler birleştirilerek futbolcuları ayırt edebilecek özellik vektörleri oluşturulmuştur.

Temel model ve önerilen modeller çeşitli deneylerle incelenmiştir. Deneyler sonucunda, Derece merkezilik değerinin futbolcuları ayırt etmede negatif etkisi olduğu görülmüştür. Bu sebeple, son modelde bu nitelik kullanılmamıştır. Temel modelde futbolcu lokasyonunu belirten niteliklerde ayırt edicilik çok sağlanamadığından, çizge özellikleri kullanarak başarının artırılması amaçlanmıştır. Sonuçlar incelendiğinde, oyuncu bazlı pas ego çizgelerinden elde edilen futbolcu gömülümleri ve takıma ait pas çizgesi üzerinden elde edilen merkezilik değerleri eklenerek futbolcuları daha başarılı bir şekilde ayırt edip tanımlayan vektörlerin oluşturulduğu görülmektedir. Bu şekilde, oyun stili olarak benzer futbolcular çıkarılabilmektedir.

Bu tez kapsamında gerçekleştirilen çalışmalar sonucunda, futbolcuları temsil edecek vektörlerin oluşturulması için çizgeler kullanılmıştır. Pas çizgeleri kullanmaktaki amaç, az veri kullanarak insan hatasını sıfıra indirgeyerek objektif öneri sistemlerinin kurulmasıdır. Deneyler, çizgeler kullanılarak elde edilen verilerle eğitilen modellerin geleneksel veri setleriyle eğitilen modellere kıyasla daha yüksek başarı verdiğini göstermektedir.

1. INTRODUCTION

"Football is the most important of the least important things in life," says Arrigo Sacchi [1]. Being the most popular sport in the world with over 3.5 billion fans worldwide, it is safe to say that this statement is true for the vast majority of people [2].

It is therefore implausible to believe that the recent advances in data collection and applied machine learning algorithms will not be used to analyze the game of football itself. Due to the significant advancement in the technologies enabling the data collection in sporting events, the field of football analytics has been expanding over the past few years. Many contributions to the domain of football have been made in recent years using various machine learning techniques. The majority of these studies focused on match score prediction, as predicting a match ahead of time is the most common goal of football fans worldwide. Other than match prediction, team and player evaluation play a huge role in football. Managers constantly evaluate and analyze the performance of the team and its players in order to build a successful team as a whole.

One of the biggest football clubs, Liverpool, is using data science to help them improve the performance of their team. Using in-game data, Liverpool's data science team creates pitch control models. They create models that advise football players on where to pass in specific situations on the field. According to their data scientists, Liverpool was the Premier League team that conceded the fewest goals in 2020.

Data science and analytics are used not only by football teams, but also by players individually. Kevin De Bruyne, a Belgian footballer, is said to have used data analysis to negotiate an £83 million contract with his team, Manchester City. With the help of data analysts, the player was able to assess his impact on the team, how he's expected to play in the near future, and predict the team's success based on the current roster.

This demonstration of Kevin De Bruyne's abilities and impact on his team enabled him to sign a new and better contract. [1]

1.1 Purpose of Thesis

The primary objective of teams in football games is to outscore their opponents in a total of 90 minutes, which is split into two halves of 45 minutes. Each team has 11 players who play in various positions such as goalkeeper, defenders, midfielders, and strikers.

Every team's main goal is to win, and the decisions made by team managers play a significant role in this. The purpose of the proposed models, substitution recommendation and similar player retrieval, will be explained in this section.

1.1.1 Substitute recommendation

During the course of a football match, a substitute is a player who enters the field in place of another player. The most common reasons for a substitution are a player who is underperforming, injured, or exhausted. Tactical moves can also be the defining motif of football managers' substitution decisions, such as replacing a striker with a defender to avoid conceding goals or bringing in a striker to score last-minute goals. Football managers also make substitution moves for strategic reasons such as bringing a defender to replace a striker to keep the current score safe or bringing a striker to score last minute goals.

As previously stated, substitution in a football match can have a huge impact on the score. To make a substitution, various strategies can be used. As a result, it is critical to make the best decision possible throughout the game. A manager's unsuitable substitution moves can have a significant impact on the match's outcome. Managers being chastised for making "wrong" substitution decisions is a common occurrence in the football world. Despite the fact that, as previously stated, football analytics is a growing field with researchers working on various areas such as match score prediction and football player evaluation, there has not been much research done on making effective recommendations for smart substitution decisions.

[1] https://www.mirror.co.uk/sport/football/news/kevin-de-bruyne-uses-data-23870686

Hence, a recommendation system that football managers can use to make better decisions when substituting a player is proposed.

1.1.2 Player retrieval for similar player recommendation

Football clubs can acquire and sell players during predetermined times each year during a period called the transfer window. The most significant one occurs during the summer, after most European league seasons have concluded. [2]

When a player is set to leave a club, a football player with a similar playing style must be found to fill the vacant position on the team. Scouts and managers face a difficult task in finding the right replacement. This task is generally done by the experts' personal opinions, which can be subjective or biased.

To overcome the possibly biased opinions of human experts' transfer decisions that can be prone to error, a machine learning approach to find the most similar football players is proposed. This task differs from the problem introduced in Section 1.1.1, which deals with substitutions during football matches, in that this task requires a generally similar football player retrieval for transfers between clubs.

1.2 Contribution of Thesis

In this section, the contribution made to the literature through the substitute recommendation and similar player retrieval models will be explained respectively.

1.2.1 Substitute recommendation

Many previous studies on player performance prediction and evaluation have been conducted using football player data consisting of skill ratings assigned to each player based on past statistics spanning many years. For example, one of the most well-known databases from FIFA[3] games is used to evaluate player performance. Adjusting these player ratings necessarily requires a large workforce, including 6,000 volunteers who

[2]https://theathletic.com/3338294/2022/06/04/transfer-window-faq/
[3]https://www.ea.com/games/fifa

help maintain and update the player database over time by attending games and observing the players.[4]

Given that such a database requires the analysis of a large amount of statistics and historical data, learning feature embeddings of football players using only their passing networks is proposed.

During a football match, decision making is pivotal for football managers because a substitute can change the outcome of the game. That is why it is extremely crucial to find the best matches for a specific player who will be substituted. The proposed model is focused on learning player embeddings using passing information in a football game. Learned player embeddings are used to recommend the best possible substitute football player to a certain another player.

This study's major objective is to determine how effectively in-game passing networks can be used to extract football players' characteristics and suggest similar players based on those extracted features for substitution during football matches. The following are the contributions of the suggested model:

- Utilizing node representation learning methods Node2vec [3] and GraphWave [4] to learn vector representations of football players using **passing networks** of football teams.
- A **recommendation system that recommends the most similar and suitable football player to be substituted** in the football match for a given input player.
- **Eliminating the possible bias of football player ratings and ranked attributes** assigned to these players by extracting key features directly from the players' passing behavior in games.

[4]https://www.goal.com/en-ae/news/fifa-player-ratings-explained-how-are-the-card-number-stats/1hszd2fgr7wgf1n2b2yjdpgynu

1.2.2 Similar player retrieval

Players change teams all the time during transfer seasons of football leagues. The vacant positions left by the players leaving the clubs should be filled with other players that are similar in their playing styles, in order to keep the team intact. To do so, hours of game footage should be viewed by the scouts and experts, which can be error-prone.

This study's objective is to find the most similar football players for a given input player, in order to help the experts with the player transfer decisions. The contributions of the proposed model are as follows:

- It has been noted that currently in the literature visual heatmaps that describe the locations of players' playing area are used to learn key player features. Heatmaps can sometimes fail to help describe a player's style, as players can play in different positions, therefore locations, for different matches. This study leverages **the use of passing networks** to overcome the problem of varying locations of players.
- The most significant player characteristics are extracted using **player-based egocentric passing graphs to describe players.** It has been observed that better representation vectors were generated through the usage of egocentric graphs for players.

1.3 Structure of Thesis

This thesis is structured as follows: Firstly in Section 2, previous related works on football match and player analysis will be introduced and discussed. Then, in Section 3 a novel football player substitution recommendation system will be explained. Section 4 introduces the player representation learning for similar player retrieval. Conclusions will be given and future works will be discussed in Section 5.

2. LITERATURE REVIEW

In this chapter, previous researches conducted on the area of football team and player analysis will be explained. In terms of objectives, related works on this subject can be divided into three categories: performance prediction, passing styles analysis, and player representation learning. Although there are other researches done on football such as football coach analysis, teams' tactic prediction, player formation analysis etc. [5]–[8]

2.1 Performance Prediction

Predicting a game's outcome in advance is one of the most common uses of football data. For instance, [9] examines the usage of neural network-based sports match prediction models. Using a dataset containing historical football match data of Union of European Football Associations (UEFA) Champions League for the season of 2016–17, a back-propagation neural network (BPNN), a multiple linear regression model (MLR) and a grey degree prediction models are built. Experiments revealed that the BPNN model had the lowest prediction error. There are also various different works focused on football match prediction using historical data [10]–[13].

Prediction is also used to assess player performance, such as [14], which focuses on predicting long-term team and player performance. It claims that rating systems have a bias toward forwards and attacking midfielders because goals are regarded as the most important aspect of a football match. This study, on the other hand, is primarily concerned with central defenders. A dataset with three categories, namely player characteristics and attributes, player statistics, and team statistics, is used. Pariath et al. use a similar set of features containing football player skills and attributes to estimate a player's performance value. A linear regression model is chosen for training based on the data. The model has an R-square value of 84.34% [15].

2.2 Passing Style Analysis

During football games, passing the ball around the field to different teammates is the best way to keep control of the ball. It is critical to keep possession of the ball when attacking with the purpose of scoring. Since passing plays a huge role throughout the course of football matches, it can be beneficial to study passing networks of football teams to gain knowledge. Several researches have been conducted for passing network analysis of football teams. These studies differ in that their objections shift between individual player and team analysis.

Srinivasan leverages the use of passing distribution statistics of players for prediction and evaluation of football team performance. The dataset stores the information of how successfully players pass the ball around between each other, resulting in a goal or other events during a football match. The number of passes made by a player during a game is examined. According to the author, the greater the number of passes made by the player, the greater the player's influence on the game. The variation in PageRank scores [16] of players across games is also examined. Experiments show that a low negative variation in PageRank scores over time demonstrates this player's consistency, as PageRank is a measure of a node's importance in the network as well as the nodes adjacent to it [17]. Moreover, Medina et al. investigate the importance of using social network analysis for football matches. For the social network analysis, the number of passes completed from one player to another in each match for both teams is used. Weighted and directed networks are built for each game to classify the topology by using the vertex degree, betweenness centrality, and entropy values. As a baseline, a regression model based solely on team performance in the past is used to predict the outcome of a match. When the knowledge gained from the weighted and directed graph measures is used, the prediction accuracy improves by 14% when compared to the baseline model. This study is a demonstration of how the utilization of social network analysis can improve a model's performance [18].

