KU LEUVEN

ARENBERG DOCTORAL SCHOOL
Faculty of Engineering Science

Soccer Analytics Meets Artificial Intelligence: Learning Value and Style from Soccer Event Stream Data

Tom Decroos

Supervisor:
Prof. dr. Jesse Davis

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Computer Science

October 2020

Soccer Analytics Meets Artificial Intelligence: Learning Value and Style from Soccer Event Stream Data

Tom DECROOS

Examination committee:
Em. prof. dr. ir. Robert (Bob) Puers, chair
Prof. dr. Jesse Davis, supervisor
Prof. dr. ir. Hendrik Blockeel
Prof. dr. Patrick De Causmaecker
Dr. ir. Jan Van Haaren
Prof. dr. Ian McHale
(University of Liverpool)

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Computer Science

October 2020

Acknowledgements

In September 2016, I embarked on an exciting journey: a PhD in the DTAI research group at KU Leuven. Like most people who commence a PhD, I had no idea what I was in for; mostly being allured by the opportunity to postpone the existential crisis that comes with deciding what you want to be when you grow up. Now, four years later, I can say with confidence that this was the right path for me. As a member of the DTAI research group, I have learned many things, travelled all over the world, and met wonderful and smart people. My journey in academia, with this dissertation as its final stop, would not have been possible without many people in my life, who I wish to thank.

First of all I wish to thank my promotor Jesse who definitely had the most direct impact on my PhD. Thank you for offering me a position under your tutelage. I imagine I did not always make it easy for you to manage me, but I think you hit the sweet spot between giving me space to work independently and correcting my course whenever I was slacking off with too little progress. I especially appreciate your guidance on producing well written texts and clear presentations. Even now, your meticulous feedback on a new piece of text or some slides can make me feel like a novice again.

I would like to thank my jury members Hendrik, Patrick, Jan, Ian, and Bob. Patrick, doing a bachelor thesis and writing my first paper together with you and Bart is what sparked my curiosity in academic research. I am glad that you are in my jury for the culmination. Jan, you mentored me in the first months of my PhD, being a finishing graduate student yourself back then. Thank you for your mentorship and your continued involvement in my research even when not at KU Leuven. I really enjoyed visiting you in Amersfoort (twice)!

I wish to thank the Research Foundation-Flanders (FWO-Vlaanderen) for supporting my research by awarding me with a personal PhD fellowship.

I would like to thank all current and former members of the DTAI research group for creating an awesome work environment. I will surely miss the coffee breaks

with Arne, Elia, Evgeniya, Gust, Jan, Jessa, Jonas (x2), Kilian, Laurens, Maaike, Mohit, Pedro, Pieter (x2), Robin, Sebastijan, Stella, Tim, Toon, Vincent, Vova, Wannes, and many others. Some colleagues also deserve a separate shoutout. Robin and Laurens, I entrust you with the task of keeping the West-Flemish dialect reverberating through the hallways when I am gone. To the group *excuses of DTAI* (formerly known as *runners of DTAI*), thank you for always making me feel guilty about not running enough. Vincent, Laurens, and Jonas, I immensely enjoyed our ~~holiday~~ summer school trip to Warsaw. Finally, I am convinced that nobody could ask for better officemates than Jessa, Vincent, and Maaike. I think we hit the sweet spot between messing around and working hard. What I especially like about our dynamic is that all of us appreciate that there is more to life than research, but also that sometimes exceptional efforts must be made when confronted with pesky conference deadlines. I sincerely thank you for your support and the fun memories we made over the years in office 02.144!

Of course, my KU Leuven colleagues are not the only people I wish to thank. Jurgen, I learned a lot from having you as my manager when I interned at Faceboook! My friends Daan, Ward, Kenneth, and Thomas, the "Kulak informatici", I am glad that I went through my bachelors and masters with you and that our friendship has continued beyond it. Thank you for occasionally giving me a glimpse of what the real world looks like outside of the ivory tower of academia. To my friends Simon, Thijs, and Liessa, I am thankful to have spent these last four years together with you in Leuven. I will remember our many DnD sessions fondly and hope we will be able to schedule many more in the future! Thank you to my parents, Erik and Katrien, for reminding me not to work too hard all the time and providing me with a home away from Leuven whenever it was needed. Thank you to my parents-in-law Lieve and Rik, for their everlasting support and help during many hectic periods.

A special thank you to my brother Wim, for his support and friendship over the years, and for always letting me stay over at his apartment in Kortrijk at any moment with no prior notice, even though I technically moved out over six years ago.

Above all, I wish to thank Ilse for her love and support. This dissertation would not have been possible without her, and not only because she initially spurred me on to follow my passion and undertake this adventure. She has always been there for me, even when juggling life, work, and building a home, often all at the same time. Ilse, I owe you everything and love you more each day.

Tom Decroos
Heverlee, September 2020

Abstract

Soccer analytics has seen an explosion of interest in the last decade. The success of data analysis in other sports has driven soccer clubs and other stakeholders in soccer to wonder if they could also deepen their understanding of the game by analyzing data and translate this deepened understanding into tangible results such as signing good players and winning matches. Consequently, more data than ever is being collected in soccer. One prominent data source is event stream data, which is collected by human annotators who watch video feeds of soccer matches through special annotation software and rigorously describe all on-the-ball actions performed on the pitch such as passes, dribbles, interceptions, tackles, and shots.

While event stream data is an incredibly rich data source, gleaning useful soccer insights from it has proven to be difficult in practice. One part of the problem is the inherent nature of soccer. It is a fluid sport that involves many complex interactions between players. Furthermore, soccer's low-scoring nature and susceptibility to chance make it hard to correlate player skill with match results. Another part of the problem is event stream data being hard to analyze in its raw form. Analysts typically have to deal with a number of issues such as parsing complicated data structures, adapting to vendor-specific terminologies, dealing with data sparsity, scaling to millions of data points, and incorporating domain knowledge.

These issues have motivated researchers to apply techniques from the field of artificial intelligence (AI) to event stream data, as these techniques are often intended to be used semi-autonomously on large and complicated data sets. Consequently, researchers have successfully used AI techniques such as classification, reinforcement learning, pattern mining, and network analysis to address soccer analytics tasks such as estimating shot quality, rating players, and detecting tactics.

However, the existing literature on learning from event stream data with AI

techniques shows a number of shortcomings. First, no efforts have been made to address the representational challenges of event stream data, severely obstructing the reproducibility of papers within the field. Second, no approach for valuing on-the-ball actions considers the full context in which actions are performed or recognizes the value of defensive actions such as tackles and clearances. Third, existing works that capture the playing style of teams and players have not sufficiently explored how to best model the locations and directions of actions; instead rudimentarily dividing the pitch into zones or ignoring the spatial component of event stream data altogether.

This dissertation makes three main contributions to the field of soccer analytics that address these shortcomings. First, to better represent event stream data, we define a new language that simplifies and unifies the data of different event stream data vendors, alleviating many data engineering challenges and encouraging the reproducibility of soccer analytics research. We also release a software package that can convert raw event stream data of popular vendors into this new language. Second, we propose a framework for assigning values to on-the-ball actions that, compared to simpler metrics and possession-based approaches, considers a more complete view of the context in which actions occur. Our framework formalizes the intuition that all actions in a match are performed with the intention of increasing the chance of scoring a goal and/or decreasing the chance of conceding a goal. The latter point is what allows our framework to recognize the value of defensive actions. Third, we introduce a number of approaches that express the playing style of teams and players based on where on the pitch they perform certain types of actions. Our approaches improve over earlier work by modelling the spatial component of event stream data in a data-driven manner using decomposition techniques such as non-negative matrix factorization and mixture models.

Beknopte samenvatting

Voetbalanalyse heeft het afgelopen decennium een explosie aan interesse gekend. Het succes van data-analyse in andere sporten heeft voetbalclubs en andere belanghebbenden in het voetbal ertoe aangezet zich af te vragen of ze hun inzichten in het spel kunnen verbeteren door data te analyseren en deze inzichten om te zetten in concrete resultaten zoals het aantrekken van goede spelers en het winnen van wedstrijden. Bijgevolg wordt er in het voetbal meer data dan ooit verzameld. Een prominent voorbeeld is event stream data. Deze data wordt verzameld door menselijke annotators die videofeeds van voetbalwedstrijden bekijken via speciale annotatiesoftware en alle acties die spelers uitvoeren op het veld - zoals passes, dribbels, intercepties, tackles en schoten - nauwkeurig beschrijven.

Hoewel event stream data een enorm rijke databron is, zijn er een aantal aspecten die het moeilijk maken om er nuttige voetbalinzichten uit te halen. Een deel van het probleem is dat voetbal een vloeiende sport is die veel complexe interacties tussen spelers bevat. Bovendien maken de weinige goals in voetbal en de invloed van geluk het moeilijk om de vaardigheid van spelers te correleren met wedstrijdresultaten. Een ander deel van het probleem is dat event stream data moeilijk te analyseren is in zijn ruwe vorm. Analisten moeten doorgaans omgaan met een aantal uitdagingen zoals het verwerken van ingewikkelde datastructuren, het aanpassen van hun code aan leverancierspecifieke terminologieën, het omgaan met ijle data, het opschalen naar miljoenen datapunten en het gebruiken van domeinkennis.

Deze uitdagingen hebben onderzoekers ertoe geleid om technieken uit het veld van artificiële intelligentie (AI) toe te passen op event stream data, aangezien deze technieken vaak ontwikkeld zijn met de bedoeling om semi-autonoom toegepast te worden op grote en gecompliceerde datasets. Onderzoekers hebben met succes AI-technieken zoals classificatie, patroonanalyse en netwerkanalyse gebruikt om voetbalanalysetaken zoals het inschatten van de kwaliteit van een schot, het beoordelen van spelers en het detecteren van tactieken aan te pakken.

Bestaande literatuur over het leren uit event stream data met AI-technieken vertoont echter een aantal tekortkomingen. Ten eerste zijn er geen pogingen ondernomen om de uitdagingen op het gebied van data-engineering van event stream data aan te pakken, waardoor de reproduceerbaarheid van papers binnen het veld wordt belemmerd. Ten tweede bestaan er geen methodes voor het waarderen van acties die rekening houden met de volledige context waarin acties worden uitgevoerd of die de waarde van defensieve acties zoals tackles en intercepties erkennen. Ten derde hebben bestaande werken niet voldoende onderzocht hoe de locaties en richtingen van acties het best kunnen worden gemodelleerd bij het analyseren van de speelstijl van teams en spelers. De meeste methodes delen het veld op in een aantal simpele zones of negeren de ruimtelijke component van event stream data volledig.

Dit proefschrift levert drie belangrijke bijdragen aan het veld van voetbalanalyse die deze tekortkomingen proberen aan te pakken. Ten eerste, om event stream data beter te kunnen analyseren, construeren we een nieuwe taal die de data van verschillende leveranciers van event stream data vereenvoudigt en verenigt. Deze taal verlicht veel uitdagingen op het gebied van data-engineering en moedigt de reproduceerbaarheid van voetbalanalyse-onderzoek aan. Ten tweede introduceren we een methode voor het toekennen van waarden aan acties die, vergeleken met eenvoudigere statistieken en methodes gebaseerd op balbezit, een vollediger beeld van de context waarin acties plaatsvinden beschouwt. De methode gebruikt een eenvoudige en elegante formule die de intuïtie formaliseert dat alle acties in een wedstrijd worden uitgevoerd met de bedoeling de kans op het scoren van een doelpunt te verhogen en / of de kans op een tegendoelpunt te verlagen. Deze laatstgenoemde eigenschap stelt de methode in staat om de waarde van defensieve acties te erkennen. Ten derde stellen we een aantal methodes voor die de speelstijl van teams en spelers uitdrukken op basis van waar op het veld ze bepaalde soorten acties uitvoeren. De verbetering van deze methodes ten opzichte van eerder werk is dat ze de ruimtelijke component van event stream data op een datagedreven manier modelleren met behulp van ontbindingstechnieken zoals niet-negatieve matrixfactorisatie en mixture modellen.

List of Abbreviations

AI Artificial Intelligence

DM Data Mining

DTW Dynamic Time Warping

EPV Expected Possession Value

EPL English Premier League

GAM Generalized Additive Model

GMM Gaussian Mixture Model

k-NN k-Nearest Neighbours

ML Machine Learning

MRR Mean Reciprocal Rank

NMF Non-negative Matrix Factorization

PCA Principal Component Analysis

SPADL Soccer Player Action Description Language

STARSS Spatio-Temporal Action Rating System for Soccer

VAEP Valuing Actions by Estimating Probabilities

xG Expected Goals

xT Expected Threat

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Soccer analytics has seen an explosion of interest in the last decade. The success of data analysis in other sports - with Billy Beane's Moneyball as the canonical example [103] - has driven soccer clubs and other stakeholders in soccer to wonder if they could also deepen their understanding of the game by analyzing data and translate this deepened understanding into tangible results such as signing good players and winning matches.

Simultaneously, soccer clubs are also collecting more data than ever. No longer limited by simple match sheet data, which provides high-level match information such as line-ups, substitutions, goals, and cards, soccer clubs are exploring richer data sources. Three particularly prominent types of data are:

Event Stream Data Event stream data is collected by human annotators who watch video feeds of soccer matches and describe all on-the-ball actions players perform on the pitch such as passes, dribbles, interceptions, tackles, and shots. Per event, annotators record properties such as the timestamp, location, type, (e.g., pass, dribble, shot), involved player, etc. Depending on the type of the action, additional information is annotated such as the end location of a pass or the outcome of a tackle. Event stream data is generated by companies such as Opta Sports [126], Wyscout [185], StatsBomb [165].