One method for analyzing passes is to use information from flow motifs in networks. According to Peña and Navarro, the way teams pass the ball around the field most accurately characterizes their unique style. They examine football players' passing sequences to determine each player's distinct playing style. They examine individual players' flow motifs and patterns to see which ones are most similar to Xavi Hernandez, a player who they claim to be an outlier in terms of his passing style [19]. Bekkers and Dabadghao use patterns to extract motifs of players and teams to assist in decision-making. They propose using radar graphs to represent the motifs in order to gain insights and extend their model to analyze the unique styles of both teams and players [20]. By examining passing sequences and the themes found in these sequences, Barbosa et al. seek to identify the players who are the most similar. They suggest that by examining both flow motifs and the specific positions the players occupy inside the motifs, the performance of similarity analysis can be improved. They assert that the position feature helps in providing a more precise identification of football player characteristics [21].

Sattari et al. propose a model to represent football player roles based on in-match passing and receiving interactions. Their aim is to define 12 different player roles by extracting linear combinations from players' passing networks using non-negative matrix factorization (NMF). The experiments show that passing and receiving networks help to identify passing styles [22].

Some studies examine the impact passes have on teams as a whole. Travassos et al. use spatial information to assess the success of passes. They analyze the spatial relations between teams and players by using two-dimensional coordinates of passes in football matches for an English Premier League team. They look into how a penetrative pass can be successful or unsuccessful [23]. [24] also calculates the success probability of a pass based on parameters such as the pass's coordinates, distance from the opponent/goal, and so on. They build an XGBoost prediction model. Ahmadi et al. examine and highlight a team's playing style patterns, strengths, and weaknesses using passing and communication networks. They construct communication networks based on the passes, goals, shoots, crosses, and other events that occur during matches. They discover that stronger interaction among team members significantly improves team performance [25].

2.3 Player Representation Learning

Geerts et al. aim to create a player vector that can characterize football players' style so accurately that the current insights of human experts, which can be subjective and prone to mistakes, can be bypassed by an objective machine learning model. They use heatmaps created from players' events in matches such as shots, goals, passes, fouls etc. After applying NMF on these heatmaps, they extract feature vectors that define players. They evaluate the performance of the model by applying a player retrieval task, through which they aim to find the similarities of the same players between training and testing sets the highest [26].

Kim et al. propose a triplet network called 6MapNet to capture the movement styles of footballers using GPS data. They create two heatmaps containing the locations and velocities of football players. Their proposed model maps these heatmap pairs into feature vectors, of which similarities indicate similarities in players' playing styles. To build the network, they employ the idea of FaceNet [27] and build a sixfold network that learns the embeddings of heatmaps belonging to players similar to each other. To evaluate their model, they use a player identification task that checks how accurately the model identifies anonymized players using different seasons' data [28].

A player feature vector called Pass2vec, proposed by Cho et al., is also used to characterize players' traits. The authors use player's pass location, length and direction features to generate a feature vector to describe players' styles. To evaluate performance, player retrieval is performed [29].

The ever-increasing amount of data from football matches is used for a variety of tasks such as football player performance prediction, evaluation, and recommendation. It has been acknowledged that using graph networks to represent player passes in a match can be efficient [30] [18] [31].

3. FOOTBALL PLAYER FEATURE LEARNING FOR PLAYER SUBSTITUTE RECOMMENDATION

In this chapter, a recommendation system to assist football managers with decision-making when a replacement is needed during football matches is introduced. Firstly, the idea behind the usage of graphs for feature learning to train the machine learning models will be explained. Then, the experimental results comparing baseline and proposed models will be discussed. This chapter is based on the paper [32].

3.1 Baseline Models

The suggested method is compared with approaches based on traditional machine learning algorithms, that make use of player features that characterize players' traits and abilities. To that end, as the first step of the framework, datasets with three different sets of player features to train recommender models are prepared. Two of these feature sets are obtained from FIFA19 player ratings. One disadvantage of this approach is that, because most of these features are based on statistical analysis of players' historical data, they must be calculated from information about players gathered over time. Thus, long historical data must be analyzed in order to form a reliable and accurate set of player features.

3.2 Proposed Model

The main goal of this proposed system is to assist football managers with the decision-making of substitutions occurring during football matches.

The proposed and third feature set for the model consists of feature embeddings of football players that are learned by only employing passing graphs of players. The features are extracted from passing networks of players. Players' node representations are learned with Node2vec [3] and GraphWave [4] algorithms. The advantage of these node embedding algorithms is, that they do not require node features to be present, which meets the main goal of this study, which is learning player representations with passing graphs without the use of external features.

The steps of this approach to learn player representations and recommend suitable substitutions during matches can be seen in Figure 3.1.

3.2.1 Player representation learning

Through representation learning, a mapping, that embeds nodes of graphs as points in a low-dimensional vector space $\mathbb{R}^d$, is extracted. The main goal is to optimize this mapping such that local neighborhoods of nodes are preserved, and the embedding space reflects the original structure of the graph. After the embedding space is optimized, the learned embeddings can be used as input features for machine learning models [33].

Thus, the main goal of this work is to learn features that describe a football players' style using the information from passing graphs, in which the players are represented as nodes, and the edges between these nodes represent the number of passes between players.

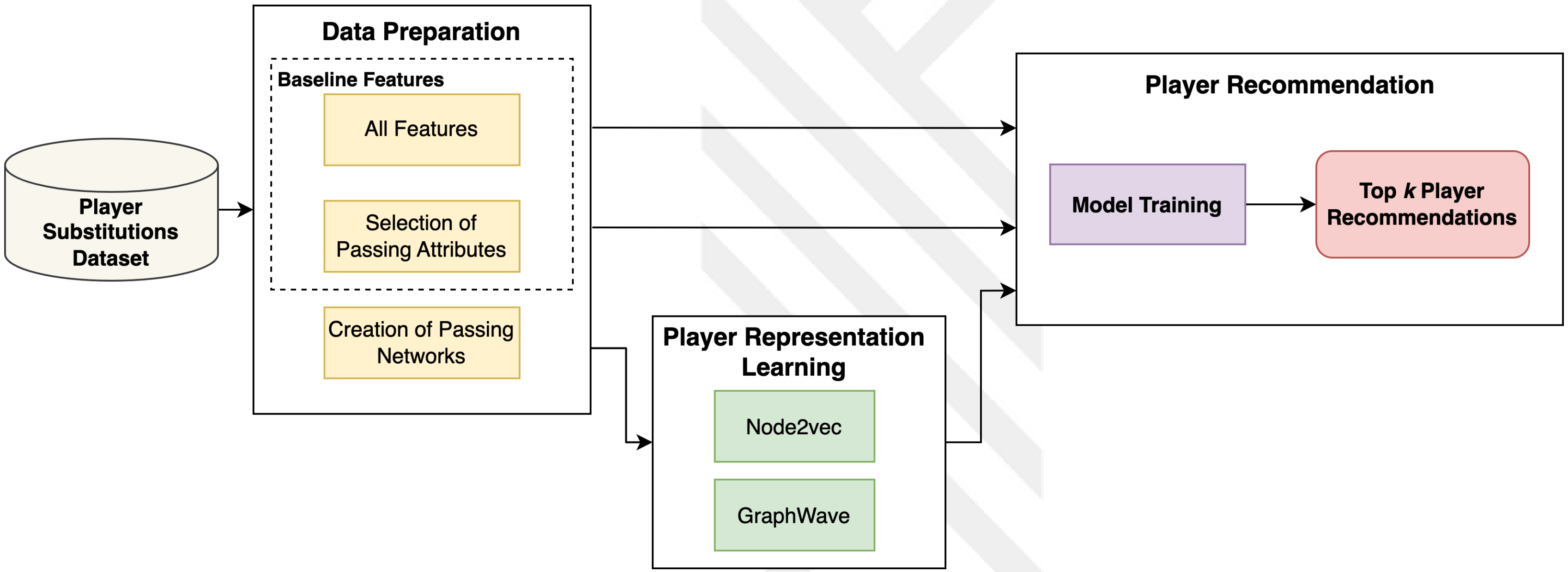

Figure 3.1 : Workflow for player recommendation system.

3.2.1.1 Node2vec

The Node2vec algorithm is employed for the player representation learning step of the model. The algorithm learns node embeddings using a flexible neighborhood sampling strategy that trades-off between Breadth-First Search (BFS) and Depth-First Search (DFS). When the BFS sampling strategy is used, the closest neighbors of the source node are considered. Whereas for the DFS sampling strategy, the nodes are sequentially sampled in increasing order of distances from the source node. The breadth-first and depth-first sampling strategies can be seen in Figure 3.2.

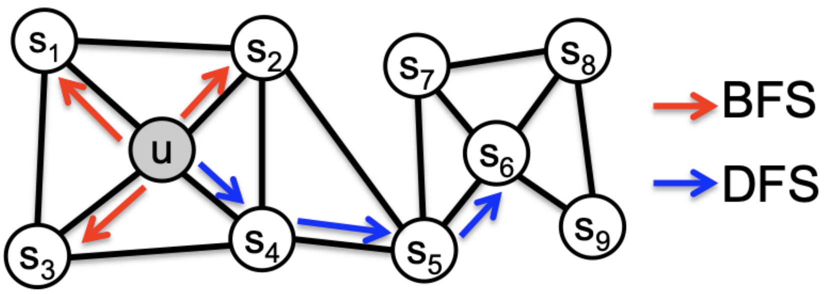

Figure 3.2 : BFS and DFS search from node u with $k = 3$ [3].

A random walk of length l is simulated given a source node u. With $c_0 = u$ being the starting node of the random walk, c_i represents the ith node of this walk. Nodes c_i are generated according to the distribution depicted in Equation (3.1), in which Z is the normalizing constant, and π_{xv} is the unnormalized transition probability between nodes v and x.

$$P(c_i = x | c_{i-1} = v) = \left\{ \begin{array}{cc} \frac{\pi_{vx}}{Z} & if(v,x) \in E \\ 0 & otherwise \end{array} \right\} \quad (3.1)$$

In order to guide the random walk two parameters p and q are used. Figure 3.3 shows a random walk that just passed through edge (t, v) and is now at node v. At this node, the walk considers the transition probabilities π_{xv} on edges (v,x) starting from v.

The unnormalized transition probability is set to $\pi_{vx} = \alpha_{pq} \cdot w_{vx}$. The value of the search bias parameter α is calculated according to Equation (3.2), in which d_{tx} is the shortest path distance between nodes t and x. With these parameters, the random walk procedure changes between breadth-first and depth-first search strategies. Parameter p affects the probability of revisiting a node in the random walk. With higher values of p, the probability of revisiting a node is decreased. Parameter q lets the search to switch between "inward" and "outward" nodes, if $q > 1$ the random walk has a bias towards inward nodes of node t, and on the contrary if $q < 1$ the walk is biased towards nodes that are further away from node t [3].

$$\alpha_{pq}(t,x) = \begin{cases} \frac{1}{p} & \text{if } d_{tx} = 0 \\ 1 & \text{if } d_{tx} = 1 \\ \frac{1}{q} & \text{if } d_{tx} = 2 \end{cases} \tag{3.2}$$

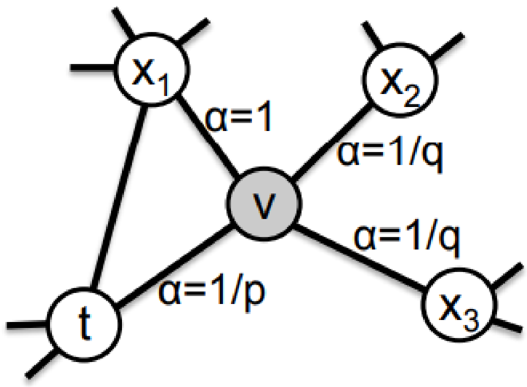

Figure 3.3 : Random walk procedure in *node2vec* [3].

3.2.1.2 GraphWave

GraphWave is another algorithm used to learn node representations in a graph. Using GraphWave, the neighborhood of every node of a network is represented by low-dimensional embedding using heat wavelet diffusion patterns. The GraphWave embeddings of nodes with similar neighborhoods have similar GraphWave embeddings even if these nodes are not close to each other in distance. As seen in Figure 3.4, although nodes a and b are widely separated from each other, both nodes have similar roles. The nodes' respective raw spectral graph wavelets are different, but the algorithm treats them as probability distributions. Thus, proving that structurally similar nodes have similar distributions of wavelet coefficients.

Let U be the eigenvector decomposition of $L = D - A = U\lambda U^T$, which is the unnormalized graph Laplacian, where λ is a diagonal matrix made of λ and $\lambda_1 < \lambda_2 \leq$

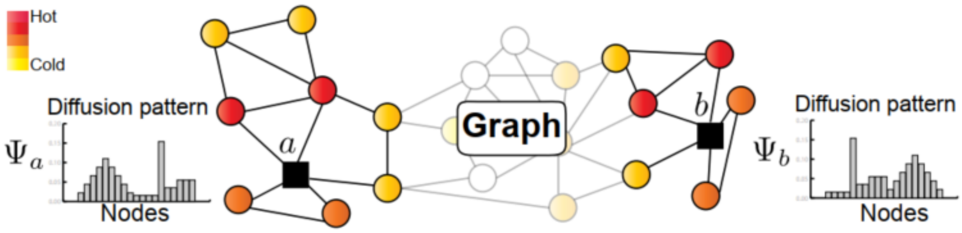

Figure 3.4 : Diffusion Patterns of nodes a and b [4].

... $\leq \lambda_N$ are the eigenvalues of L. Let g_s be a filter kernel with the scaling parameter s. The modulation in the spectral domain of a Dirac signal centered around node a results in the spectral graph wavelet of g_s. Equation (3.3) shows the spectral graph wavelet Ψ_a with an N-dimensional vector, $\delta_a = 1(a)$ is the one-hot vector of node a.

$$\Psi_a = UDiag(g_s(\lambda_1), ..., g_s(\lambda_N))U^T \delta_a \tag{3.3}$$

The basis of spectral graph wavelets is a relation between the temporal frequencies of a signal and the eigenvalues of the Laplacian.

GraphWave computes a $2d$-dimensional vector χ_a for every node a. Spectral graph wavelets are employed to obtain diffusion patterns belonging to each node. An NxN matrix Ψ stores the information of diffusion patterns, in which the a-th column is the spectral graph wavelet for a heat kernel centered at node a.

The spectral graph wavelet coefficient distributions are embedded into $2d$-dimensional space. To do so, the characteristic function for each node's coefficients Ψ_a is calculated and sampled at d evenly spaced points. The characteristic function of a probability distribution X is defined in Equation (3.4).

$$\phi_X(t) = \mathbb{E}\left[e^{itX}\right] \tag{3.4}$$

Given node a and scale s, the empirical characteristic function of Ψ_a is as in Equation (3.5):

$$\Psi_a(t) = \frac{1}{N} \sum_{m=1}^{N} e^{it\Psi_{ma}} \tag{3.5}$$

For node a, the embedding is calculated by sampling the parametric function in Equation (3.6) at d evenly spaced points $t_1...t_d$ and concatenating the values [4].

$$\chi_a = [Re(\phi_a(t_i)), Im(\phi_a(t_i))] \tag{3.6}$$

3.2.1.3 Creating substitution pairs

Three different datasets are used for training models, in order to compare the performances of using different feature sets. Two of these baseline datasets contain features extracted from FIFA19 dataset. The third dataset consists of player features that are extracted using node embedding algorithms.

Using the past player substitution information of matches, pairings of players that have been substituted for each other are built. The feature vectors of players' consist of the players' attributes in the different datasets respectively.

Let $P_{t,i,m}$ be the player feature vector of length n, where i denotes the player of team t of match m as shown in Equation (3.7).

$$P_{t,i,m} = [p_1, p_2, ...p_n] \tag{3.7}$$

The training set is constructed using the first 70% of the matches from every team, whereas the remaining 30% of the matches are used for testing. $P_{t,i}$ in Equation (4.11) represents the average feature vector for football player i over matches M in the training set.

$$P_{t,i} = \frac{\sum_{m=1}^{M} P_{t,i,m}}{M} \tag{3.8}$$

The test set is not used at any step of this procedure; such as calculating the feature vectors of players, construction passing networks, learning features embeddings. It is only used for model performance evaluation purposes.

The construction of the training set is as follows: Let the individual player feature vectors be $P_{t,i}$. All pairs of feature vectors of players i and j ($P_{t,i}$ and $P_{t,j}$) who have been substituted for one another in the first 70% of matches from every team are included in the training set. The pairs that have historically been substituted for one another are the positive samples in the dataset. In order to learn the relationship of players being replaced for each other better, negative samples consisting of player pairings that have **never** been substituted for each other are randomly selected from the first 70% of the matches for every team and are added to the training set.

A sample for a substitution pair $S_{t,i,j}$ in the training set is denoted as shown in Equation (3.9), where y takes values of 1 or 0 according to whether such a substitution between $P_{i,m}$ and $P_{j,m}$ has been made historically.

$$S_{i,j} = [P_{i,m}, P_{j,m}, y] \tag{3.9}$$

Figure 3.5 depicts an example of the resulting training dataset, where every row depicts substitution pairs $S_{t,i,j}$. Player B has been substituted for Player A, so the output for this sample is 1. However, Player C has never been substituted for Player A, making the output value 0. The feature vectors for the players vary according to the dataset used for training, which can be either the FIFA19 player attributes of Node2vec/GraphWave player embeddings.

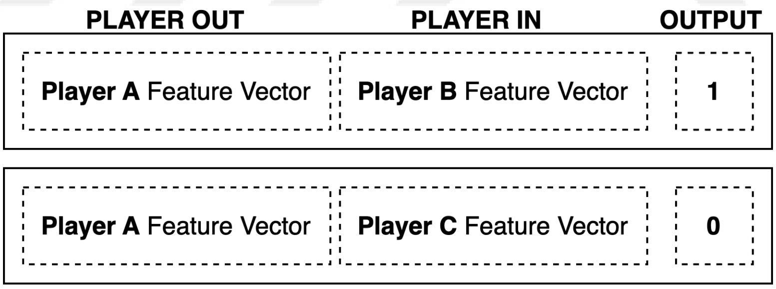

Figure 3.5 : Structure of the data consisting of substitution pairs [32].

In the entire dataset, 30% of the matches are used for testing. For features that are extracted using passing graphs, embeddings for a player leaving a match in the test set are extracted using the passing graph constructed from the passes that have occurred in the corresponding match up until the time point that the player is leaving. The recommendation set is created by only considering the players of the team who are not on the pitch, in other words actively playing, at the time of the substitution. The passing information does not exist for substitute players before the player is in the match, so for the recommendation step, the average feature vectors of these players that have been calculated from the training set are used.

3.2.2 Player substitute recommendation

K-nearest neighbors, XGBoost and neural network models are trained using the training set that contains the first 70% of the matches in the dataset. To recommend a substitution for a particular football player that is leaving the match, only the players from the same team that are not playing on the pitch at the time of the substitution are taken into account. Teams can have from 7 up to 12 players on the bench for every match. Using this pool of players, recommendations are made.

3.2.2.1 K-nearest neighbors

A simple k-nearest neighbors model (k-NN) is used to compare with the other models. The cosine similarity (CS) is calculated for player pairs in the same team in match m. Equation (3.10) shows the cosine similary between two vectors x and y.

$$CS(x,y) = \frac{x^T y}{\|x\| \, \|y\|} \tag{3.10}$$

The output of the model is the top k most similar players to the input player in regards to their cosine similarity.

3.2.2.2 Extreme gradient boosting (XGBoost)

Extreme gradient boosting (XGBoost) is an ensemble machine learning method that uses gradient boosted trees. Using the gradient boosting technique, multiple models are built and combined to get the final prediction [34]. When new models are added, the loss is minimized using the gradient descent algorithm. The probability that a team's bench players will be used to replace a certain player is predicted using the trained XGBoost model. The top k players with the highest probabilities are then suggested as a replacement by the model.

3.2.2.3 Artificial neural network (ANN)

Artificial neural networks are computational models that behave similarly to biological neurons in human brains. An ANN is made of nodes that mimick neurons. Feed forward networks are one of the most commonly used models of neural networks. They are composed of an input layer that receives input information, hidden layers that determine and represent the relationship between the inputs and outputs using node weights, and an output layer that generates outputs for the specific problem [35]. Figure 3.6 shows an example architecture of a basic artificial neural network. Every neuron in a layer processes the weighted sum of its inputs using a non-linear activation function, which becomes the input of another neuron in the next layer of the network [36].