Optical Tracking Data Tracking data is a highly detailed data source that is collected by optical tracking camera systems that are physically installed in stadiums. These camera systems record the locations of all players and the ball at a high frequency of up to 100 Hz [4]. Examples of companies

that record tracking data are ChyronHego [36], STATS Perform [164], BallJames [10], Metrica Sports [120], and Second Spectrum [154].

Monitoring Data Monitoring data is collected from players by fitting them with wearable technology that contain sensors such as accelerometers, gyroscopes, heart rate monitors, and GPS. This data is collected by companies such as Catapult [31], STATSports [166], and many others [164, 71, 135, 148]. Monitoring data is also often augmented with questionnaire data (e.g., fatigue ratings, muscle soreness scores) and medical data from physical testings (e.g., flexibility, speed, maximum rate of oxygen consumption). While other data sources are only collected during matches, monitoring data is also collected off-match (e.g., rest periods and training sessions).

This data can be used to answer a multitude of questions relevant to soccer clubs such as:

1. "Should a player shoot or pass in a particular game situation?"
2. "Is a player who is currently part of the starting line-up contributing enough to their team or should they be benched?"
3. "What tactics does an opposing team commonly employ to score goals?"
4. "How well does a prospective player's profile fit what the club coach is looking for?"
5. "How should a club schedule its players' trainings in order to optimize their physical progress and minimize risk of injury?"

The first and second question are examples of hard decisions that are taken every day at soccer clubs. One way to take these decisions is based on objective information such as the values of specific actions. The third and fourth question have a less clear yes/no answer; they involve understanding the style of play of teams and players. The fifth question is related to health [92] and falls out of the scope of this dissertation.

To answer the first four questions related to value and style, event stream data is the most practical data source. Match sheet data is often too simple to glean non-trivial insights from. Tracking data is more detailed and extensive than event stream data in its description of a match, but it is also harder to analyze, more expensive, and only available for a restricted number of competitions. Finally, monitoring data is mostly relevant for analyzing the general health of athletes.

While event stream data is a more practical fit than other data sources to capture interesting soccer insights such as a player's skill or a team's tactics, this does not make the task easy. The reason why it is hard to learn concepts such as value and style from soccer event stream data is two-fold.

On the one hand, soccer as a sport has a number of characteristics that makes it inherently harder to analyze than many other sports. First, soccer is extremely fluid (i.e., it cannot be divided into clear phases like other sports such as baseball or tennis). Second, soccer involves many complex interactions such as passes between teammates and tackles of opponents [115]. Third, players have particularly varied roles within their team and these roles are constantly in flux depending on factors such as the time, the score, and where on the pitch the ball is currently in play [109]. Finally, because soccer is low-scoring and consequently more luck-based [136], player skill and team skill are often not as clearly correlated with match results as they are in other sports.[1]

On the other hand, event stream data is plagued by a number of challenges that make it hard to analyze. One data engineering challenge is that event stream data was not designed specifically for data analysis, but also for other use cases such as informing broadcasters. Another data engineering challenge is that each vendor uses their own unique terminology and data structure to describe events, limiting the wider applicability of analysis efforts. Finally, event stream data often contains optional information snippets that provide additional information on events but are difficult to automatically parse.

Furthermore, beyond these representational challenges, the analysis is hindered by other challenges more inherent to event stream data. First, events have both discrete (e.g., type, outcome) and continuous (e.g., time, location) properties. Most analysis techniques were designed to work exclusively on either discrete or continuous data and handle this mix of domains poorly. Second, there is very little exact repetition in sequences of game play. That is, the same players rarely perform the same actions in the same order in the same location. This makes generalizing over multiple data points difficult as it is often unclear when two actions should be considered to be similar or not. Third, event stream data only gives a partial picture of game play. A crucial piece of context that is missing from event stream data is the locations of the other 21 players on the field that are currently not in possession of the ball.

Many of the challenges listed above are non-trivial to address. Since most soccer clubs focus more on winning their next match than on being good at data

[1] While these characteristics make soccer hard to analyze, they are also part of the appeal of the sport. For example, the low-scoring and luck-based nature of soccer make it a great spectator sport, as there is always a chance that an underdog can turn a match around in its final minutes.

analysis, they often lack the expertise needed to properly leverage event stream data. This means that in practice soccer clubs usually default to tried and tested approaches for their soccer research such as looking at simple metrics like goals and assists when scouting new players or employing video analysts who watch many hours of match footage to prepare for an upcoming match.

However, tools that can (partially) replicate, complement, or even transcend human soccer insights have the potential to be extremely valuable to soccer clubs. For example, scouts at soccer clubs are often flooded by the sheer volume of prospective players. An algorithm that assesses the value of players from event stream data can help funnel this set of prospective players to the most promising top-10 that are worthy of the scout's further attention. Similarly, video analysts can only watch so much match footage in a single day to help their team prepare for an upcoming match. An algorithm that can capture the style of players and teams from event stream data can empower video analysts to produce more substantive opponent reports that help teams prepare for their upcoming matches.

Analysis techniques from the research field of artificial intelligence (AI) are a natural fit to build these tools as these techniques are often intended to be used semi-autonomously on large and complicated data sets. Hence, there has been an explosion of interest in applying AI techniques to automatically analyze event stream data for a variety of tasks such as estimating shot quality [108, 27, 88, 6], rating players [127, 51, 159], and detecting tactics [174, 175, 83]. While tremendous progress has been made in the field of soccer analytics in the last years, we have identified three important shortcomings in the literature that respectively concern (1) representing event stream data, (2) valuing on-the-ball actions, and (3) capturing the playing style of teams and players.

First, no efforts have been made to address the representational and data engineering challenges of event stream data (e.g., complicated data structures and vendor-specific terminologies). Consequently, researchers often spend considerable efforts writing one-off preprocessing scripts (which are never publicly released) to extract the information relevant to their analysis. Additionally, their analysis results are then tied to one specific vendor's data, which severely obstructs the reproducibility of all papers within the field of soccer analytics.

Second, most approaches that value on-the-ball actions do not leverage event stream data to its fullest extent. Either they only value shots [108, 27, 6, 110] - which account for less than 2% of a player's actions - or they only consider an action's location [121, 159, 178, 84, 20] and throw away most of the action's context (e.g., the goal difference, time remaining, speed of play). Additionally, these approaches only consider an action's offensive value (i.e., its effect on

the likelihood of scoring a goal) and thus are unable to recognize the value of defensive actions that reduce the risk of conceding a goal (e.g., tackles, clearances).

Third, existing works that capture the playing style of teams and players have not explored how to properly model the spatial component of event stream data (i.e., the locations and directions of actions). In practice, they either rudimentarily divide the pitch into simple zones to model an action's location [37, 174, 175] or ignore the spatial component altogether [12, 132, 83]. However, the locations and directions of actions are a fundamental aspect of playing style. For example, these properties are needed to capture the difference between players who play probing forward passes deep into the opponent's half and players who give safer lateral passes to the flanks.

1.1 Dissertation Statement

In this dissertation we use techniques from the field of artificial intelligence to learn value and style from soccer event stream data. Here, *learning value* means assigning values to individual on-the-ball actions (which can be aggregated to also value players and teams), while *learning style* means capturing the playing style of teams and players. Concretely, we aim to address the aforementioned gaps in the existing literature and answer the following questions.

1. How can event stream data be simplified and unified between different vendors?
2. Is there a way to automatically value on-the-ball actions while taking into account more context than just their locations as well as consider their defensive value?
3. Can AI techniques improve how the spatial component of event stream data is modeled when capturing the playing style of teams and players?

1.2 Contributions and Structure of the Dissertation

This section summarizes the structure of this dissertation and its three contributions.

We provide background information in Chapter 2, introduce new representations for event stream data in Chapter 3, value on-the-ball actions in Chapters 4-5,

capture the playing style of teams and players in Chapters 6-8, and finally offer our conclusions in Chapter 9.

1.2.1 Background

Chapter 2 details the necessary background to understand the subsequent chapters. It first introduces the game of soccer and its three most common data sources. Next, the chapter discusses various techniques from the research field of artificial intelligence that are used in later chapters. Finally, the chapter provides a bird's-eye view on the soccer analytics research field. Parts of this chapter are based on the following publication:

Van Haaren, J., Robberechts, P., Decroos, T., Bransen, L., and Davis, J. Analysing Performance and Playing Style using Ball Event Data. *Football Analytics: Now and Beyond. A Deep Dive into the Current State of Advanced Data Analytics.* (2019), 36–47

1.2.2 Contribution 1: Representing Event Stream Data

Chapter 3 discusses how to best represent event stream data. It first touches on the data engineering challenges that arise when processing event stream data in its raw form. Next, the chapter introduces **SPADL** and **Atomic-SPADL**, which are two new languages to describe on-the-ball player actions that simplify and unify the event stream data of different vendors. Finally, the chapter discusses a number of challenges inherent to event stream data that arise when applying techniques from the field of artificial intelligence. Implementations of SPADL and Atomic-SPADL are publicly available at `https://github.com/ML-KULeuven/socceraction`. Parts of this chapter are based on the following publication and blog post:

Decroos, T., Bransen, L., Van Haaren, J., and Davis, J. Actions Speak Louder than Goals: Valuing Player Actions in Soccer. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2019), KDD '19, ACM, pp. 1851–1861

Decroos, T., Robberechts, P., and Davis, J. Introducing Atomic-SPADL: A New Way to Represent Event Stream Data. DTAI Sports Analytics Lab Blog, 2020. `https://dtai.cs.kuleuven.be/sports/blog/introducing-atomic-spadl-a-new-way-to-represent-event-stream-data`

1.2.3 Contribution 2: Learning Value

This part of the dissertation deals with learning *value* from soccer event stream data. It attempts to answer the following question: "What is the value of an action in a soccer game?" In essence, we value an action by estimating its expected impact on the scoreline.

Chapter 4 introduces **STARSS**, which is a technique to first assign a value to a phase in a soccer game based on outcomes of similar phases and then distribute the value of that phase over its constituent actions. This chapter is based on the following publication:

Decroos, T., Van Haaren, J., Dzyuba, V., and Davis, J. STARSS: A Spatio-Temporal Action Rating System for Soccer. In *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2017 workshop* (2017)

Chapter 5 introduces **VAEP**, which is a framework for valuing all on-the-ball actions in soccer using a probabilistic classifier. VAEP values an action by assessing the change in probability of a team scoring and conceding in the near future as a result of that action moving the game from one game state to the other. An implementation of VAEP is publicly available at `https://github.com/ML-KULeuven/socceraction`. This chapter is based on the following publications:

Decroos, T., Bransen, L., Van Haaren, J., and Davis, J. Actions Speak Louder than Goals: Valuing Player Actions in Soccer. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2019), KDD '19, ACM, pp. 1851–1861

Decroos, T., and Davis, J. Interpretable Prediction of Goals in Soccer. In *AAAI 2020 Workshop on Artificial Intelligence in Team Sports* (2020)

Decroos, T., Bransen, L., Van Haaren, J., and Davis, J. VAEP: An Objective Approach to Valuing On-the-Ball Actions in Soccer (Extended Abstract). In *Proceedings of the 29th International Joint Conference on Artificial Intelligence* (2020), AAAI Press

1.2.4 Contribution 3: Learning Style

This part of the dissertation deals with learning *style* from soccer event stream data. In essence, we attempt to capture the playing style of teams and players by expressing where on the pitch they tend to perform certain types of actions.

Chapter 6 introduces **Tactics Discovery**: an approach to mine patterns in soccer event stream data and characterize teams by their most frequent

patterns. More specifically, the chapter studies how event stream data can best be converted into a format compatible with popular sequential pattern mining algorithms. This chapter is based on the following publication:

Decroos, T., Van Haaren, J., and Davis, J. Automatic Discovery of Tactics in Spatio-Temporal Soccer Match Data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2018), pp. 223–232

Chapter 7 introduces **Player Vectors**: a method to describe players' playing style in a small vector by factorizing the spatial heatmaps of players' passes, dribbles, crosses, and shots. This chapter is based on the following publication:

Decroos, T., and Davis, J. Player Vectors: Characterizing Soccer Players' Playing Style from Match Event Streams. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (2019), Springer

In both Chapters 6 and 7, the sparsity of event stream data (i.e., players rarely perform the same action in the same location more than once) is identified as a major challenge. **Chapter 8** introduces an approach to overcome this sparsity: **SoccerMix**. Soccermix is a mixture-models-based approach to soft cluster actions in event stream data based on their type, location, and resulting ball direction. Players and teams can then be represented in terms of which clusters of actions they perform most often. An implementation of SoccerMix is publicly available at `https://github.com/ML-KULeuven/soccermix`. This chapter is based on the following publication:

Decroos, T., Van Roy, M., and Davis, J. SoccerMix: Representing Soccer Actions with Mixture Models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (2020), Springer

1.2.5 Conclusion

Chapter 9 summarizes the contributions presented in this dissertation, discusses some general lessons learned from the perspective of both soccer analytics and artificial intelligence, and provides possible directions for future work.