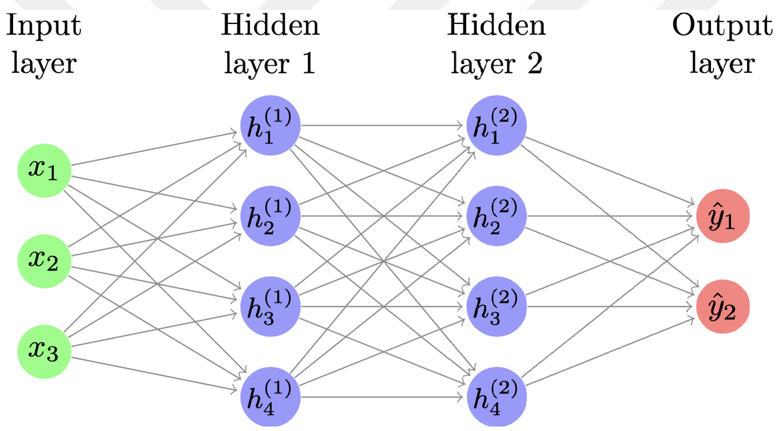

Figure 3.6 : An example of an artificial neural network model.

$$z = f\left(\sum_{i=1}^{n} x_i w_i\right) \tag{3.11}$$

Equation (3.11), where n denotes the input size, shows the computation during the process of forward-propagation, which has been described so far. After forward-propagation, an error is calculated using the final output of the network and the actual target output of the data using a loss function. Equation (3.12) shows the Binary Cross-Entropy loss. Using the gradient descent, back-propagation is performed on the network to adjust the weights between the neurons that contribute to the error. The weights in the network are updated sequentially through forward- and back-propagation until the error is minimized.

$$Loss_{BCE} = -(y\log(p) + (1-y)\log(1-p)) \tag{3.12}$$

A supervised learning method is employed in order to fulfill this task. The dataset of positive and negative substitution pairs is used to train a neural network with a binary output. The network is composed of two hidden layers, each with 16 neurons and 32 neurons with the activation functions Rectified Linear Unit (ReLU) and Tanh respectively, which are defined in Equations (3.13) and (3.14). Softmax, defined in Equation (3.15), is used as the activation function of the output layer.

$$Relu(z) = max(0, z) \tag{3.13}$$

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{1 - e^{-2x}}{1 + e^{-2x}} \tag{3.14}$$

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad for\ i = 1, 2, \ldots, K \tag{3.15}$$

Adam [37] has been chosen as the network's optimization method. The trained network is used to build a recommender system. The system takes a player's features as input and returns the top k suitable substitute recommendations with the highest probability of being substituted in as an output.

Adam optimization algorithm

Adam is an adaptive learning rate method that computes individual learning rates for different parameters. The name, Adam, is derived from adaptive moment estimation. It uses estimations of the first and second moments of gradient to adapt the learning rate for each weight of the neural network [37].

3.3 Experiments and Evaluation

In this section, the different data and feature sets used for experiments will be introduced.

3.3.1 Dataset

In order to evaluate the usage of features extracted from passing networks, two sets of baseline features are used. Sections 3.3.1.1 and 3.3.1.2 introduce the baseline feature sets, and Section 3.3.1.3 explains the features used for the proposed model. The detailed information about the attributes belonging to different feature sets can be seen on Table 3.1.

3.3.1.1 All features of the FIFA19 dataset

In the baseline model, 30 football player attributes of *Crossing, Finishing, HeadingAccuracy, ShortPassing, Volleys, Dribbling, Curve, FKAccuracy, LongPassing, BallControl, Acceleration, SprintSpeed, Agility, Reactions, Balance, ShotPower, Jumping, Stamina, Strength, LongShots, Aggression, Interceptions, Positioning, Vision, Penalties, Composure, Marking, StandingTackle, SlidingTackle, Skill Moves* from the FIFA19 Dataset [1] are used for training. This dataset consists of 17,194 football players playing for 651 different clubs.

3.3.1.2 Pass related features from the FIFA19 dataset

In order to compare the features extracted from passing networks better, only the attributes *ShortPassing* and *LongPassing* from the FIFA19 dataset are used for training and recommendation.

[1] Retrieved 21 October 2021, from https://www.kaggle.com/karangadiya/fifa19

Table 3.1 : Features of datasets.

Datasets	Number of Players	Number of Teams	Time Interval	Number of Features
All Features (FIFA19)	17194	651	N/A	30
Passing Features (FIFA19)	17194	651	N/A	2
Passing Network Graphs	4299	154	2016, 2017/2018, 2018	8/16/32

3.3.1.3 Passing networks dataset

The dataset, that is used to build passing networks, is a public dataset consisting of every event occurring in matches from the Italian, English, Spanish, French and German first division leagues, European and World Championships. National competition data are from the 2017/2018 season, while European and World Championship data are from 2016 and 2018, respectively. The dataset includes 1,941 matches and 4,299 football players. Each match contains events and player data. These events include passes, fouls, shots, duels, free kicks, and so on.

The frequencies of numerous types of events can be seen in Figure 3.7. The frequency of passes is greater than 50%. Almost every player in a football match completes successful passes. That is why this study focuses solely on passes made between team players to extract key features of players.

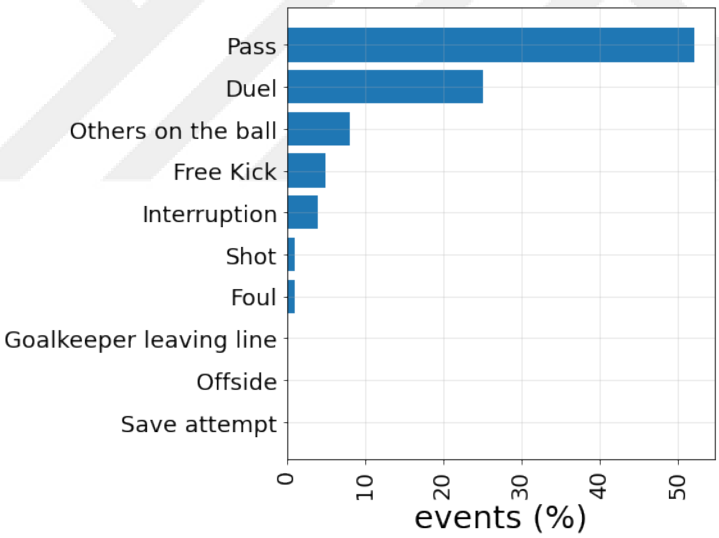

Figure 3.7 : Frequency of events in the dataset [38].

For each of their matches, passing graphs are created for each team. Nodes represent players, and edges represent connections between them. As a result, a directed weighted graph is created, with the edges' weights determined by how many successful passes have been made between players.

Figures 3.8 and 3.9 show examples of passing graphs belonging to Sampdoria and Atalanta teams in a match between them. The data also include substitution

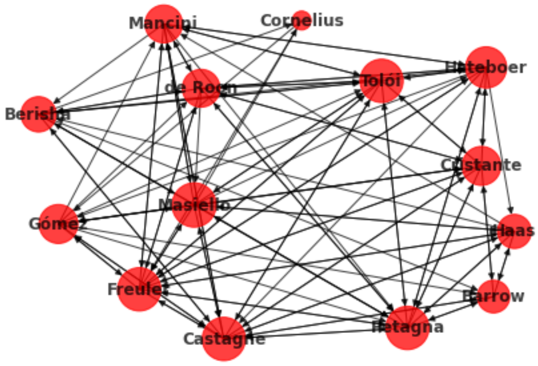

Figure 3.8 : Passing network of Atalanta against Sampdoria [32].

information for each match. Player pairs that are substituted during matches are recorded for each team and match [38]. This data will be used to assess the performance of the models.

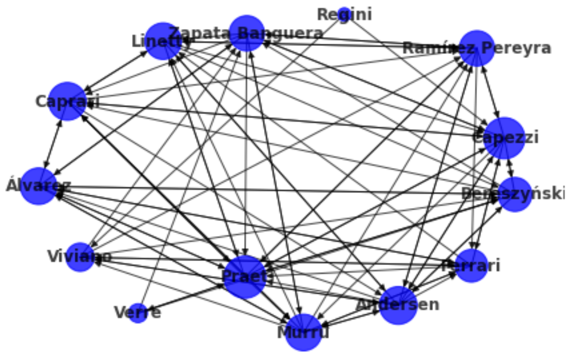

Figure 3.9 : Passing network of Sampdoria against Atalanta [32].

3.3.2 Experimental results

In this section, the experimental results will be explained along with the evaluation metric chosen for the task.

3.3.2.1 Evaluation metric

Precision shown in Equation (3.16) is used for evaluating the perfomances of the models.

$$Precision = \frac{|\bigcap(PredictionSet, ReferenceSet)|}{|PredictionSet|} \tag{3.16}$$

Using the k-NN, XGBoost and neural network models, $k = \{1, 2, 3\}$ recommendations were made. Then, the testing set was used to check if such a substitution between player i and player j were made in the past.

3.3.2.2 Results

The experimental results obtained using the baseline features from the FIFA19 dataset can be seen in Table 3.2, and the results from the proposed model, which uses player features extracted from passing graphs, can be seen in Table 3.3. It can be said that XGBoost models perform better overall.

Table 3.2 : Experimental results using the baseline features of the FIFA19 dataset.

		Precision@k		
		k		
Feature Set	**Model**	**1**	**2**	**3**
Selected Features	k-NN	69.36%	73.86%	76.16%
	XGBoost	72.01%	**75.96%**	77.23%
	ANN	63.72%	63.14%	66.94%
All Features	k-NN	78.81%	**79.14%**	**80.54%**
	XGBoost	78.2%	**79.12%**	**80.52%**
	ANN	76.76%	77.89%	78.81%

Using only the two selected player attributes *ShortPassing* and *LongPassing*, we obtain the highest precision of 75.96% for $k = 2$ recommendations with the XGBoost model.

With the XGBoost model, the highest precision of 75.96% for $k = 2$ recommendations is obtained, when only the two selected pass related attributes from the FIFA19 dataset, namely *ShortPassing* and *LongPassing* are used. The XGBoost has a better performance compared with the other two models. When all 30 attributes, that are pre-defined in the FIFA19 dataset, are used, the k-NN and XGBoost models have the best performance when top-$k = 2$ and $k = 3$ players are being recommended.

Using the passing graphs, different feature sets consisting of Node2vec and GraphWave embeddings are created. These feature vectors are of lengths $8, 16$, and 32.

To compare the proposed model, an experiment using only statistical data is also carried out. For this experiment, player pairings are created for the pairs that have been substituted for one another most of the times according to the historical training data. For the recommendation of $k = 3$ players using only historical data, the model has obtained a precision of only 31.27%.

The effect of different embedding methods

For smaller embedding sizes, models that are trained using Node2vec node embeddings perform better overall. Using a smaller size of player feature vectors may have a positive effect on the identification of the most significant player features. Thus, making the model perform better using the most important key features compared to the other models trained using graph embeddings of bigger sizes. The models that are trained using the feature vectors with the bigger embedding sizes could not learn the important features due to the high complexity of the vectors.

When the results of embeddings acquired using the Node2Vec and GraphWave algorithms are compared, it is clear that the latter method provides better recommendations. The highest precision value, 97.92%, is achieved using GraphWave embeddings of size 32 with an XGBoost model. Due to the consideration of the GraphWave algorithm, that distant nodes can have similar structural roles, the models using GraphWave embeddings produce better results. The primary goal of this work is to identify and recommend similar football players based on their passing behavior among teammates. Identifying players who play similar structural roles in the team is therefore vital, regardless of the fact that they are positioned far apart on the team passing graph. The outcomes demonstrate that the Node2vec algorithm falls short in this regard compared to the GraphWave algorithm.

The effect of using graph embeddings

One can argue that the models that use graph features outperform the models that use all 30 player attributes. Given that employing graph node embeddings requires fewer features to function well, this can also demonstrate its robustness and effectiveness. Users typically want to see fewer items in a recommendation set when using recommender systems. These results demonstrate that the proposed approach, even for a small number of recommendations, has a greater precision. The models often perform better when recommending the top-$k = 1$ player.

Table 3.3 : Experimental results using graph embeddings.

		Precision@k								
		1			2			3		
		Embedding Size								
Embedding Method	**Model**	**8**	**16**	**32**	**8**	**16**	**32**	**8**	**16**	**32**
Node2Vec	k-NN	62.61%	63.45%	62.71%	67.72%	69.95%	70.75%	73.41%	75.32%	77.22%
	XGBoost	74.42%	**80.81%**	77.01	**75.82%**	73.75%	73.82	**79.37%**	76.96%	76.34%
	ANN	67.97%	70.1%	71.26	70.35%	71%	74.54	73.24%	77.5%	76.73
GraphWave	k-NN	83.21%	85.31%	91.62%	83.49%	81.64%	83.36%	76.92%	79.28%	80.55%
	XGBoost	93.58%	**95.34%**	**97.92%**	82.55%	84.37%	**86.81%**	82.51%	82.88%	83.42%
	ANN	86.06%	89.13%	**94.23%**	81.04%	**86.22%**	84.65%	80.45%	**85.5%**	84.1%

4. FOOTBALL PLAYER FEATURE LEARNING FOR SIMILAR PLAYER RETRIEVAL

This chapter introduces a player retrieval system to assist football scouts and experts in identifying the most similar football players to specific players on the team who are being transferred to other clubs and creating a vacant position on the team. Firstly, the baseline model for player representation learning will be explained. Then, the improved proposed method for representing and retrieving similar players will be introduced. Lastly, the experimental results will be given.

4.1 Baseline Model

In this section, the baseline model used to compare the proposed method is explained. Pass2vec, proposed by Cho et al., is used as a baseline. Pass2vec is introduced as a football player style descriptor that characterizes football players' passing style by combining various aspects of their in-game pass information. Pass location, length, and direction features are combined to form a feature vector that accurately describes a football player's playing style. The novel player descriptor vector's performance is evaluated by performing a player retrieval task, where the goal is to find the same player in a list of top-k most similar players [29].

4.1.1 Pass location features

For each player in the dataset, two heatmaps indicating where a player's passes began and ended are created. Figure 4.1 shows an example of the heatmaps that are created in this step. The starting and ending points of a player's passes are marked and then overlaid on a grid of size 50×50 for each player p. The number of passes are normalized according to the minutes that player p played. Then, the counts are smoothed using the Gaussian blur technique. The resulting RGB images are $48 \times 32 \times 3$ in size. Using a Convolutional Autoencoder, the players' pass heatmaps are compressed to size $12 \times 8 \times 4$ and then flattened to length 384.

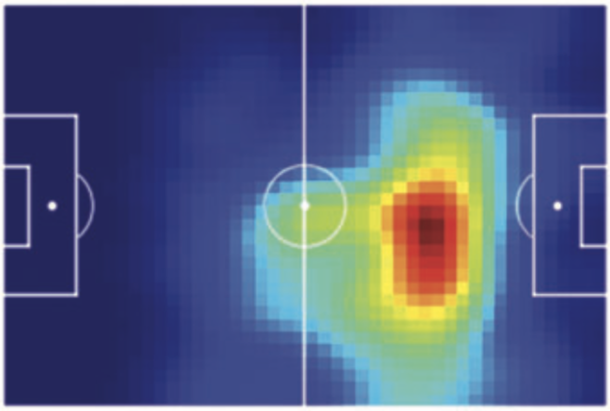

Figure 4.1 : A heatmap of a player's pass locations [29].

4.1.1.1 Gaussian blur

The Gaussian blur is the result obtained by smoothing an image using a Gaussian function for noise reduction. It is a non-uniform low-pass filter that preserves low spatial frequency while reducing image noise and insignificant details in an image. It can be achieved by applying the convolution operation on an image using a Gaussian kernel, which is defined in Equation (4.1) [39].

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4.1}$$

4.1.1.2 Convolutional autoencoder

Convolutional autoencoders consist of convolution layers instead of the fully connected layers that simple autoencoders have. The input and the output sizes are the same. The data fed into the neural network is reconstructed in the output layer. Aa an unsupervised learning approach, the autoencoder does not identify explicit tags when training the data set. The autoencoder also passes as a self-supervised learning model because it creates its own labels as it trains on the input. The network consists of two phases, namely encoder and decoder [40].

Figure 4.2 shows the compression of heatmaps using a convolutional autoencoder model.

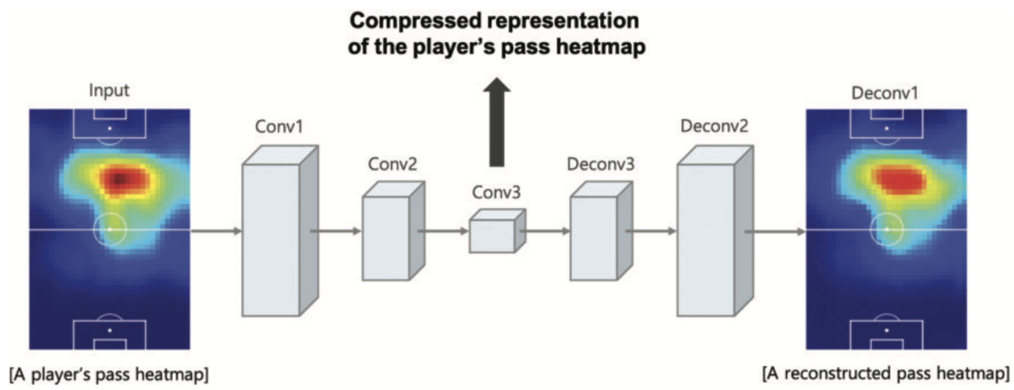

Figure 4.2 : Compression of heatmaps using a convolutional autoencoder [29].

4.1.2 Pass length features

The length of a player's pass is calculated using the start and end locations in the dataset. The Euclidean distance is used to calculate this metric, which is defined in Equation (4.2), in which p and q are two points in Euclidean n-space. Figure 4.3 shows how the pass length is calculated.

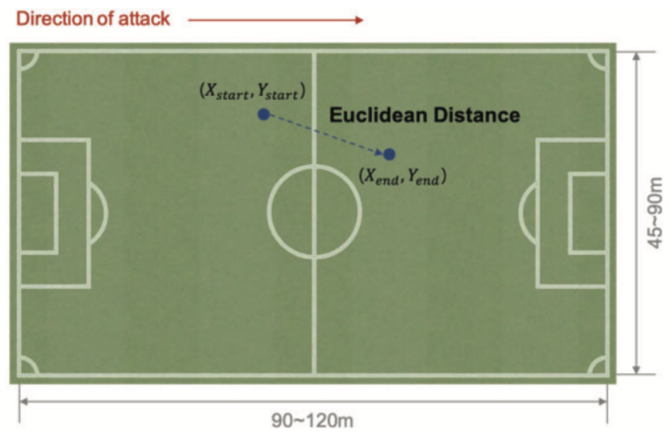

Figure 4.3 : The length of a player's pass [29].

$$dist_{Euclidean}(p,q) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \tag{4.2}$$

The pass length feature vector x_{len} of length 2 is defined as the proportion of short and long passes of a player. A pass is considered long if it is longer than 30 meters, and short if it is less than 30 meters in distance. For example, if 40% of a player's passes are shorter than 30 meters, the pass length feature vector for this player will be $[40, 60]$.

4.1.3 Pass direction features

The pass direction feature vector x_{dir} of length is defined as the proportion of forward and backward passes of a player. A pass is forward if it is played towards the opponent's goal, namely in the direction of the attack. The directions of forward and backward passes can be seen in Figure 4.4.

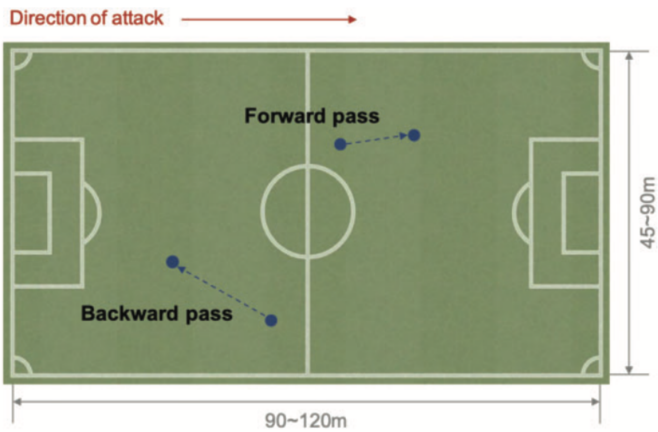

Figure 4.4 : The direction of a player's pass [29].

4.2 Proposed Model

The objective of this proposed model is to retrieve the most similar football players for a given input football player. The proposed system is designed to be of help to football managers and scouts, when making transfer decisions, especially in the process of finding the perfect replacement for players leaving the team.

Using the features from the baseline model and the new enhanced proposed features, namely centralities and ego graph embeddings of football players, the proposed system retrieves and recommends the top k most similar players.

The workflow of the model can be see in Figure 4.5.

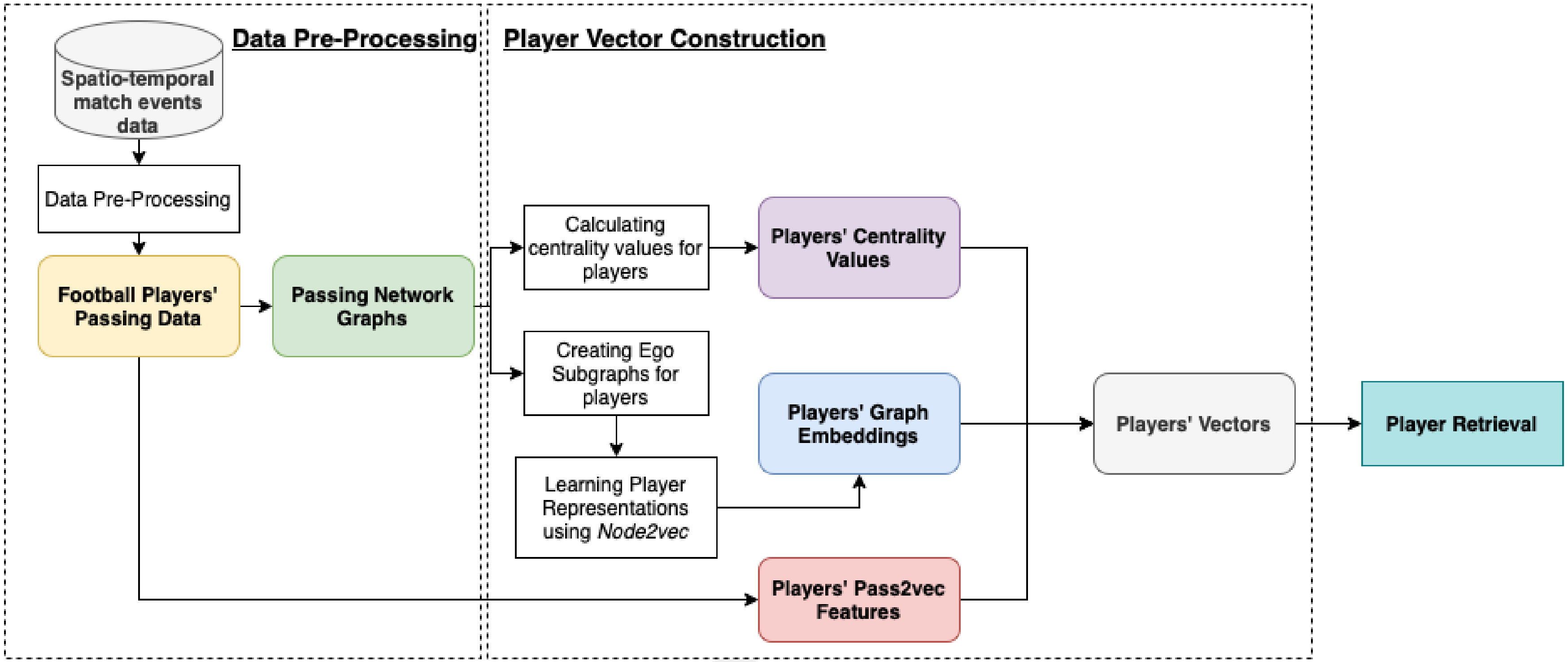

Figure 4.5 : Workflow for player retrieval system.