1.3 Other Research Conducted

To present a coherent story, this dissertation focuses on six research projects that produced first-author publications, were conducted exclusively over the

course of my PhD, and concern learning either value or style from soccer event stream data. However, these six research projects are only a subset of my contributions during my PhD. This section gives a short summary of my other publications.

Peer-reviewed Conference Papers

Decroos, T., Dzyuba, V., Van Haaren, J., and Davis, J. Predicting Soccer Highlights from Spatio-Temporal Match Event Streams. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence* (2017), pp. 1302–1308

This paper presents the POGBA algorithm [47] for automatically predicting highlights in soccer matches from event stream data. The paper leverages the intuition that highlights in soccer matches often concern situations with a higher chance of scoring than usual. POGBA first estimates the probability of a game state leading to a shot, and then that of a possible shot resulting in a goal. Finally, a peak detection algorithm is applied to the predicted goal probabilities to detect the highlights of a soccer game. We compared the highlights predicted by POGBA with those of real highlights collected by a sports website and found that POGBA outperforms a number of baseline algorithms in terms of precision and recall. This paper is not included in this PhD dissertation because the research for it was mostly performed as part of my Master's thesis at KU Leuven. However, this paper served as the inspiration for later action valuing systems that are discussed in this dissertation such as STARSS [51] and VAEP [42].

Decroos, T., Schütte, K., De Beéck, T. O., Vanwanseele, B., and Davis, J. AMIE: Automatic Monitoring of Indoor Exercises. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (2018), Springer, pp. 424–439

This paper presents AMIE [49], a machine learning approach for automatically monitoring the execution of commonly used rehabilitation exercises using the Kinect, a lowcost and portable 3D-camera system. This paper contributed with respect to existing work by being one of the first to comprehensively detail the collected data set, describe the used classification system in depth, report quantitative results about the approach's performance, and publicly release both the collected data set and used software tools. AMIE was evaluated on a data set of ten test subjects who each performed six sets of ten repetitions of three commonly used rehabilitation exercises (i.e., squat, forward lunge and side lunge). AMIE detected the type of exercise with 99% accuracy and the type of mistake that was made with 73% accuracy.

Peer-reviewed Workshop Papers

Vroonen, R., Decroos, T., Van Haaren, J., and Davis, J. Predicting the Potential of Professional Soccer Players. In *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2017 workshop* (2017), vol. 1971, pp. 1–10

This paper presents APROPOS [181], a system to predict the potential of professional soccer players. We developed and evaluated the APROPOS projection system which predicts a player's future skill levels using a k-nearest neighbors approach. We introduced metrics to measure the similarities between players and methods to predict a player's future skill level from these similarities. When applied to the expert ratings of player skills on the `SoFIFA.com` website, our best model obtained a mean absolute error of 2.15 on skills rated from 0 to 100.

Geerts, A., Decroos, T., and Davis, J. Characterizing Soccer Players' Playing Style from Match Event Streams. In *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2018 workshop* (2018), vol. 2284, Springer, pp. 115–126

In this paper, we construct player vectors by transforming sets of events from match stream event data to fixed-size feature vectors using non-negative matrix factorization [72]. These player vectors offer a complete view of a player's playing style, are constructed in a purely data-driven manner, are human-interpretable and can be used in machine learning systems such as clustering and nearest neighbor analysis. We presented a number of use cases in player type discovery, scouting, and player development to show the potential of this approach. This workshop paper is the precursor to our paper on player vectors that was published at the 2019 ECML/PKDD conference [44] and is discussed in Chapter 7.

Verstraete, K., Decroos, T., Coussement, B., Vannieuwenhoven, N., and Davis, J. Analyzing Soccer Players' Skill Ratings Over Time Using Tensor-Based Methods. In *Proceedings of the 6th International Workshop on Machine Learning and Data Mining for Sports Analytics at ECML/PKDD 2019* (2019), Springer

In this paper, we made three contributions to the field of soccer analytics. First, we highlighted a variety of challenges that arose while processing player skill data from the `SoFIFA.com` website. Second, we explored the SoFIFA data tensor using the canonical polyadic decomposition (CPD) in order to extract interpretable latent structures. We showed how these latent structures group together related skills, how they evolve as players age, and how each player

can be summarized as a linear combination of these latent structures. Third, we employed the Tucker decomposition of a tensor in order to project how a specific player's skills will evolve as he ages.

Van Roy, M., Robberechts, P., Decroos, T., and Davis, J. Valuing On-the-Ball Actions in Soccer: A Critical Comparison of xT and VAEP. In *AAAI 2020 Workshop on Artificial Intelligence in Team Sports* (2020)

Expected Threat (xT) [159] and Valuing Actions by Estimating Probabilities (VAEP) [42] are two prominent approaches for the important task of valuing actions in a soccer match. In this paper, we performed a critical comparison of these two approaches, conceptually, qualitatively, and quantitatively. Key differences arise in how each approach represents the game state and what actions are valued. These lead to interesting differences such as VAEP better capturing the risk-reward trade-off of actions and xT being more robust. Importantly, both metrics produce rankings that deviate from those produced by considering traditional metrics (goals or assists). Hence, they provide additional insights into player performance.

Chapter 2

Background

In this chapter, we briefly discuss the sport of soccer, the three most prominent types of data collected in soccer matches, touch upon some techniques from Artificial Intelligence that will be used in this dissertation, and provide an overview of the current state of the art in soccer analytics. Sections 2.2, 2.4.2 and 2.4.4 in this chapter are based on the following publication [176]:

Van Haaren, J., Robberechts, P., Decroos, T., Bransen, L., and Davis, J. Analysing Performance and Playing Style using Ball Event Data. *Football Analytics: Now and Beyond. A Deep Dive into the Current State of Advanced Data Analytics.* (2019), 36–47

2.1 Soccer

Soccer is a ball sport played between two teams of eleven players on a grass pitch (Figure 2.1). The objective in soccer is to score goals by getting the ball into the opponent's goal. A match is won by the team that scores the most goals. Each team consists of ten outfield players and one goalkeeper. Outfield players mostly use their feet and head to move the ball from one place on the pitch to another, while goalkeepers are the only players that are allowed to touch the ball with their hands and arms in a designated area of the pitch.

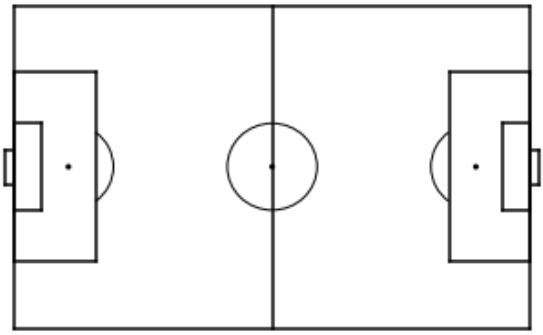


Figure 2.1: A 105m x 68m soccer field.

2.2 Data Sources of Soccer Matches

Soccer clubs are collecting increasing amounts of data during matches. As shown in Figure 2.2, the technical data that is currently collected during matches can be broadly divided into three types: match sheet data, event stream data, and tracking data. Match sheet data provides high-level information about matches such as line-ups, substitutions, goals and cards. Event stream data describes the actions that players perform with the ball such as passes, dribbles, interceptions, tackles and shots. It is collected by human annotators who match video footage of matches and is sold by vendors such as Opta Sports [126], Wyscout [185], and StatsBomb [165]. Tracking data provides the exact spatial locations of the players and the ball at all times during a match. It is collected by optical tracking camera systems that are physically installed in stadiums and is sold by vendors such as ChyronHego [36], STATS Perform [164], BallJames [10], and Second Spectrum [154].

These three types of soccer data differ not only in their granularity but also in their availability. Match sheet data, which provides only limited high-level summaries about what happened in a soccer match, is available for virtually all professional and semi-professional soccer matches in the world. In contrast, tracking data, which provides the highest level of detail possible, is available only for a restricted number of competitions, mostly top divisions of the better-ranked European countries. Interestingly, event stream data, which attempts to strike a balance between the limited match sheet data and the detailed tracking data, has become widely available in recent years.

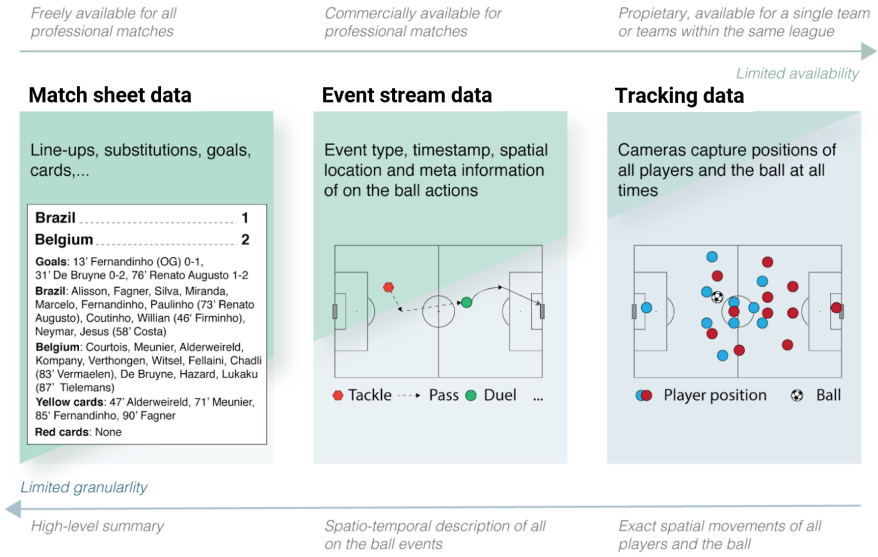


Figure 2.2: The data collected about soccer games can be broadly divided into three types: match sheet data, event stream data and optical tracking data. Event stream data strikes a balance between match sheet data and tracking data in terms of granularity and availability. © Pieter Robberechts [176].

Event stream data has become an increasingly popular data source for developing soccer analytics tools for four reasons. First, event stream data is easier to process and analyze than tracking data due to its smaller volume and easier structure. Second, event stream data is extremely interesting for player recruitment purposes due to its wide coverage of players in smaller competitions as well as important youth competitions. Third, event stream data can be purchased from specialized companies, whereas tracking data is typically only available to the teams in the league. Fourth, event stream data is becoming increasingly information rich. In addition to describing the actions with the ball, data collection companies have recently started registering the locations of the relevant players at the time of key events such as shots.

2.3 Artificial Intelligence Techniques

In this section we introduce a number of techniques from the field of artificial intelligence that will be applied to soccer event stream data in later chapters of

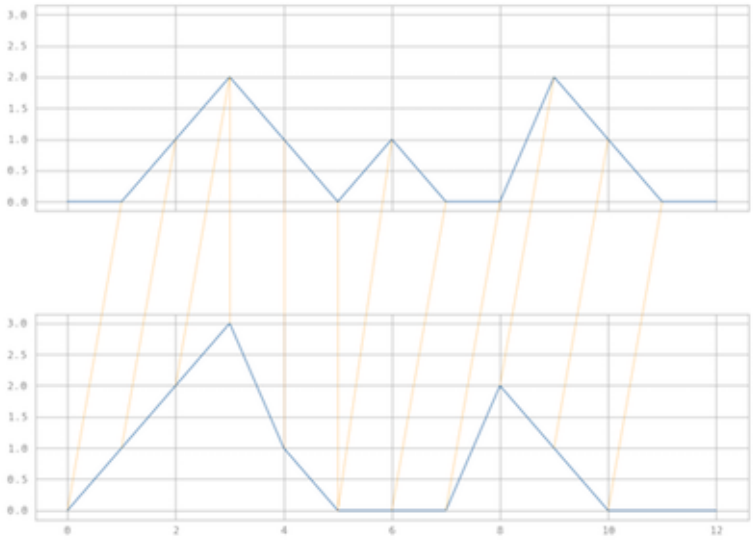


Figure 2.3: Illustration of how Dynamic Time Warping warps two sequences to match each other as closely as possible. © Wannes Meert [118].

this dissertation.

2.3.1 Dynamic Time Warping

Dynamic Time Warping (DTW) is a state-of-the-art distance measure for time-dependent sequences [123]. Unlike basic Euclidean distance, DTW does not require that sequences have the same length and is insensitive to minor mismatches between sequences, such as delays or shifts. Intuitively, the sequences are *warped* in a nonlinear fashion to match each other (Figure 2.3). Given two univariate numeric sequences $[a_1, \ldots, a_n]$ and $[b_1, \ldots, b_m]$, DTW typically employs a dynamic programming approach to evaluate the costs of all possible alignments. The DTW distance is the cost associated with the best alignment (i.e., the alignment with the lowest total cost). One way to compute the DTW-distance between two sequences is to construct a cost-matrix C where

$$C_{i,j} = |a_i - b_j| + \min\{C_{i-1,j}, C_{i-1,j-1}, C_{i,j-1}\} \tag{2.1}$$

and $C_{n,m}$ is the DTW-distance.

In the context of soccer analytics this technique is useful for identifying similar phases in event stream data as DTW can take into account that not all phases have the same length. Examples of this approach are given in Chapters 4 and 6.

2.3.2 Binary Probabilistic Classification

One of the most common tasks in artificial intelligence is learning a binary predictive function from data. The task can be described as follows: given a dataset D with instances of the form (x, y) with examples $x \in X$ and binary labels $y \in \{0, 1\}$, learn a function $f : X \to \{0, 1\}$ that can be used to predict the label of any instance x [17]. The instances $(x, 1)$ are commonly referred to as positive examples, while the instances $(x, 0)$ are commonly referred to as negative examples.

In the case of binary probabilistic classification, this task is generalized from learning a binary function $f : X \to \{0, 1\}$ to learning a probabilistic function $P : X \to [0, 1]$ that for any instance x can predict the probability of that instance having the positive class label 1. The most common classification models to learn such a probabilistic function are logistic regression [130], gradient boosted decision trees [35], and neural networks [5].

An example of this task in the context of soccer analytics is predicting (the probability of) whether a given shot will result in a goal or not based on some attributes that describe the shot (e.g., location). In this case the positive examples are scored goals, while the negative examples are shots that missed or were saved by the keeper. A model that addresses this specific task is often referred to as an expected-goals model as the model predicts how many goals can on average be expected from a particular shot. We discuss expected-goals models more in-depth in Section 2.4.2.

2.3.3 Sequential Pattern Mining

Pattern mining is a task within the field of artificial intelligence that focuses on discovering interesting, useful, and unexpected patterns in databases [68]. It emerged in the 1990s with the introduction of the Apriori algorithm [2], which was designed to find frequent item sets (i.e., groups of items frequently appearing together) in a database of customer transaction. For example, the Apriori algorithm can discover the pattern $\{milk, chocolate\_cookies\}$ in a retail store database, indicating that these products are often bought together by customers [68].

A downside of many popular pattern mining algorithms is that they ignore the sequential ordering of events. If such pattern mining algorithms are applied to data where the sequential ordering of events is relevant to the data, they may fail to discover important patterns or find nonsensical patterns that are useless in practice. To address this, the task of sequential pattern mining was proposed [3]. This task focuses on discovering interesting subsequences in a set of sequences of item sets, where the interestingness of a subsequence can be measured in terms of various criteria such as its occurrence, frequency, length, and profit [68].

Some of the most popular algorithms to discover sequential patterns in sequence databases are GSP [163], Spade [188], PrefixSpan [131], Spam [8], CM-Spam [66], and CM-Spade [66] (see Fourier-Viger et al.'s "A Survey on Sequential Pattern Mining" [68] for a complete overview). These algorithms take as input a set of sequences and a chosen minimum support threshold; and output the set of frequent sequential patterns. The difference between these algorithms lies not in their output, as there is always only one correct solution to a pattern mining query, but in their efficiency at discovering the patterns through specialized data structures and search strategies.

In Chapter 6, we apply sequential pattern mining algorithms to soccer event stream data by converting actions to item sets that detail their most important properties.

2.3.4 Non-Negative Matrix Factorization

Non-Negative Matrix Factorization (NMF) can be seen as the task of decomposing a non-negative matrix $V \in \mathbb{R}_+^{n \times m}$ into two non-negative matrix factors $W \in \mathbb{R}_+^{n \times k}$ and $H \in \mathbb{R}_+^{k \times m}$, where typically $k \ll \{m, n\}$, such that:

$$V \approx WH. \tag{2.2}$$

When V is a matrix that describes m examples with n features each, the k columns of W are often referred to as the "components" of the factorization, while the m columns in H are often referred to as the "encodings" of the m examples. The matrices W and H can be constructed by solving the following minimization problem:

$$\min_{W,H} \|V - WH\|_F \quad \text{subject to} \quad W, H \geq 0 \tag{2.3}$$

where $\|\|_F$ is the Frobenius norm of a matrix.

Compared to Principal Components Analysis (PCA) (which is similar to NMF but relaxes the non-negativity constraint), NMF has been shown to learn a

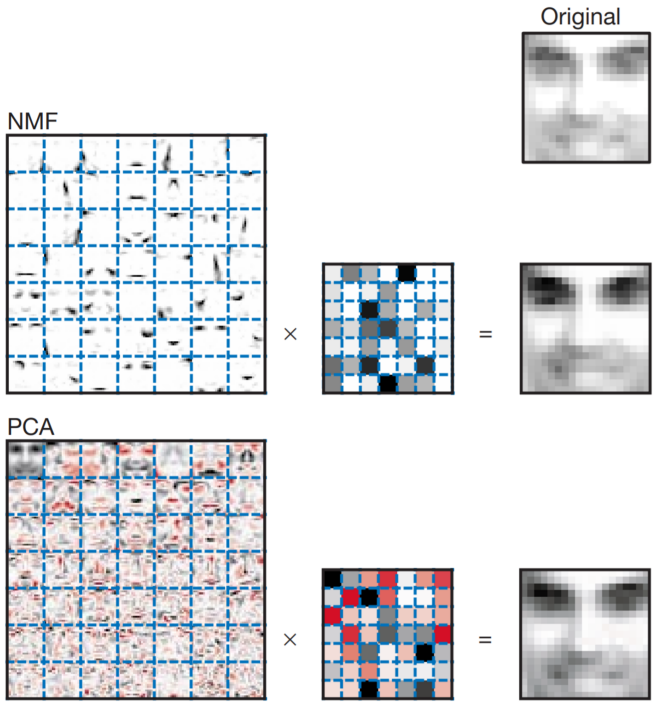


Figure 2.4: When decomposing a data set of face images, Non-Negative Matrix Factorization (NMF) learns a parts-based representation of faces, while Principal Component Analysis (PCA) learns a less interpretable holistic representation. Positive values are illustrated with black pixels and negative values with red pixels. Figure adapted from *"Learning the Parts of Objects by Non-Negative Matrix Factorization"* by Daniel Lee & Sebastian Seung [102].

more parts-based representation of a matrix with more sparse components and encodings [102] (Figure 2.4). Additionally, the components are also often more human-interpretable than those of other factorization techniques. This is because PCA's components are used in linear combinations that generally involve complex cancellations between positive and negative numbers, while NMF's components can only be added together and thus require individual meanings without posterior modifications [102].

In Chapter 7, we use NMF to decompose the heatmaps of players. Heatmaps are matrices that detail where on the pitch a player tends to perform certain

actions. They are constructed by laying a grid over the field and counting per grid cell how often the player performed an action in that grid cell.

2.3.5 Mixture Models

Mixture models are probabilistic models that assume that all data points of a data set are generated from a mixture of a finite number of distributions with unknown parameters [150]. Formally, a mixture model calculates the probability of generating observation x as:

$$p(x) = \sum_{j=1}^{k} \alpha_j \cdot F_j(x|\Theta_j) \tag{2.4}$$

where k is the number of components in the mixture model, α_j is the probability of the j^{th} component, and F_j is a probability distribution or density parameterized by Θ_j for the j^{th} component. Usually, mixture models are fitted to a data set using the well-known Expectation-Maximization algorithm [117]. Intuitively, mixture models can be thought of as a soft clustering variant of k-means clustering.

The most common type of distribution to be used with mixture models is a Gaussian distribution (Figure 2.5). However, mixture models can work with any distribution that has a probability density function F and update equations to estimate the parameter set Θ. One example of a more exotic distribution is the Von Mises distribution, which arises in the directional statistics literature [112, 13].

In Chapter 8, we use Gaussian mixture models to model the locations of actions and Von Mises mixture models to model the directions of actions.

2.4 Related Work

In this section we discuss literature related to this dissertation. We first discuss learning value from respectively match sheet data, event stream data, and tracking data. Next, we discuss learning style from event stream data and tracking data. The reason why we do not discuss learning style from match sheet data is because this is essentially impossible due to the extremely limited granularity of the information in match sheet data.

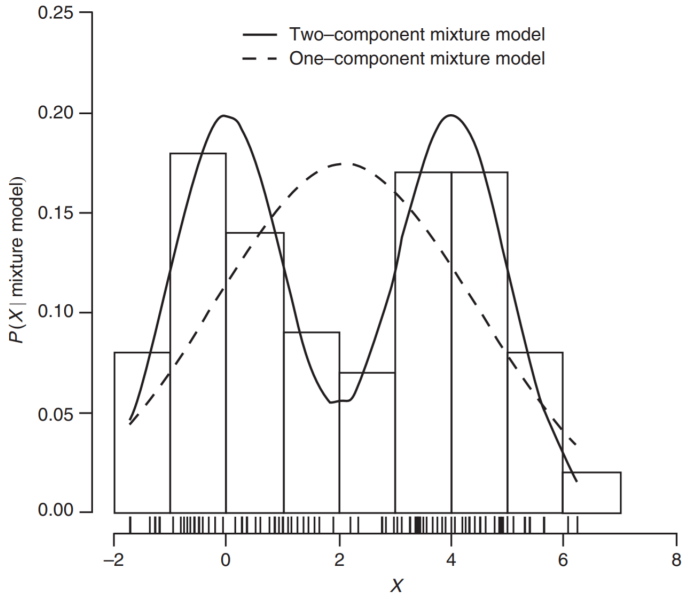


Figure 2.5: A one-component and two-component Gaussian Mixture Model (GMM) are fitted to a synthetic dataset of 100 univariate data points. The data points were sampled 50/50 from two univariate Gaussian distributions with parameter sets $\{\mu = 1, \sigma = 1\}$ and $\{\mu = 4, \sigma = 1\}$. The two-component GMM is clearly a better fit to the data set. Figure copied from *"Encyclopedia of Machine Learning"* by Claude Sammut & Geoffrey Webb [150].

2.4.1 Learning Value from Match Sheet Data

In this section, we discuss what can be learned from the information in match sheet data such as a match's outcome, the starting line-ups of teams, and the number of goals scored by each team in a match.

Rating Teams

The most basic piece of information that is always recorded for every match is its outcome: win, draw, or loss. Using nothing but match outcome, one can construct ratings for teams that reflect their relative strengths. One way to do is through the famous ELO rating system (originally introduced by Arpad Elo in 1978 to rate chess players [56]). Hvattum and Arntzen [87] show how on the one hand, the ELO rating system is a better fit to predict match outcomes

than simpler alternatives, but on the other hand, it is also vastly outperformed by predictive ratings based on bookmakers' odds.

Another piece of information that is usually recorded in match sheet data is a match's final score, i.e., the number of goals scored by each team. Constantinou and Fenton [39] introduced the pi-ratings for soccer teams, which, in addition to the outcomes of matches, also use the goal differences in the final scores of matches. They show how match predictions based on pi-ratings considerably outperform match predictions based on ELO ratings and also how pi-ratings can be used in a betting strategy that beats the bookmakers' odds.

Rating Players

Other pieces of information that are more extensive, but still often recorded in match sheet data, are player starting line-ups, substitutions, goals scored, and red cards handed out. This information can be leveraged to assign ratings based on match outcomes to not just teams, but players as well. For example, SciSports [152] leverages match sheet data to construct the SciSkill index [153], which calculates the influence that the player has on the team he plays for. Another way to rate players is through the plus-minus rating system which provides an answer to the question: "How does a team perform with a player, compared to without the player?" [93]. This rating system already existed in other sports, but Sæbø and Hvattum [149] were the first to introduce the plus-minus rating system for soccer players in 2015. Kharrat et al. [93] improve this plus-minus rating system by using not the actual outcome of the match, but the expected number of goals scored or the expected number of points gained from the match. However, a small caveat here is that the *expected* number of goals is computed not from match sheet data, but from shot characteristics usually found in event stream data.

Modelling Goal Scores

Rather than come up with rating systems for teams and players, statisticians have been more interested in ways to directly model and predict the final score of matches. Maher [111] kicked off this research area by modelling the number of goals team i scores in a match against team j as a Poisson distribution with mean $\alpha_i\beta_j$ where α_i represents the offensive strength of team i and β_j represents the defensive strength of team j. Dixon and Coles [54] improved Maher's model by (1) introducing a dependence between the number of goals scored by the two teams in a match and (2) using a time-decay function to make more recent match results have stronger effects on teams' estimated offensive and defensive

strengths. McHale and Scarf [114] observed that the correlation between the goals scored by the two teams in a match is not a static value, but depends on the difference in strength between both teams. More specifically, a larger gap in relative strength between two teams leads to a more negative correlation (i.e., if one team scores more goals, the other team usually scores fewer), but when both teams are evenly matched in strength, the correlation in scored goals gets close to zero (i.e., one team scoring more or fewer goals has no effect on the goals scored by the other team). Boshnakov et al. [19] introduced a new model called the bivariate Weibull count model and showed how it is a better fit to predict the number of goals scored by teams in soccer matches than the simpler Poisson-distribution-based model.

2.4.2 Learning Value from Event Stream Data

We will first discuss learning the value of teams and players based on box scores, which are simple count-based statistics queried from event stream data. Next, we introduce the well-known expected-goals (xG) metric to objectify shot quality. Finally, we show how researchers have extended the concept of expected goals to quantify passes and other actions.

Box Scores

The most common way to leverage event stream data is to query it for simple count-based statistics (commonly referred to as box scores) such as distance covered, number of assists, number of shots, pass percentage, etc. One way these statistics can be used to value players and teams is to assign a static value to each type of action and reward players with an action's associated value each time they perform an action of that type. This is the approach of the PlayeRank framework by Pappalardo et al. [127] and the EA SPORTS Player Performance Indicator by Mchale et al. [115], which was used until the end of the 2017/18 season by the English Premier League as its official player rating system [93]. Tiedemann et al. [171] use an operations research technique called Data Envelopment Analysis [33] to compute players' efficiency from their playing time, goals, assists, tackle win ratio, and pass completion ratio. Duch et al. [55] measure the performance of a player in a match by combining his individual box scores (e.g. pass success rate, frequency of on-target shots, number of ball recoveries) with his influence on his team quantified from a passing network.