4.2.1 Player representation learning

As explained in Section 3.2.1, a mapping that represents football players has to be learned in order to be used as input features for machine learning models. This study focuses on representing football players using centrality values obtained from passing graphs, and Node2vec embedding vectors extracted using player-based egocentric passing networks. The creation process of passing network graphs are explained in Section 3.3.1.3.

4.2.1.1 Player centrality values

The importance of nodes in a graph is indicated by centrality. Centrality measures are used to determine how much a node is involved in a network's walk structure [41]. Using several centrality measures, this study seeks to characterize a football player's significance and role inside his team's passing network. In this section, the calculation of player centrality values will be explained.

PageRank centrality

Google utilizes the concept of PageRank centrality to assess the importance of web pages [16]. PageRank centrality is a variant of EigenVector centrality, and its core argument is that a web page is important if it is linked by other important pages. There are three elements that influence a page's PageRank:

- the number of incoming links
- the number of outbound connections
- the PageRank of the pages that are linked to [42]

The PageRank centrality of node A is calculated as shown in Equation (4.3), where d is the damping factor ($d \in [0,1]$), $T_{1..n}$ are the nodes that are connected to node A and $C(T_n)$ is the number of outgoing links to other nodes for node T_n [43].

$$PR(A) = (1-d) + d(\frac{PR(T_1)}{C(T_1)} + ... + \frac{PR(T_n)}{C(T_n)}) \tag{4.3}$$

Closeness centrality

Closeness centrality is a centrality concept that is based on the distance between nodes. It is defined as the average length of the network's shortest paths between the node and the other nodes in the network as seen in Equation (4.4), where d is the length of the shortes path between nodes x and y and N is the number of nodes in the graph [44].

$$C(x) = \frac{N-1}{\sum_y d(y,x)} \tag{4.4}$$

Betweenness centrality

Betweenness centrality is a measure to assess how much a certain node is in-between other nodes in the graph, it captures how much a node has a role in the passing of information from one part of the network to the other [45]. It is calculated according to Equation (4.5), where $\sigma_{v,w}$ is the number of shortest paths from v to w, and $\sigma_{v,w}(u)$ is the number of paths that pass through u [44].

$$B(u) = \sum_{u \neq v \neq w} \frac{\sigma_{v,w}(u)}{\sigma_{v,w}} \tag{4.5}$$

Degree centrality

The number of edges a node has determines its degree centrality. A node with 10 edges has a degree centrality of 10 [45].

PageRank [16], Closeness, Betweenness and Degree centrality values are calculated for every player in every match. Then, these respective centralities are averaged for every player according to Equations (4.6), (4.7), (4.8), and (4.9), where M denotes the number of matches the player p has played.

$$PR_p = \frac{\sum_{m=1}^{M} PR_{p,m}}{M} \tag{4.6}$$

$$CC_p = \frac{\sum_{m=1}^{M} CC_{p,m}}{M} \tag{4.7}$$

$$BC_p = \frac{\sum_{m=1}^{M} BC_{p,m}}{M} \tag{4.8}$$

$$DC_p = \frac{\sum_{m=1}^{M} DC_{p,m}}{M} \tag{4.9}$$

4.2.1.2 Player ego graph embeddings

In this section egocentric networks will be introduced. Then, the generation of ego graphs from teams' passing networks will be explained.

Egocentric networks

An ego centric network, also known as an ego network, is a network composed of an ego node and all other nodes to which the ego is linked (alter) [46]. A 1.0 degree ego graph is a graph that includes the ego node and its alters, whereas a 1.5 degree ego network includes the links between the alters as well. A 2.0 degree ego graph is consists of all of the alters' other connections, which can be seen on Figure 4.6.

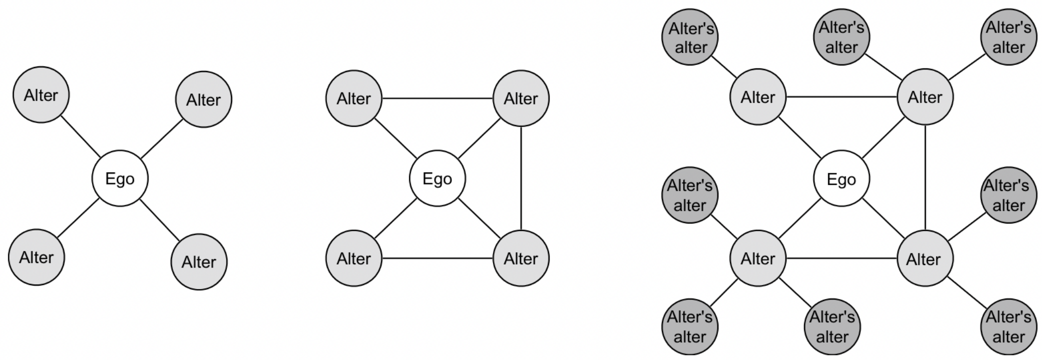

Figure 4.6 : 1.0, 1.5 and 2.0 egocentric networks [47].

Creating player embeddings from ego graphs

The passing style descriptor of a player can be influenced by the players with whom he interacts the most. As a result, we used knowledge gained from team passing graphs. Passing graphs store the direction and number of passes made between teammates during a match. In order to define a player more accurately, ego graphs are used, which take only a node's close neighbors into account. In comparison to using the entire team's passing graphs, this allowed us to capture a player's personal style better.

An embedding of length n for player p from match m can be defined as shown in Equation (4.10).

$$E_{p,m} = [e_1, e_2, ...e_n] \tag{4.10}$$

Equation (4.11) shows, how the average embedding of size n for player p for a total number of matches M is calculated.

$$E_p = \frac{\sum_{m=1}^{M} E_{p,m}}{M} \tag{4.11}$$

E_p is the node2vec graph embedding for player p, and is of length $n = 128$. The node embedding algorithm node2vec is explained in Section 3.2.1.1.

4.2.1.3 Player feature vector

Player descriptor features from the baseline model namely pass location, pass length and pass direction features are concatenated with the new features PageRank, Closeness, Betweenness, and Degree centrality values and the Node2vec embedding for every player p. The resulting player vector is shown in Equation (4.12).

$$P_i = [p_1, p_2, ...p_n] \tag{4.12}$$

The newly calculated features, embeddings from ego graphs and centrality values, can be seen in Figure 4.7.

4.2.2 Player Retrieval

The same method used in the baseline model is chosen to evaluate the performance of the player descriptors. Using the feature vectors computed for every player, a player retrieval task is carried out to assess the performance.

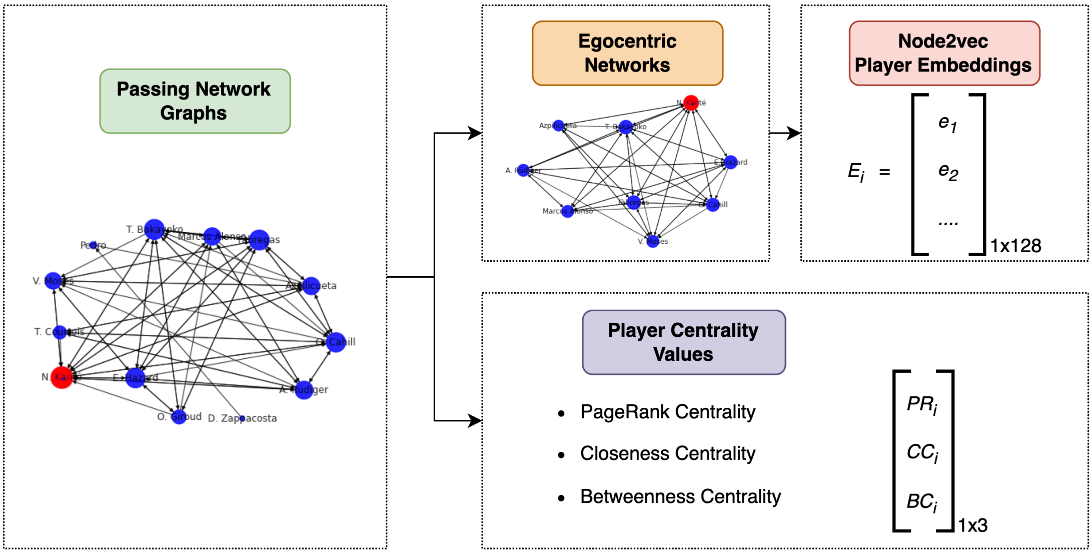

Figure 4.7 : Newly calculated features for feature descriptors.

The main reason for choosing to build a player retrieval model is that evaluating the performance for a model that finds the most similar football players can be tricky. It is assumed that a player's style remains the same or similar throughout seasons. This assumption implies that the model is successful if it can retrieve the same player as the input player, since the goal is to find similar players.

Several experiments were carried out to calculate the distances and similarities between player vectors using the Manhattan distance metric.

The Manhattan distance between two football players P and Q is calculated as shown in Equation (4.13), where $P = [p_1, p_2, ...p_n]$ and $Q = [q_1, q_2, ...q_n]$.

$$dist_{Manhattan}(P,Q) = \sum_{i=1}^{n} |q_i - p_i| \tag{4.13}$$

4.3 Experiments and Evaluation

In this section, firstly the dataset used for the model training will be explained. Then, the experimental results will be given.

4.3.1 Dataset

The dataset used for both baseline and proposed models is explained in Section 3.3.1.3. The dataset consists of first division football leagues of Italy, England, France, Spain and Germany. Figure 4.8 shows passing networks of English teams Chelsea and Liverpool from a match against each other, that took place on May 6th, 2018.