While box scores provide some insights into the value and style of teams and players, they largely fail to account for the circumstances under which players performed their actions. For example, successfully completing a forward pass

deep into the opponent's half is both more difficult and more valuable than performing a backward pass in your own half without any pressure from the opponent whatsoever. The shortcomings of these simple count-based statistics have motivated researchers to propose several techniques to analyze actions in ways that account for the circumstances under which these actions were performed. These techniques are discussed in the next sections.

Expected Goals

Motivated by the fact that some shots are better than others, the expected-goals (xG) metric, which has received significant attention, attempts to quantify the quality of a goal-scoring opportunity [78]. To do so, an expected-goals model assigns a value between zero and one to each shot which represents the probability that the shot will result in a goal. The reason why expected-goals is an interesting tool to measure the offensive output of players is that the difference between good attackers and bad attackers lies mostly in how many high-quality scoring chances they create and not in their finishing rate, which rarely varies much between players [116, 7].

Building an expected-goals model requires addressing three key points:

- What game situations (e.g., penalty, shot from open play) should the expected-goals model distinguish among?
- What features of a shot are recorded in event stream data and are informative of the value of that shot?
- What type of learning model will best estimate the probability of a shot being converted into a goal?

The game situation in which a shot takes place affects the chance that it will result in a goal. For example, a penalty is more likely to result in a goal than a shot arising from a counter attack. While some approaches ignore the game situation [47], it is far more common to distinguish among shots arising from a small number of distinct game situations and build one expected-goals model for each one. Distinguishing among different situations better captures the context of the opportunity, which in turn results in more accurate models. Caley [27] considers six different shot types: shots from direct free kicks, shots following a dribble past the keeper, headed shots assisted by crosses, headed shots not assisted by crosses, non-headed shots assisted by crosses, and non-headed shots not assisted by crosses. In contrast, IJtsma [88] considers ten different shot types: open-play shots, open-play headers, penalties, shots from direct free

kicks, shots following an indirect free kick, shots arising from a corner, shots arising from a throw-in, shots following a rebound from a save, shots following a rebound from the woodwork, and shots during counters.

Defining features about a shot that are indicative of its quality is the key step in building an expected-goals model. Moreover, this step is where domain-knowledge about soccer comes into play. One category of features captures characteristics of the shot itself such as the location on the field, the distance to the goal, and the shot angle. Another category of features is based on considering the actions that preceded the shot. One such feature is whether the previous action was a completed take-on. By definition, this entails removing the on-the-ball defender from the play, which should make the shot easier. Similarly, whether the previous action was a through-ball also increases the quality of the shot. A huge number of features have been considered; both Caley's [27] and IJtsma's [88] blog posts provide detailed descriptions of possible features.

Ultimately, an expected-goals model needs to assign a real-valued number between zero and one to each shot. Typically, this is done by applying a machine learning model to a large historical data set of shots that for each shot contains both the observed features as well as its true result (i.e., goal or no goal). Any machine learning model that returns a probability is suitable, but typically either a logistic regression model [130] or gradient boosted tree model [35] is used in practice. A gradient boosted tree model can learn more fine-grained differences between shots than a logistic regression model [42], but this comes at the cost of usually not being able to interpret the model [45]. Additionally, Robberechts and Davis [142] showed how more complex learners also need more training data to produce accurate predictions.

Valuing Passes

The fact that expected-goals models only assign values to shots is problematic for two reasons. First, shots arise very infrequently, particularly compared to other types of actions such as passes, crosses or dribbles. Second, focusing on just the shot ignores the contributions that other players made to creating the attempt. This has motivated attempts to measure the impact on the scoreline of actions other than shots.

Since passes constitute the lion's share of the actions that happen in a match, passes have been a topic of particular interest in recent years. Most methods for valuing passes attempt to quantify a player's involvement in creating goal-scoring chances. Traditional pass-based metrics usually only reward passes that result in goals (i.e., assists). However, very few passes can be valued according to this criteria. Therefore, like for shots, methods for valuing passes typically

resort to measuring their expected impact on the scoreline based on historical observations. The traditional approach to address this task is to compute the difference between the value of possessing the ball in the location before the pass and the value of possessing the ball in the location after the pass. The differences between methods arise in how they determine the value of possessing the ball in a particular location.

At a high level, three approaches to determine the value of a pitch location have been proposed. The first approach assigns each pitch location an expected-goals value that is the xG value if a shot would have been attempted from that location [121]. This approach is particularly appropriate for passes close to the opponent's goal. However, passes further away from the goal will all have expected-goals values close to zero. The second approach is to determine the proportion of play outs from a pitch location that result into a goal within a given number of actions or seconds [20, 22, 84]. The challenges are to determine the optimal number of actions or seconds to look ahead and to measure the similarity between play outs. For instance, a pass during a slow build up will likely have a different value than a pass during a fast counter attack. The third approach is to distribute the reward of a possession sequence (e.g., a goal) starting in a given pitch location to its constituent passes [26]. The challenge is to decide on the optimal weighting scheme to distribute the credit across all the sequence's passes. Typically, passes at the end of the possession sequence receive more credit than passes at the start of the sequence.

Valuing Actions

Naturally, the interest in assessing passes led to the desire to evaluate more actions. While a number of different approaches have been proposed that evaluate a large number of actions, at a high-level they all function in the same way. When the team possesses the ball, each action within that possession is undertaken with the high-level objective of helping the player's team, either by increasing the chance that his team scores or decreasing the chance that the opposing team scores. The practical result of each (successful) on-the-ball action such as a pass or dribble is a change in the ball's location on the pitch. Intuitively, certain locations on the pitch are more valuable than others as they more readily lend themselves to generating goal scoring opportunities. For example, possessing the ball near the sideline close to the midfield line is generally not as threatening as possessing the ball in the center of the pitch just outside the opponent's penalty box. This suggests that the value of an action can be derived by simply taking the difference between the value of the ball's new location (i.e., where the ball ends up as a result of the action) and the ball's original location. The value of a location can be thought of as the

probability of a team scoring during its current possession given the location where the ball is currently possessed.

Metrics such as Valuing Actions by Estimating Probabilities (VAEP) [44] (Chapter 5), xG Added [110], Expected Threat (xT) [159], Attacking Contributions [186], Goals Added (g+) [100], Goal Impact Metric (GIM) [106] all exploit this type of reasoning. The differences arise in the technical modeling choices made to value the different actions. The first approach, introduced by Sarah Rudd in 2011 [147], was to view the possession as a Markov model. This involves discretizing the pitch into zones. Moreover, the model also assumes that previous actions will have no effect on how the rest of the possession will play out. That is, it does not differentiate between receiving the ball in the center of the pitch just outside the penalty area via a long through ball versus receiving the ball in the exact same location via a short, lateral pass. Another approach involves training a machine learning model to predict the probability that the team possessing the ball after an action will score in the near future (e.g., the next five to ten actions). This enables reasoning about the characteristics of past actions.

Once we have a way to assign values to individual actions, the most straightforward way to use them is to aggregate them into player ratings and team ratings. However, these action values can also serve as building blocks for other interesting applications such as investigating how players perform under mental pressure [21] and measuring the mutual chemistry between two players [23].

2.4.3 Learning Value from Tracking Data

Similarly to learning value from event stream data, researchers have first explored tracking-data based expected goals models. Then they moved on to metrics that value passes and even more general approaches that assign a value to any moment in a match. The latter approaches are often referred to as Expected Possession Value (EPV) models and can be used to produce stock tickers that visualize the evolution of the estimated danger over a window in a match.

Expected Goals

Tracking data can be used to build better expected-goals models, as tracking data contains crucial context that is missing from event stream data such as the positions of all the players. For example, Lucey et al. [108] built an expected goals model that, in addition to basic features such as the location of a shot

and the game phase in which it occurred (e.g., corner, free-kick, open-phase), also leverages features computed from the ten-second window preceding the shot such as defender proximity, interaction of surrounding players, and speed of play.

Valuing Passes

Similar to how tracking data can be used to value shots, it can also be used to value passes. Power et al. [137] estimate the risk and reward of passes using additional contextual features from tracking data such as the speed of relevant players (e.g., the pass-giver, the intended receiver, the nearest defender) and the angle and distance of the nearest defender towards the passing line. These features were initially proposed by Szczpenanski and McHale [168], but they had to build proxy features from event stream data as they did not have access to tracking data in their work. Power et al. [137] define the risk of a pass as the probability that the pass will not reach its intended receiver and the reward of a pass as the probability that the pass made will result in a shot within the next ten seconds. The authors note themselves that a drawback in their approach is the sparsity of shots.

To combat the sparsity of goals and shots as the intended end result of passes, other approaches have been explored to value passes using tracking data. Goes et al. [75] value passes by how much a pass will disrupt the defensive organization of the opposing team. Chawla et al. [34] had human observers rate a number of passes as *Good*, *Ok*, or *Bad* and were able to reproduce these ratings with 90% accuracy using a machine learning model. Their machine learning model leveraged simple geometric properties within tracking data such as the velocity of players and the ball as well as the Euclidean distances and angles between them.

Expected Possession Value

Similarly to event stream data, the interest in valuing passes led to the desire to create even more general metrics. However, as tracking data consists of location traces over time and contains no inherent definition of an action, researchers have focused on metrics that can compute a value for any frame, where a frame is the locations of the players and the ball at a specific moment during a match.

Link et al. [104] introduce the concept of *dangerousity*, which is the probability of a goal being scored and can be computed for every point in time at which a player is in possession of the ball. *Dangerousity* is computed from four

components: (1) the goal-scoring danger of the ball's current zone, (2) the extent to which a player can implement his tactical intention, (3) the pressure put on the player currently in possession of the ball by the opponent, and (4) the probability of successfully maintaining possession of the ball in the near future.

Spearman [161] constructed a probabilistic physics-based model that computes the probability that a player not currently in possession of the ball will score. This probability is called the off-ball scoring opportunity (OBSO) and is computed only from the instantaneous game state. At the core of the approach lies a pitch control field model [162] which quantifies the potential regions of control in the near future and can be used to reward players for creating space in a scoring location even if the ball is never delivered to them.

Fernández et al. [60] compute the Expected Possession Value (EPV), which is a value in the $[-1, 1]$ range that can be computed at any moment during a match and represents the likelihood of a possession ending in a goal for the attacking team (1) or a goal for the defending team (-1). Their approach is similar in spirit to the original EPV approach for basketball [32], but differs in the technical modelling choices made to account for the nuances of soccer such as looser notions of possession and the ability of passes to happen at any location on the pitch.

Dick and Brefeld [53] proposed a deep reinforcement learning approach to learn values of multi-player positionings. The biggest difference to other works is that the approach of Dick and Brefeld does not require any prior knowledge of the domain of soccer. Their approach successfully learns meaningful and interpretable valuations and can be seen as a first step towards computational tactics.

2.4.4 Learning Style from Event Stream Data

A recurring concept when discussing soccer is the style of play, which is applicable on both the player and team level. On the player level, this refers to a player's behaviour on the pitch. For example, both Messi and Ronaldo are great players, but each one approaches the game in a different way. On the team level, this manifests itself in terms of the tactics the team employs. Naturally, a player's behaviour is inherently linked to his team's tactics. There is substantial value in gaining a better understanding of playing style as this can be leveraged in areas such as player scouting and match preparation. Simple descriptive statistics such as pass percentage or shot count are usually insufficient to capture playing style. Hence, there has been an explosion of interest in applying automated techniques to try to glean insights into both player and team behaviours.

Analyzing Player Behavior

Analyzing player behavior essentially boils down to summarizing the player's playing style in a way that is both human-interpretable and suitable for data analysis. Typically, the goal is to construct a fingerprint of a player's playing style which captures distinguishing characteristics of a player's behavior such as which types of actions a player tends to perform and where or what types of gameplay patterns he tends to participate in. There are currently two distinct ways to do this: location-based and interaction-based. Location-based approaches consider the locations and action type information in an event. Typically, they then attempt to summarize which locations a player prefers to occupy on the field and what actions he tends to perform in each of these areas [44, 82] (Chapters 7 and 8). Interaction-based approaches consider the players involved in an event. They focus on detecting player interaction patterns (e.g., a one-two pass where player A passes the ball to player B who immediately returns the ball to player A) and then for each player count how often they are involved in various patterns [12, 83, 182].

Analyzing Team Tactics

As interesting as analyzing individual player behavior is, most works tend to focus on team tactics, which can also add substantial value to the workflow of the tactical decision maker, i.e., the coach. Analyzing team tactics is in some ways easier than analyzing player behavior because there is more event stream data available per team than per player. However, it can also be more difficult as team tactics are more complex than player behavior. Usually, the more complex the concept you are trying to infer from your data, the harder it is on a technical level to infer it successfully.

There are roughly three different ways to analyze team tactics. The first way is to summarize a team's playing style in a number of features (usually simple counts such as event counts or location occurrence counts) and then cluster teams based on those features [26, 81]. The second way is to extract patterns from the data using a pattern mining algorithm [50]. Often, the biggest challenge in this approach is getting the representation of the data right so that the patterns extracted by the pattern mining algorithm are informative, intuitive and make sense to the end user. The third way is to attempt to model the complete behavior of the team in a network-based approach such as a passing network or a Markov network [37, 132, 182].

An important parameter of all three approaches is what information from events to consider. There are three main categories of information per event:

(1) the player(s) involved, (2) the location, and (3) the type of the event. Some approaches focus only on one of them. For example, Bekkers and Dabadghao [12] mine patterns that focus exclusively on involved players and Peña [132] summarizes team's playing style purely based on event types.

However, the insights that can be gained from a single category of information is limited. Hence, most approaches combine two categories. Combining these categories is often technically challenging and the biggest technical contribution is in the way the approach combines them. Wang et al. [182] and Cintia et al. [37] analyze both the involved players and the locations of events by applying a network-based approach to model the transitions between players and zones on the pitch. Bojinov et al. [18] and Van Haaren et al. [174] attempt to analyze team tactics using both location and event type. Typically, the locations of events are discretized by manually dividing the pitch into simple zones.

Van Haaren et al. [175] and our own contributions in Chapter 6 are even more ambitious and attempt to detect patterns that capture all three categories of information at the same time. While interesting, the results are so far more proof-of-concepts than useful in practice, because as mentioned earlier in this section, the more complex the concept you are trying to infer from your data, the harder it is technically to infer it successfully.

2.4.5 Learning Style from Tracking Data

In the context of tracking data, playing style on the level of players relates to how players fit into their team and how they interact with their team mates. Playing style on the level of teams mostly concerns how teams attempt to gain control of the pitch. In this section, we discuss identifying player roles in soccer based on tracking data (which is considered to be mostly a solved problem [16]) and the more recent research efforts on pitch control models.

Player Roles

One characteristic of soccer that makes tracking data hard to analyze is the fact that players have varied roles within their team and that these roles are constantly in flux depending on factors such as the time, the score, and where on the pitch the ball is currently in play [115]. To combat this challenge, Bialkowski et al. [16] introduced a role-based representation of soccer teams and an approach to dynamically update each player's relative role in each frame. The approach discovers player roles using a minimum entropy data partitioning method. This role-based representation of teams and players has many interesting applications

such as detecting and visualizing team formations [183], identifying teams from anonymized tracking data [15], and illustrating how teams tend to play higher up the field when playing at home than away [14].

Pitch Control

In a soccer match, teams are constantly fighting for control over the pitch. Tracking data is an especially good fit to analyze this process as it records the locations of all players on the field. Castellano et al. [30] analyzed simple spatial characteristics of a team such as the length, width, and surface of the bounding box over the locations of the teams' players. Their main finding was that stronger teams tend to play more spread out over the field than weaker teams.

A reoccurring concept in pitch control models is the *dominant region* of a player. The dominant region is the region in which that player can arrive earlier than any other player [169]. The larger the total dominant region of a team's players, the more that team can be considered to be in control of the pitch. The most common way to compute dominant regions is through the use of Voronoi diagrams [169, 70, 80, 65, 119]. Fonseca et al. [65] found that attackers tend to have larger dominant regions than defenders.

Recently, Voronoi-based dominant regions have been criticized as not being realistic [162, 59]; reasons being that Voronoi-based dominant regions produce hard borders and do not fully take into account important characteristics of players such as their typical behavior and current velocity. Gudmundsson and Wolle [80] augmented the underlying motion models of the Voronoi-based approach with personalized motion models based on players' historical movement data. Spearman [162] introduced a new passing model for soccer that is based on the physical concepts of interception and control time. The model uses a statistical framework to accurately predict the precise receiver in the majority of passes using only the game state at the point the ball is kicked. Using his model, Spearman can quantify the control a team exerts over any arbitrary region on the pitch. Fernández et al. [59] provide a pitch control model that fixes the shortcomings of Voronoi-based models by (1) taking into account key information such as the velocity of players and (2) rather than having hard borders between dominant regions, introducing the idea of a soft surface of control where for a given location on the field, nearby players have a certain level of influence.

Chapter 3

Representing Event Stream Data

In this chapter, we discuss event stream data and its challenges. We first describe the raw event stream data provided by three different vendors and the engineering challenges related to processing it. Next, we introduce two ways to represent event stream data: SPADL, an event stream data format that simplifies and unifies data from different vendors, and Atomic-SPADL, a format that transforms the initiation and completion of some actions into separate "atomic" events. Finally, we discuss nine challenges inherent to event stream data that arise when applying Artificial Intelligence techniques.

Some parts in this chapter are based on the following publication [42] and blog post [48]:

Decroos, T., Bransen, L., Van Haaren, J., and Davis, J. Actions Speak Louder than Goals: Valuing Player Actions in Soccer. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2019), KDD '19, ACM, pp. 1851–1861

Decroos, T., Robberechts, P., and Davis, J. Introducing Atomic-SPADL: A New Way to Represent Event Stream Data. DTAI Sports Analyics Lab Blog, 2020. `https://dtai.cs.kuleuven.be/sports/blog/introducing-atomic-spadl-a-new-way-to-represent-event-stream-data`

3.1 Raw Event Stream Data

Event stream data describes all notable events during a soccer game with a focus on players' on-the-ball actions. This data is collected by human annotators who watch video feeds of soccer games through specialized annotation software. Per event, annotators record attributes such as the timestamp, location, type (e.g., pass, dribble, shot), involved player, etc. Depending on the type of the action, the annotator also collects additional information such as the end location of a pass or the outcome of a tackle.

In this section, we briefly discuss three popular vendors that supplied the event stream data used in this dissertation and highlight three key engineering challenges that occur when processing their event stream data.

3.1.1 Event Stream Data Vendors

Three prominent vendors of event stream data are:

Opta Opta Sports [126] is a British sports analytics company that currently provides data for 30 sports in 70 countries, with clients ranging from leagues to broadcasters and betting websites [113, 126]. The company was founded in 1996, and acquired by Perform Group in 2013. In turn, it was acquired by Vista Equity Partners and STATS LLC in 2019 [90]. Opta Sports started collecting event stream data in 2006.

Wyscout Wyscout [185] is an Italian sports analytics company that was founded in 2004 and started collecting event stream data for soccer in 2013 [91, 185]. As of 2020, the company has over 450,000 players, 220,000 games, and 250 competitions in their database. In 2019, they publicly released event stream data of the 2016 UEFA Euro cup, the 2018 FIFA world cup, and the 2017/18 season of the first soccer divisions in England, Spain, Germany, Italy, and France [128].

StatsBomb StatsBomb [165] is a British soccer analytics company that covers over 40 soccer leagues worldwide and aims to provide the most accurate and granular data on the market [165]. The company was founded in 2016 and started collecting event stream data for soccer in 2017. In 2018, they publicly released soccer event stream data of the 2018 FIFA world cup.[1] Since then, they have also publicly released event stream data of the FA Women's Super League and all games of Lionel Messi's career.

[1] `https://github.com/statsbomb/open-data`

```
{"@attributes":{
    "id":1673965276, "event_id":857, "type_id":30,
    "period_id":2, "min":93, "sec":23,
    "player_id":0, "team_id":413, "outcome":1,
    "x":0, "y":0, "assist":0, "keypass":0,
    "timestamp":"2017-09-24T17:09:45.000",
    "TimeStamp":{ "locale":"2017-09-24T17:48:45.000Z",
                  "utc":"2017-09-24T17:48:45.000Z"},
    "version":{ "lo":1577123840, "hi":2222260661}},
 "Q":[ {"@attributes":{ "id":1209383978,
                        "qualifier_id":57,
                        "value":""}},
       {"@attributes":{ "id":1904522716,
                        "qualifier_id":209,
                        "value":""}}]}
```


Figure 3.1: An event provided by Opta Sports

Additionally, StatsBomb has handed out event stream data of the 2017/18 English Premier League at promotional events.

An example of an event in a soccer game recorded by Opta, Wyscout, and StatsBomb are available in respectively Figures 3.1, 3.2, and 3.3. For efficiency reasons, all vendors use identifiers to represent entities such as players, teams, event types, tags, and qualifiers. Analyzing event stream data thus requires unifying the identifiers in each event with their respective team, player, event type, tag, or qualifier, which can be a costly operation if done without proper care and intelligent software engineering.

Wyscout's event stream data has a simpler structure than the data of the other two event stream data vendors. No official study has been conducted yet, but in the experience of the author of this dissertation and some prominent soccer analytics practitioners, Wyscout's data tends to be less accurate than that of their competitors (e.g., wrong event labels, inaccurate locations, large gaps in event sequences). The upside however is that Wyscout collects data in a wider variety of leagues (often smaller leagues in South-America and Asia) and also sells its data at a cheaper price.

While vendors such as Opta and StatsBomb often like to emphasize the many details and sheer volume of information recorded in their data, one consequence of this that often goes unspoken is the engineering cost of processing their detailed event stream data into a workable format.

```
{"id": 190079081,
"playerId": 14763, "teamId": 1631, "matchId": 2499773,
"matchPeriod": "1H", "eventSec": 20.65864,
"eventName": 1, "subEventName": 11,
"positions": [ {"x": 73, "y": 14}, {"x": 77,"y": 26} ],
"tags": [ {"id": 501}, {"id": 701}, {"id": 1802}]}
```


Figure 3.2: An event provided by Wyscout


```
{"id": "17dd9550-1632-49c1-9e71-a2f04fa2e14f",
"index": 26, "period": 1, "minute": 0, "second": 22,
"timestamp": "00:00:22.419",
"type": {"id" : 42, "name" : "Ball Receipt*"},
"possession": 2,"possession_team":{"id":902,"name":"Huesca"},
"play_pattern": { "id":9, "name":"From Kick Off"},
"team":{"id":902,"name":"Huesca" },
"player":{"id":11671, "name":"Juan Camilo Hernandez Suarez"},
"position":{"id":22, "name":"Right Center Forward"},
"location":[ 92.0, 78.0 ],
"related_events":["e2ef62bf-6ca4-45ed-a169-7af61eed1031"],
"ball_receipt":{"outcome":{"id":9, "name":"Incomplete"}}}
```


Figure 3.3: An event provided by StatsBomb

3.1.2 Engineering Challenges

When processing event stream data, three key engineering challenges arise:

Challenge 1: Event stream data serves multiple different objectives (e.g., reporting information to broadcasters, computing box scores of players). Hence, the data is not necessarily designed to facilitate analysis. Some important information is missing (e.g., Wyscout does not record exact end locations for shots) or irrelevant information is included (e.g., Opta records weather changes).

Challenge 2: Each vendor of event stream data uses their own unique terminology and definitions to describe the events that occur during a game. For example, the action of trying to get past an opposing player while keeping possession of the ball is denoted as a *take-on* by Opta and a *dribble* by StatsBomb. Hence, software written for event stream data has to be tailored to a specific vendor and cannot be used without modifications to analyze data from another vendor.

Challenge 3: All vendors include optional information snippets that provide more detailed information on the event that occurred (e.g., whether the ball was passed through the air or over the ground, whether a goal attempt was a header or a foot shot). Opta encodes these optional information snippets as *Qualifiers* (`"Q"` in Figure 3.1), Wyscout encodes them as *tags* (`"tags"` in Figure 3.2), and StatsBomb includes them at the top-level of the event data structure (`"possession"`, `"play_pattern"`, and `"ball_receipt"` in Figure 3.3). While sometimes useful, these optional information snippets cause event stream data to have a dynamic structure and thus make it extremely difficult to automatically parse the data and apply automatic analysis techniques.

3.2 SPADL: A Language for Describing Actions

Based on domain knowledge and feedback from soccer experts, we propose SPADL (**S**occer **P**layer **A**ction **D**escription **L**anguage) [42] as an attempt to unify the existing event stream formats into a common vocabulary that enables subsequent data analysis. SPADL was designed to accurately define and describe actions on the pitch in a way that is *human-interpretable*, *simple* and *complete*. The human-interpretability allows reasoning about what happens on the pitch and verifying whether the values assigned to those actions correspond to soccer experts' intuitions. The simplicity reduces the chance of making mistakes when automatically processing the data. The completeness enables expressing all the information required to analyze actions in their full context.

To address the challenges posed by the variety of event stream data formats and to benefit the soccer analytics community, we released a Python package that can automatically convert the data of different vendors to SPADL.[2] The package currently supports event stream data of Opta, Wyscout, and StatsBomb.

SPADL is a language for describing player *actions*, as opposed to the formats by commercial vendors that describe *events*. The distinction is that actions are a subset of events that require a player to perform the action. For example, a passing event is an action, whereas an event signifying the end of the game is not an action. We represent a game as a sequence of on-the-ball actions $[a_1, a_2, \ldots, a_m]$, where m is the total number of actions that happened in the game. Each action is a tuple of eight attributes:

Time: the time in the game when the action occurred,

StartLocation: the (x, y) location where the action started,

[2] `https://github.com/ML-KULeuven/socceraction`

EndLocation: the (x, y) location where the action ended,

Player: the player who performed the action,

Team: the player's team,

ActionType: the type of the action (e.g., *pass*, *shot*, *dribble*),

BodyPart: the player's body part used for the action,

Result: the result of the action (e.g., *success* or *fail*).