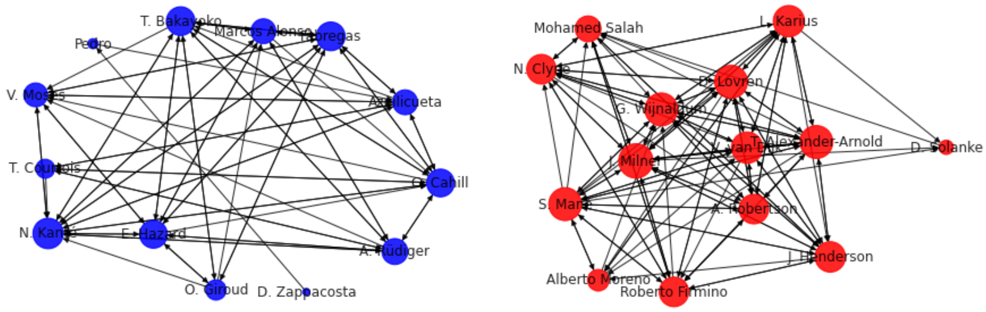

Figure 4.8 : Passing networks of Chelsea (left) vs Liverpool (right) (May 6, 2018).

Figure 4.9 is an example for the player based $r = 1.5$ egocentric graphs computed for every match. This example illustrates the $r = 1.5$ ego graph belonging to Chelsea player N'Golo Kanté.

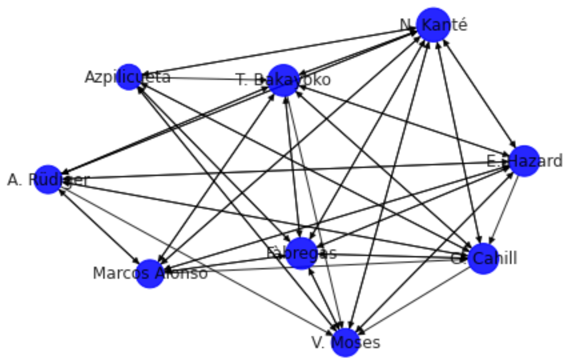

Figure 4.9 : $r = 1.5$ ego network for N. Kanté.

4.3.2 Experimental results

Several experiments to investigate the effect of using different features are conducted. These experiments will be explained and analyzed further in this section. The first halves of every dataset were used for training, and the second halves of the datasets were used for testing purposes. Players with less than 810 minutes (9 matches) of play time were removed from the datasets, in accordance with the baseline model.

4.3.2.1 Evaluation metric

Top-k most similar players to player p are calculated using a distance metric of choice. If player p is present in these retrieved most similar k players, this is counted as a successful retrieval. Total performance is calculated as the ratio of successful retrievals and total number of retrievals.

4.3.2.2 Results

The results investigating the impact different feature sets have on the performance will be explained and discussed in this section.

The effect of using centrality values

The effect of adding centrality values to the feature set was investigated in this section. First of all, player retrieval results were tested using only the Node2vec embeddings of length 128 of players. Then, several experiments using PageRank, Closeness, Betweenness and Degree centralities along with the Node2vec embeddings were carried out. The results can be seen on Table 4.1. The enhancement of the feature set by adding PageRank, Closeness and Betweenness has a positive effect on the performance of the model, whereas the Degree centrality feature has a negative impact on the identification of a player's passing style. Hence, in the following experiments the Degree centrality feature will not be used, as it does not affect the retrieval outcome positively.

It has been demonstrated that degree centrality could not capture players' roles in the networks due to its simplicity. However, the use of PageRank centrality values has a significant positive impact on performance due to their ability to consider both the direction and weight of the edges between nodes, which is an important metric for describing passes between players. Furthermore, Closeness and Betweenness centrality measures have a positive impact on the outcome because they capture the pass communication dynamics and flow in a team, as well as the key players who influence the passing network. Other results obtained using different combinations of centrality values can be seen on Tables A.1, A.2, A.3, A.4.

Table 4.1 : Experimental results obtained by adding centrality values.

Top-k	**Node2vec**	PR**,** CC	PR,CC,BC	PR,CC,BC,DC
1	0.0047	0.019	0.0095	0.033
3	0.014	0.08	0.066	0.071
5	0.023	0.114	0.123	0.123
10	0.042	0.185	0.228	0.18
15	0.076	0.266	0.271	0.242
20	0.104	**0.328**	**0.342**	0.2857

The effect of using ego graphs

To observe the effect of using player based egocentric passing graphs instead of complete passing graphs, some experiments were carried out. Table 4.2 shows the results of these experiments. Tests using $r = 1$, $r = 1.5$ and complete passing networks of players along with the PageRank and Closeness centrality features on the Italian league dataset.

As seen from the results, egocentric graphs perform better than complete passing graphs. Comparing the performances of $r = 1$ and $r = 1.5$ ego graphs, it can be said that for smaller values of k, the $r = 1$ ego graphs yield better retrieval results. However, for $k \in \{15,20\}$ the performances are better with the usage of $r = 1.5$ ego graphs. This can be due to the nature of the player retrieval task, where the main goal is to find the same item from the test list. $r = 1$ graphs consist of the relationships between a node and its immediate neighbors, the nodes that are most similar to the center node.

Table 4.2 : Results obtained by using different subgraphs along with the centralities.

Top-k	$r = 1$ **Ego N.**	$r = 1.5$ **Ego N.**	**Complete Network**
1	0.023	0.019	0.019
3	0.09	0.08	0.071
5	0.119	0.114	0.095
10	0.2	0.185	0.166
15	0.26	0.266	0.219
20	0.3	**0.328**	0.276

Baseline model results

Five major European football league data (Italian, English, French, Spanish and German) were used as separate datasets for these experiments. The baseline model was implemented and tested using these five datasets. Table 4.3 shows the results from the implementation. Pass location, pass length and pass direction features are used to get the baseline results. It can be stated that the model's success varies depending on the dataset used. The Italian and German league datasets have better performance overall in comparison with the others. It has been noted that the baseline model showed relatively poor performance in describing the most key player features for similar player retrieval. This can be due to the usage of heatmaps to describe player pass locations, as the location of a player's passes can differ from time to time. As a result, using heatmaps to distinguish between different players is insufficient.

Proposed model results

In this section, the improved results obtained using the proposed model with the enhanced feature set are explained. The improved results can be seen on Table 4.3. Across all five datasets, it can be said that the model outperforms the baseline results. The feature set consists of pass location, pass length, pass direction, PageRank, Closeness, Betweenness centralities and node2vec player embeddings obtained from $r = 1.5$ ego networks.

The knowledge of graphs was used to overcome the problems encountered when using heatmaps for the baseline models. The models' performances improved after the addition of centrality values and ego graph embeddings. The proposed model performs better overall.

The differences between the retrieval performances obtained for different football leagues can be due to the playing styles adopted by these respective leagues. The English and Spanish leagues are full of individual football stars, who always show their personal talents during games. However, in Italy and Germany the game play is strictly determined by the head of the team and the players play according to their specific roles. This may have lead to better retrieval results, as the passing flow of teams stay similar throughout the season.

Table 4.3 : Experimental results comparing the baseline and proposed models.

Top-k	Italy		England		France		Spain		Germany	
	Baseline Model	Proposed Model	Baseline Model	Proposed Model	Baseline Model	Proposed Model	Baseline Model	Proposed Model	Baseline Model	Proposed Model
1	0.066	0.221	0.037	0.047	0.031	0.053	0.024	0.0537	0.067	0.225
3	0.193	0.424	0.084	0.094	0.05	0.11	0.077	0.1182	0.157	0.45
5	0.253	0.528	0.147	0.1647	0.075	0.25	0.114	0.1935	0.238	0.575
10	0.422	0.726	0.256	0.305	0.119	0.39	0.203	0.3225	0.403	0.85
15	0.536	0.83	0.34	0.4235	0.219	0.55	0.305	0.473	0.537	0.9
20	0.626	**0.886**	0.42	**0.5294**	0.275	**0.643**	0.389	**0.5698**	0.634	**0.95**

5. CONCLUSIONS

In this thesis, the usage of graphs for football player representation learning is investigated. The efficiency of using passing graphs to acquire knowledge for player representation is demonstrated for two different use-cases, namely football player substitution recommendation and similar player retrieval.

For player substitute recommendation, a system that takes an input player and recommends the best *k* substitute players as the output is proposed. The model utilizes the help of passing graphs of teams to extract Node2vec and GraphWave embeddings of players that are used for recommendation. Two baseline features consisting of FIFA19 player attributes that are assigned to players by human experts are used for comparison. The experiments show that through the usage of graph embeddings, possible bias and human errors can be bypassed and better recommendations for in-match substitutions can be given. It has also been demonstrated that for player substitution learning representations of players using the GraphWave algorithm compared to the Node2vec algorithm yields higher performance, due to the former considering distant nodes having structural similarities in graphs.

A model to assist managers in finding similar players to help during the transfer season is also proposed in this study. The proposed model takes features consisting of player pass length, pass direction and pass location from the baseline model into account, and on top of that it enhances the feature set using centrality values and player based ego graph embeddings. Experiments show that the baseline model suffers from the inconsistency of pass location features extracted using player pass heatmaps, which could be due to the constantly changing playing locations of footballers. It has been demonstrated that with the feature enhancement of centralities and ego graph embeddings this problem has been solved, resulting in a better similar player retrieval performance.

Varying node features to describe players can be included in future work. Different team networks containing different in-game event information such as the shots, duels, fouls etc. can be investigated.

REFERENCES

[1] **Black, J.** (2021). Football is “the most important of the least important things”: The Illusion of Sport and COVID-19, *Leisure Sciences*, *43*(1-2), 97–103.

[2] **Giulianotti, R.** (2012). Football, *The Wiley-Blackwell Encyclopedia of Globalization*.

[3] **Grover, A. and Leskovec, J.** (2016). node2vec: Scalable feature learning for networks, *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp.855–864.

[4] **Donnat, C.**, **Zitnik, M.**, **Hallac, D. and Leskovec, J.** (2018). Learning structural node embeddings via diffusion wavelets, *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp.1320–1329.

[5] **Cintia, P. and Pappalardo, L.** (2021). Coach2vec: autoencoding the playing style of soccer coaches, *arXiv preprint arXiv:2106.15444*.

[6] **Lee, G.J. and Jung, J.J.** (2022). DNN-based multi-output model for predicting soccer team tactics, *PeerJ Computer Science*, *8*, e853.

[7] **Kim, H.**, **Kim, B.**, **Chung, D.**, **Yoon, J. and Ko, S.K.** (2022). SoccerCPD: Formation and Role Change-Point Detection in Soccer Matches Using Spatiotemporal Tracking Data, *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp.3146–3156.

[8] **Bialkowski, A.**, **Lucey, P.**, **Carr, P.**, **Yue, Y.**, **Sridharan, S. and Matthews, I.** (2014). Large-scale analysis of soccer matches using spatiotemporal tracking data, *2014 IEEE international conference on data mining*, IEEE, pp.725–730.

[9] **Li, H.** (2020). Analysis on the construction of sports match prediction model using neural network, *Soft Computing*, *24*(11), 8343–8353.