Note that, unlike all other event stream data formats, we always store the same eight attributes for each action. Excluding optional information snippets enables us to more easily apply automatic analysis tools.

We distinguish between 21 possible types of actions including, among others, *passes*, *crossed corners*, *dribbles*, *throw-ins*, *tackles*, *shots*, *penalty shots*, *clearances*, and *keeper saves*. These action types were, in collaboration with domain experts, designed to be interpretable and specific enough to accurately describe what happens on the pitch, yet general enough such that similar actions have the same type. The list of all possible action types can be found in Table 3.1 and their relatively frequency can be observed in Figure 3.4.

We consider up to four different body parts and up to six possible results. The possible body parts are *foot*, *head*, *other*, and *none*. The two most common results are *success* or *fail*, which indicates whether the action had its intended result or not. For example, a pass reaching a teammate or a tackle recovering the ball. The four other possible results are *offside* for passes resulting in an off-side call, *own goal*, *yellow card*, and *red card*.

3.3 Atomic-SPADL: A Language for Describing Atomic Actions

When building models to value actions, a heavy point of debate is how to handle the results of actions. In other words, should our models make a distinction between a failed and a successful action or not? On the one hand, an action should be valued on all its properties, and whether or not the action was successful (e.g., did a pass receive a teammate, was a shot converted into a goal) plays a crucial role in how useful the action was. That is, if you want to measure a player's contribution during a match, successful actions are important. This is the viewpoint of SPADL and VAEP which will be discussed in Chapter 5.

Table 3.1: Overview of the 21 action types in SPADL alongside their descriptions. The *Success?* column specifies the condition the action needs to fulfil to be considered successful, while the *Special* column lists additional possible results.

Action type	**Description**	**Success?**	**Special result**
Pass	Normal pass in open play	Reaches teammate	Offside
Cross	Cross into the box	Reaches teammate	Offside
Throw-in	Throw-in	Reaches teammate	-
Crossed corner	Corner crossed into the box	Reaches teammate	Offside
Short corner	Short corner	Reaches teammate	Offside
Crossed free-kick	Free kick crossed into the box	Reaches teammate	Offside
Short free-kick	Short free-kick	Reaches team mate	Offside
Take on	Attempt to dribble past opponent	Keeps possession	-
Foul	Foul	Always fail	Red or yellow card
Tackle	Tackle on the ball	Regains possession	Red or yellow card
Interception	Interception of the ball	Always success	-
Shot	Shot attempt not from penalty or free-kick	Goal	Own goal
Penalty shot	Penalty shot	Goal	Own goal
Free-kick shot	Direct free-kick on goal	Goal	Own goal
Keeper save	Keeper saves a shot on goal	Always success	-
Keeper claim	Keeper catches a cross	Does not drop the ball	-
Keeper punch	Keeper punches the ball clear	Always success	-
Keeper pick-up	Keeper picks up the ball	Always success	-
Clearance	Player clearance	Always success	-
Bad touch	Player makes a bad touch and loses the ball	Always fail	-
Dribble	Player dribbles at least 3 meters with the ball	Always success	-

On the other hand, including the result of an action intertwines the contribution of the player who initiated the action (e.g., provided the pass) and the player who completed it (e.g., received the pass). Perhaps a pass was not successful because of its recipient's poor touch or because he was not paying attention. It would seem unfair to penalize the player who provided the pass in such a circumstance. Hence, it can be useful to generalize over possible results of an action to arrive at an action's "expected value". This is exactly the purpose of the expected goals metric which values shots by generalizing over the two possible results of shots (*success* and *fail*) [78].

To accommodate this alternative viewpoint, we introduce Atomic-SPADL, which removes the *result* attribute from SPADL and adds a few new action and event types. In this representation, all actions are "atomic" in the sense that they are always completed successfully without interruption. Consequently, while SPADL treats a pass as one action consisting of both the initiation and completion of the pass, Atomic-SPADL sees giving and receiving a pass as two separate actions. Because not all passes successfully reach a teammate, Atomic-SPADL introduces an *interception* action if the ball was intercepted by the other team or an *out* event if the ball went out of play. We similarly divide shots, freekicks, and corners into two separate actions. Practically, the effect is that this representation helps to distinguish the contribution of the player who initiated the action (e.g., gives the pass) and the player who completed it (e.g.,

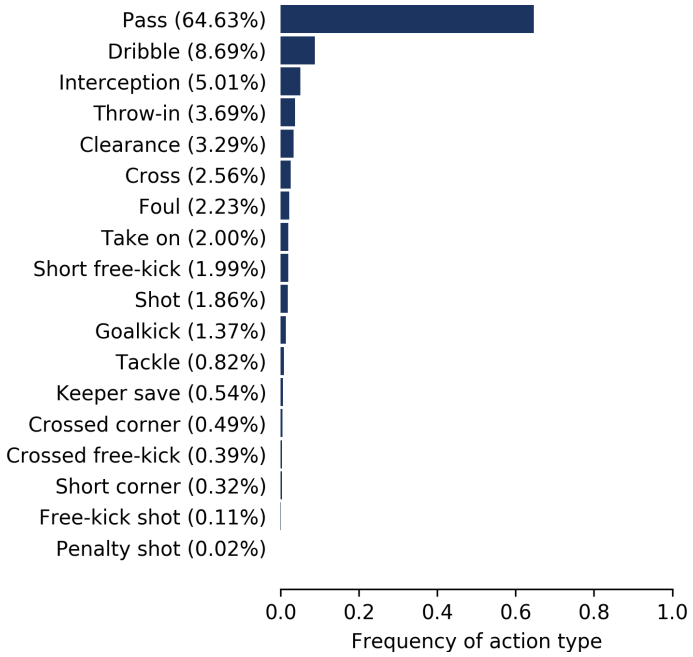


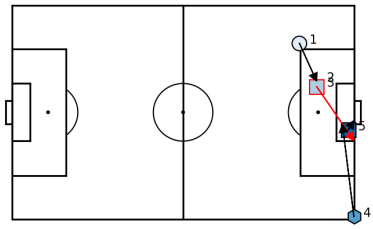
Figure 3.4: Frequency of each SPADL action type when converting Wyscout data to the SPADL format. The converted Wyscout data spanned all games in the English, Spanish, German, Italian, French, Dutch, and Belgian top divisions in the 2012/2013 through 2017/2018 seasons. After converting the Wyscout data to SPADL, each game contained ±1250 actions on average.

receives the pass).

Figure 3.5 and Figure 3.6 illustrate the differences between SPADL and Atomic-SPADL by plotting the same phase in both formats. The phase considered is the actions leading up to the only goal in the France vs Belgium match in the semi-final of the 2018 FIFA World Cup. Both the SPADL and Atomic-SPADL representations were converted from raw StatsBomb data of this match.

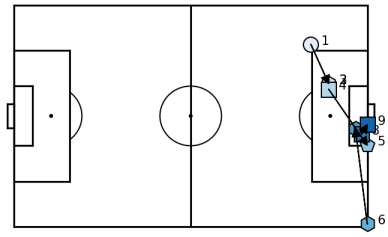
3.4 Analysis Challenges

While converting event stream data to the SPADL or Atomic-SPADL format addresses some key engineering challenges, there also exist other challenges



	time	actiontype	player	team
1	50m0s	pass	Blaise Matuidi	France
2	50m1s	dribble	Olivier Giroud	France
3	50m2s	shot	Olivier Giroud	France
4	50m33s	corner_crossed	Antoine Griezmann	France
5	50m35s	shot	Samuel Umtiti	France

Figure 3.5: The actions leading up to the only goal in the France vs Belgium match in the semi-final of the 2018 FIFA World Cup encoded in the SPADL format. The first three actions (pass, dribble, shot) detail France's first attempt to score, which they missed. The next two actions detail the subsequent corner from which they headed the ball in.



	time	actiontype	player	team
1	50m0s	pass	Blaise Matuidi	France
2	50m0s	receival	Olivier Giroud	France
3	50m1s	dribble	Olivier Giroud	France
4	50m2s	shot	Olivier Giroud	France
5	50m2s	out	Olivier Giroud	France
6	50m33s	corner	Antoine Griezmann	France
7	50m34s	receival	Samuel Umtiti	France
8	50m35s	shot	Samuel Umtiti	France
9	50m35s	goal	Samuel Umtiti	France

Figure 3.6: The same actions as in Figure 3.5 encoded in the Atomic-SPADL format. Both the pass and the corner kick now have a subsequent *receival* action by the player successfully receiving the ball. In addition, the results of the shots have also been explicitly added to the event stream in the form of *out* and *goal* events. Finally, we removed the distinction between *corner_crossed* and *corner_short* (as encoded in SPADL), as at the exact moment the player kicks the ball, we have no idea whether the corner will be crossed or taken short. Hence, these two actions were merged into the simpler *corner* action.

that cannot be solved by converting the data to a new format. In this section, we discuss nine challenges inherent to all forms of soccer event stream data that arise when analyzing this data with techniques from the field of artificial intelligence.

1. Actions have both discrete (e.g., *type*, *player*, *team*) and continuous (e.g., *time*, *location*) attributes. Most artificial intelligence techniques prefer to work on either discrete or continuous data exclusively and rarely work well on a mix of both.

2. There is little exact repetition in sequences of game play. That is, the same players rarely perform the same actions in the same order in the same locations. This makes counting interesting occurrences difficult, as the counting method must intelligently generalize over action attributes.

3. It is often necessary to augment the data with domain knowledge such as the role of a player within their team's formation, the location of the goal, and the playing direction of a team on the pitch (i.e., left to right or right to left). The challenge lies in determining what domain knowledge to include and how to encode the domain knowledge in such a way that it can aid the analysis technique.

4. Event stream data is constructed by human annotators and thus imperfect. While most vendors have rigorous procedures in place to check the quality of the data they produce, the human element in the data collection procedure will inevitably lead to both missing data (i.e., unexplainable gaps in sequences of events) and incorrect data such as faulty event names and wrong coordinates.

5. The forced discreteness of event stream data is sometimes unable to capture the nuances and fluidity of the game. For example, whether a specific action was an intended shot or an accidental cross or whether a certain player was involved in an action or not can be up for debate. There is not always a clear yes or no answer to these questions, yet human annotators are forced to pick an answer and annotate the data this way.

6. Event stream data mostly focuses on on-the-ball actions (e.g., passes, shots). However, over the course of a game, many important off-the-ball actions occur such as strikers sprinting towards a position to open up space for a teammate, midfielders putting pressure on opposing players to deny their further advancement, and defenders carefully positioning themselves to obstruct their opponent's line of sight to the goal.

7. Most events within event stream data only detail the location of the one player currently in possession of the ball. This means that even the most

advanced analysis techniques are inherently limited in how well they can analyze the data because they are always missing often crucial context, i.e., the locations of the other 21 players on the field. For example, how can we evaluate the decision making behind a pass if we do not know what teammates were available and where they were positioned at that moment in the match?

8. Many popular AI techniques often required fixed-size feature vectors and cannot natively handle a sequence or set of data points of varying size.

9. There is often no ground truth present in the data. Both value and style are subjective concepts for which there is no ground truth in the data to compare the learned results against (i.e., there is no test or validation set like in more traditional AI/ML tasks).

Chapter 4

STARSS: A Spatio-Temporal Action Rating System for Soccer

In this chapter, we present STARSS (**S**patio-**T**emporal **A**ction **R**ating **S**ystem for **S**occer), which leverages historical match data to assign a rating to the actions (e.g., a pass or a shot) performed by the players in a match. For a given match, the presented approach proceeds in three steps. First, the approach splits the match into phases, which are uninterrupted sequences of actions where one team is in possession of the ball. Second, it assigns a phase rating to each phase based on historical match data. The higher the assigned rating, the more likely that the phase will end in a goal. Hence, our approach focuses on the actions that contribute to the offensive output of the team. Third, the approach distributes the phase rating across the individual actions that constitute the phase.

We use STARSS to rate players and teams in individual matches as well as throughout the course of a season. We present the top-15 players for the 2015/2016 season in the English Premier League, the German Bundesliga, and the Spanish La Liga. We find that wingers and attacking midfielders tend to contribute more to a team's STARSS rating than strikers, that five of the top-15 players in the German Bundesliga play for FC Bayern Munich, and that Lionel Messi is the best player in these three leagues.

The content of this chapter is based on the following publication [51]:

Decroos, T., Van Haaren, J., Dzyuba, V., and Davis, J. STARSS: A Spatio-Temporal Action Rating System for Soccer. In *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2017 workshop* (2017)

4.1 Methodology

STARSS is an approach for automatically rating the actions performed by soccer players. Unlike traditional approaches [115, 171, 55], which simply compute a weighted sum of the frequencies of a hand-picked set of action types, our approach accounts for the spatio-temporal context in which the actions were performed. More specifically, STARSS leverages the outcomes of similar actions that were performed in similar circumstances in the past to assess the value of a particular action to the team.

To rate the player actions in a given match, our approach performs the following three steps. First, it splits the match into phases of related actions. Second, it assigns *phase ratings* to the resulting phases based on historical match data. Third, it assigns *action ratings* to the individual actions that constitute the phases.