[10] **Zaveri, N.**, **Shah, U.**, **Tiwari, S.**, **Shinde, P. and Teli, L.K.** (2018). Prediction of football match score and decision making process, *International Journal on Recent and Innovation Trends in Computing and Communication*, *6*(2), 162–165.

[11] **Rudrapal, D.**, **Boro, S.**, **Srivastava, J. and Singh, S.**, (2020). A deep learning approach to predict football match result, Computational Intelligence in Data Mining, Springer, pp.93–99.

[12] **Rahman, M.** *et al.* (2020). A deep learning framework for football match prediction, *SN Applied Sciences*, *2*(2), 1–12.

[13] **Guan, S. and Wang, X.** (2022). Optimization analysis of football match prediction model based on neural network, *Neural Computing and Applications*, *34*(4), 2525–2541.

[14] **Pantzalis, V.C. and Tjortjis, C.** (2020). Sports Analytics for Football League Table and Player Performance Prediction, *2020 11th International Conference on Information, Intelligence, Systems and Applications (IISA*, IEEE, pp.1–8.

[15] **Pariath, R.**, **Shah, S.**, **Surve, A. and Mittal, J.** (2018). Player Performance Prediction in Football Game, *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, IEEE, pp.1148–1153.

[16] **Page, L.**, **Brin, S.**, **Motwani, R. and Winograd, T.** (1999). The PageRank citation ranking: Bringing order to the web., **Technical Report**, Stanford InfoLab.

[17] **Srinivasan, B.** (2017). A Social Network Analysis of Football-Evaluating Player and Team Performance, *2017 Ninth International Conference on Advanced Computing (ICoAC)*, IEEE, pp.242–246.

[18] **Medina, P.**, **Carrasco, S.**, **Rogan, J.**, **Montes, F.**, **Meisel, J.D.**, **Lemoine, P.**, **Peñas, C.L. and Valdivia, J.A.** (2021). Is a social network approach relevant to football results?, *Chaos, Solitons & Fractals*, *142*, 110369.

[19] **Peña, J.L. and Navarro, R.S.** (2015). Who can replace Xavi? A passing motif analysis of football players, *arXiv preprint arXiv:1506.07768*.

[20] **Bekkers, J. and Dabadghao, S.** (2019). Flow motifs in soccer: What can passing behavior tell us?, *Journal of Sports Analytics*, *5*(4), 299–311.

[21] **Barbosa, A.**, **Ribeiro, P. and Dutra, I.** (2022). Similarity of Football Players Using Passing Sequences, *International Workshop on Machine Learning and Data Mining for Sports Analytics*, Springer, pp.51–61.

[22] **Sattari, A.**, **Johansson, U.**, **Wilderoth, E.**, **Jakupovic, J. and Larsson-Green, P.** (2022). The Interpretable Representation of Football Player Roles Based on Passing/Receiving Patterns, *International Workshop on Machine Learning and Data Mining for Sports Analytics*, Springer, pp.62–76.

[23] **Travassos, B.**, **Monteiro, R.**, **Coutinho, D.**, **Yousefian, F. and Gonçalves, B.** (2022). How spatial constraints afford successful and unsuccessful penetrative passes in elite association football, *Science and Medicine in Football*, 1–8.

[24] **Anzer, G. and Bauer, P.** (2022). Expected passes, *Data Mining and Knowledge Discovery*, *36*(1), 295–317.

[25] **Ahmadi, A.H.**, **Noori, A. and Teimourpour, B.** (2020). Social Network Analysis of Passes and Communication Graph in Football by mining Frequent Subgraphs, *2020 6th International Conference on Web Research (ICWR)*, IEEE, pp.1–7.

[26] **Geerts, A.**, **Decroos, T.** *et al.* (2019). Characterizing Soccer Players' Playing Style from Match Event Streams, *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2018 workshop*, volume2284, Springer, pp.115–126.

[27] **Schroff, F.**, **Kalenichenko, D. and Philbin, J.** (2015). Facenet: A unified embedding for face recognition and clustering, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.815–823.

[28] **Kim, H.**, **Kim, J.**, **Chung, D.**, **Lee, J.**, **Yoon, J. and Ko, S.K.** (2022). 6MapNet: Representing soccer players from tracking data by a triplet network, *International Workshop on Machine Learning and Data Mining for Sports Analytics*, Springer, pp.3–14.

[29] **Cho, H.**, **Ryu, H. and Song, M.** (2022). Pass2vec: Analyzing soccer players' passing style using deep learning, *International Journal of Sports Science & Coaching*, *17*(2), 355–365.

[30] **Cordón-Carmona, A.**, **García-Aliaga, A.**, **Marquina, M.**, **Calvo, J.L.**, **Mon-López, D. and Refoyo Roman, I.** (2020). What Is the Relevance in the Passing Action between the Passer and the Receiver in Soccer? Study of Elite Soccer in La Liga, *International Journal of Environmental Research and Public Health*, *17*(24), 9396.

[31] **Zhao, Y. and Zhang, H.** (2020). Eigenvalues make the difference–A network analysis of the Chinese Super League, *International Journal of Sports Science & Coaching*, *15*(2), 184–194.

[32] **Yılmaz, Ö.İ. and Öğüdücü, Ş.G.** (2022). Learning football player features using graph embeddings for player recommendation system, *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, pp.577–584.

[33] **Hamilton, W.L.**, **Ying, R. and Leskovec, J.** (2017). Representation learning on graphs: Methods and applications, *arXiv preprint arXiv:1709.05584*.

[34] **Chen, T. and Guestrin, C.** (2016). Xgboost: A scalable tree boosting system, *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp.785–794.

[35] **Dongare, A.**, **Kharde, R.**, **Kachare, A.D.** *et al.* (2012). Introduction to artificial neural network, *International Journal of Engineering and Innovative Technology (IJEIT)*, *2*(1), 189–194.

[36] **Gardner, M.W. and Dorling, S.** (1998). Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences, *Atmospheric environment*, *32*(14-15), 2627–2636.

[37] **Kingma, D.P. and Ba, J.** (2014). Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980*.

[38] **Pappalardo, L.**, **Cintia, P.**, **Rossi, A.**, **Massucco, E.**, **Ferragina, P.**, **Pedreschi, D. and Giannotti, F.** (2019). A public data set of spatio-temporal match events in soccer competitions, *Scientific data*, *6*(1), 1–15.

[39] **Misra, S. and Wu, Y.** (2019). Machine learning assisted segmentation of scanning electron microscopy images of organic-rich shales with feature extraction and feature ranking, *Machine learning for subsurface characterization*, *289*.

[40] **Zhang, Y.** (2018). A better autoencoder for image: Convolutional autoencoder, *ICONIP17-DCEC. Available online: http://users. cecs. anu. edu. au/Tom. Gedeon/conf/ABCs2018/paper/ABCs2018_paper_58. pdf (accessed on 23 March 2017)*.

[41] **Borgatti, S.P. and Everett, M.G.** (2006). A graph-theoretic perspective on centrality, *Social networks*, *28*(4), 466–484.

[42] **Franceschet, M.** (2011). PageRank: Standing on the shoulders of giants, *Communications of the ACM*, *54*(6), 92–101.

[43] **Turan, E.**, **Arslan, E.**, **Çağatay, T. and Orhan, U.** (2020). A Comparison of Graph Centrality Algorithms For Semantic Distance, *Lapseki Meslek Yüksekokulu Uygulamalı Araştırmalar Dergisi*, *1*(2), 61–70.

[44] **Perez, C. and Germon, R.**, (2016). Graph creation and analysis for linking actors: Application to social data, Automating open source intelligence, Elsevier, pp.103–129.

[45] **Golbeck, J.** (2015). *Introduction to social media investigation: A hands-on approach*, Syngress.

[46] **Arnaboldi, V.**, **Passarella, A.**, **Conti, M. and Dunbar, R.I.**, (2015). Chapter 2 - Human Social Networks, **V. Arnaboldi**, **A. Passarella**, **M. Conti and R.I. Dunbar**, editors, Online Social Networks, Computer Science Reviews and Trends, Elsevier, Boston, pp.9–35, `https://www.sciencedirect.com/science/article/pii/B9780128030233000023`.

[47] **Hogan, B.** (2011). *Visualizing and Interpreting Facebook Networks in" Analysing Social Media Networks with NodeXL", DL Hansen et al.*

APPENDICES

APPENDIX A : Player Retrieval Experimental Results

APPENDIX A : Player Retrieval Experimental Results

Table A.1 : Experimental results obtained by using centralities.

Top-*k*	*PR*	*CC*	*BC*	*DC*
1	0.0	0.0142	0.0	0.0047
3	0.0285	0.0238	0.0238	0.033
5	0.038	0.0571	0.0476	0.0571
10	0.076	0.1095	0.0857	0.1476
15	0.1333	0.1619	0.1047	0.1809
20	0.1666	0.1857	0.1428	0.2476

Table A.2 : Experimental results obtained by using centralities with Node2vec embbedings.

Top-*k*	**Node2vec,** *PR*	**Node2vec,** *CC*	**Node2vec,** *BC*	**Node2vec,** *DC*
1	0.0095	0.019	0.0142	0.0095
3	0.0285	0.038	0.0238	0.0333
5	0.0571	0.0428	0.0571	0.066
10	0.1	0.1047	0.1238	0.119
15	0.1571	0.1619	0.1476	0.19
20	0.2047	0.195	0.2095	0.257

Table A.3 : Experimental results obtained by using different combinations of centralities along with Node2vec embbedings.

Top-*k*	*PR*, *BC*	*PR*, *DC*	*CC*, *BC*	*CC*, *DC*	*BC*, *DC*
1	0.0047	0.0142	0.019	0.0142	0.014
3	0.019	0.038	0.038	0.047	0.052
5	0.042	0.066	0.0619	0.08	0.08
10	0.0761	0.166	0.1238	0.1619	0.166
15	0.1523	0.233	0.20	0.2047	0.223
20	0.219	0.266	0.252	0.238	0.271

Table A.4 : Experimental results obtained by using different combinations of centralities along with Node2vec embbedings.

Top-k	PR, CC, DC	PR, BC, DC	CC, BC, DC
1	0.023	0.014	0.0238
3	0.061	0.066	0.0476
5	0.09	0.0904	0.0857
10	0.176	0.1857	0.1619
15	0.223	0.2428	0.214
20	0.261	0.3047	0.2571

CURRICULUM VITAE

Name SURNAME: Öznur İlayda YILMAZ

EDUCATION:

- **B.Sc.:** 2020, Yildiz Technical University, Faculty of Electrical and Electronics Engineering, Computer Engineering

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Yılmaz, Ö. İ.**, & Öğüdücü, Ş. G. (2022, April). Learning football player features using graph embeddings for player recommendation system. In Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing (pp. 577-584).

- **Yılmaz, Ö. İ.**, & Öğüdücü, Ş. G. (2022). Learning football player representations for similar player retrieval. [Under Review].