4.1.1 Splitting Matches into Phases

We represent a match as a sequence of n player actions $[a_1, \ldots, a_n]$. We start a new phase each time possession switches from one team to the other or too much time (i.e., 10 seconds) has passed between consecutive actions (Figure 4.1). For example, when the ball goes out of play for a throw-in or corner kick, when a goal is scored, or when a free kick is awarded.

This way, a match is split into a number of phases, where each phase P is a subsequence of consecutive actions $[a_a, \ldots, a_b]$.

4.1.2 Valuing Phases

We assign a rating to a given phase P in two steps as follows. In the first step, we search the k most-similar phases in terms of their spatial location on the pitch in historical match data. This historical match data bas been split up into a large number of phases using the procedure of the previous section. To measure the similarity between two phases P and P', we employ dynamic time

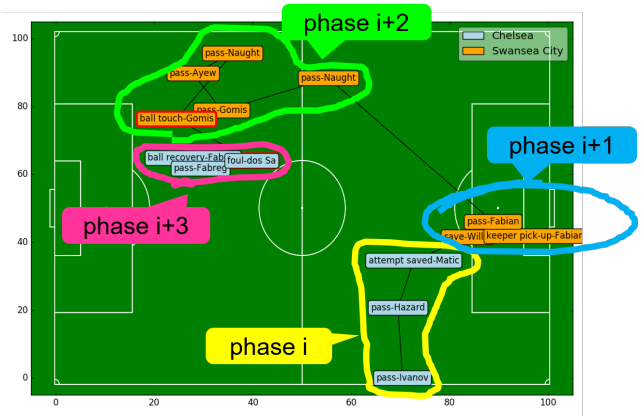


Figure 4.1: A sequence of actions being split in coherent phases.

warping (DTW) [123] as this distance function does not require two sequences to have the same length and is insensitive to minor mismatches. Specifically, we compute the distance $d(\cdot, \cdot)$ between two phases P and P' as follows:

$$d(P, P') = \sqrt{DTW(P_x, P'_x)^2 + DTW(P_y, P'_y)^2} \tag{4.1}$$

where P_x (P'_x) and P_y (P'_y) denote the x and y coordinates of the actions that constitute phase P (P'). The core idea behind this approach is to reward teams and players for phases that get the ball into dangerous places, even if the phases do not lead to a shot. Furthermore, we are less interested in which action is used to get the ball into the location, as we do not want to favor one style of play over another. Most existing approaches to rate phases in soccer (e.g., [88, 27, 108]) only take phases that lead to a goal or a shot into account.

As exhaustive search to find nearest neighbors is computationally expensive, we use a vantage-point (VP) tree [187], which achieves efficient ordering using an application-specific distance metric. This permits look up in logarithmic time (in the size of the training data) for identifying the nearest neighbors. There is a small risk that we will miss a nearest neighbor with the VP as it technically requires a distance metric and DTW violates the triangle inequality. However, DTW almost always satisfies the triangle inequality on real-world data [180], so we prefer the run time speed up over the small risk of missing a nearest neighbor.

In the second step, we compute the *phase rating* as the proportion of similar phases that end in a goal:

$$phase\_rating(P) = \frac{1}{k} \sum_{P' \in NN_k(P)} \mathbb{1}_{goal}(P') \tag{4.2}$$

where $\mathbb{1}_{goal}(P)$ is an indicator function that is 1 if P ends in a goal and 0 if it does not and $NN_k(P)$ is the set of the k most similar phases to P (i.e., the Nearest Neighbors) according to the distance function $d(\cdot, \cdot)$ in Equation 4.1. For example, if we want to rate a phase P using $k = 100$ similar phases and 13 out of the 100 most similar phases end in a goal, then the rating of P amounts to 0.13.

4.1.3 Distributing Phase Values over Actions

Assume we are given a phase $P = [a_a, \ldots, a_b]$ with a_a the first action of the phase and a_b the last action of the phase. We then assign a rating to each action a_i in P in two steps. We first compute an action weight w_i that indicates the relevance of the action a_i to the phase P, and then compute the *action rating* by multiplying the rating for the phase P with the normalized action weight w'_i.

We use an exponential-decay-based approach to compute the action weights. We consider actions that happen at the end of a phase more important than actions that happen at the start of a phase. Hence, we assign low weights to actions that happen at the start and high weights to actions that happen at the end. Intuitively, this makes sense as the last few actions in a phase have the highest influence on its outcome. We compute the action weights in reverse order, starting with the last action a_b and working our way down to the first action a_a, using the following formula:

$$w_i = \begin{cases} 1 & \text{if } i = b, \\ (1 - \lambda) \cdot w_{i+1} & \text{if } a \leq i < b \end{cases} \tag{4.3}$$

where λ is a user-specified parameter.

We normalize the weights such that they sum to one, which means that the phase rating is completely distributed across the individual actions.

$$w'_i = \frac{w_i}{\sum_{j=a}^{b} w_j} \tag{4.4}$$

Finally, the rating for an action $a_i \in P$ is computed by multiplying the phase rating with the normalized action weight.

$$action\_rating(a_i) = w'_i \cdot phase\_rating(P) \tag{4.5}$$

4.2 Experiments

The dataset in our experiments consists of event stream data for the English Premier League, German Bundesliga, and Spanish La Liga for the 2012/2013 through 2015/2016 seasons.[1] Our dataset comprises 4253 matches, 7,569,802 game events, 110,290 shots, and 11,842 goals.

We use STARSS to rate players in individual matches as well as throughout the course of an entire season as follows. For a given player, we first sum the ratings for the actions performed by the player during a match or season, then divide this sum by the total number of minutes that the player played, and finally multiply by 90 to obtain a rating normalized per 90 minutes. We refer to this metric as *starss/90*. Similarly, we also rate teams in individual matches and over the course of a season.

We respect the chronological order of the matches when rating player actions. To rate a player or team in a particular match, we only leverage matches from the same league that had already been played at the time of that match to discover similar phases. For example, to rate the Leicester City players in their 1-3 win over Manchester City on 6 February 2015, we only consider the Premier League matches in our dataset that were played before that date.

This section presents case studies that illustrate the utility of the presented STARSS approach. Concretely, we address the following two questions:

1. Can STARSS identify the top-performing players in a league?
2. Can STARSS identify the top-performing players in a match?

For each of the case studies, we set the parameters $k = 100$ and λ for computing the action weights to 0.25 based on domain knowledge and an empirical analysis of the available data.

4.2.1 Identifying the Top Players in a Season

We compute the player ratings for all players that played at least 600 minutes in the 2015/2016 season of the English Premier League, German Bundesliga,

[1] Note that in this chapter we do not use the SPADL format of Chapter 3, but instead use raw event stream data of a prominent vendor. The reason for this is that the contributions in this chapter chronologically precede the creation of the SPADL format.

and Spanish La Liga. We present the top-15 players for each league in tables 4.1, 4.2, and 4.3, respectively. Note that these rankings differ from simply ranking players based on a combination of goals and assists per 90 minutes, and hence they provide insight beyond using these traditional metrics. Alexis Sánchez was a key player for championship contenders Arsenal and tops the Premier League ranking. Zlatko Junuzović, whose assists were instrumental for Werder Bremen in their battle against relegation, is the top-ranked player in the Bundesliga. Lionel Messi, who helped Barcelona claim the league title, tops the La Liga ranking. Unsurprisingly, the five-time FIFA Ballon d'Or winner is also the top-ranked player across the three leagues. Unlike the Premier League and Bundesliga, the La Liga ranking exhibits a clear gap between Lionel Messi and the rest, indicating that the Argentine forward is a class apart as was also suggested by earlier work [122].

The Premier League ranking suggests that Arsenal's offensive compartment excelled in the 2015/2016 season. The Gunners, who eventually finished second, have four players in the top 15 with Santi Cazorla, Mesut Özil, and Olivier Giroud alongside top-ranked Alexis Sánchez. In contrast, surprise champions Leicester City have not a single player in the top 15, despite the fact that Riyad Mahrez won the Player's Player of the Year award and Jamie Vardy won the Premier League Player of the Season Award. These players were ranked first and second in the league according to summing total goals and assists. Traditional metrics like expected-goals indicated that Leicester were hugely over-performing last season, that is, the results were much better than the underlying numbers. Additionally, Leicester were also awarded a remarkably high number of penalties (13 in total, while the league average was 4.5 league average).

The Bundesliga ranking clearly shows FC Bayern Munich's superiority in the 2015/2016 season. Although Werder Bremen's free-kick specialist Zlatko Junuzović tops the ranking, the eventual champions have five players in the top 15. The La Liga ranking sees most of the usual suspects near the top of the ranking with Lionel Messi (Barcelona), Neymar (Barcelona), Cristiano Ronaldo (Real Madrid), and Gareth Bale (Real Madrid) occupying the first four positions.

These rankings also indicate several highly-ranked players who made moves to larger clubs following the season. These include Ilkay Gundogan and Nolito, who both transferred to Manchester City, and Henrikh Mkhitaryan, who moved to Manchester United.

Table 4.1: Top-15 players in the 2015/2016 English Premier League season. starss/90 refers to our STARSS player ratings, while g/90 and a/90 refer to respectively goals and assists per 90 minutes.

Rank	Team	Player	starss/90	g/90	a/90
1	Arsenal	Alexis Sanchez	0.289	0.478	0.147
2	West Ham	Dimitri Payet	0.279	0.315	0.420
3	West Ham	Mauro Zarate	0.262	0.342	0.000
4	Chelsea	Willian	0.249	0.164	0.196
5	Liverpool	Philippe Coutinho	0.244	0.359	0.225
6	Arsenal	Santi Cazorla	0.240	0.000	0.209
7	Arsenal	Mesut Ozil	0.240	0.177	0.561
8	Sunderland	Wahbi Khazri	0.240	0.167	0.084
9	Aston Villa	Rudy Gestede	0.237	0.272	0.109
10	Manchester City	Kevin De Bruyne	0.233	0.315	0.404
11	Tottenham	Christian Eriksen	0.231	0.184	0.398
12	Arsenal	Olivier Giroud	0.231	0.594	0.223
13	Liverpool	Christian Benteke	0.229	0.474	0.178
14	Tottenham	Erik Lamela	0.228	0.189	0.340
15	Manchester City	David Silva	0.222	0.100	0.550

Table 4.2: Top-15 players in the 2015/2016 German Bundesliga season. starss/90 refers to our STARSS player ratings, while g/90 and a/90 refer to respectively goals and assists per 90 minutes.

Rank	Player	Team	starss/90	g/90	a/90
1	SV Werder Bremen	Zlatko Junuzovic	0.271	0.153	0.383
2	Borussia Dortmund	Ilkay Gundogan	0.264	0.045	0.135
3	VfB Stuttgart	Alexandru Maxim	0.263	0.086	0.430
4	FC Augsburg	Tobias Werner	0.258	0.000	0.129
5	SV Darmstadt 98	Sandro Wagner	0.257	0.436	0.145
6	Borussia Dortmund	Henrikh Mkhitaryan	0.257	0.384	0.524
7	FC Bayern Munchen	Thiago Alcantara	0.255	0.110	0.165
8	FC Bayern Munchen	Franck Ribery	0.253	0.264	0.396
9	FC Bayern Munchen	Robert Lewandowski	0.252	0.950	0.068
10	FC Bayern Munchen	Thomas Muller	0.250	0.576	0.192
11	FC Bayern Munchen	Arjen Robben	0.248	0.245	0.082
12	FC Ingolstadt 04	Dario Lezcano	0.247	0.145	0.072
13	Bayer 04 Leverkusen	Hakan Calhanoglu	0.246	0.080	0.199
14	VfB Stuttgart	Daniel Didavi	0.245	0.407	0.111
15	VfB Stuttgart	Daniel Ginczek	0.244	0.286	0.143

Table 4.3: Top-15 players in the 2015/2016 Spanish La Liga season. starss/90 refers to our STARSS player ratings, while g/90 and a/90 refer to respectively goals and assists per 90 minutes.

Rank	Team	Player	starss/90	g/90	a/90
1	Barcelona	Lionel Messi	0.387	0.759	0.528
2	Barcelona	Neymar	0.339	0.559	0.353
3	Real Madrid	Cristiano Ronaldo	0.320	0.820	0.311
4	Real Madrid	Gareth Bale	0.310	0.984	0.518
5	Malaga	Duda	0.304	0.085	0.085
6	Celta de Vigo	Nolito	0.297	0.400	0.255
7	Real Madrid	James Rodriguez	0.286	0.416	0.475
8	Sevilla	Yevhen Konoplyanka	0.273	0.167	0.223
9	Sevilla	Ever Banega	0.266	0.209	0.104
10	Real Madrid	Isco	0.251	0.148	0.345
11	Barcelona	Luis Suarez	0.247	1.057	0.457
12	Atletico de Madrid	Angel Correa	0.247	0.475	0.380
13	Real Madrid	Jese	0.243	0.544	0.653
14	Celta de Vigo	Orellana	0.238	0.185	0.216
15	Eibar	Saul Berjon	0.238	0.064	0.257

4.2.2 Identifying the Top Players in a Match

To identify the top players in a match, we perform three steps. First, we compute the rating for each player in each match in the 2015/2016 season of the English Premier League, German Bundesliga, and Spanish La Liga. Second, we compute the team rating for each team in each match by summing the individual player ratings. Third, we compute each player's share of the team rating.

Figure 4.2 shows the shares of the player ratings in the team ratings for El Clásico, a match between fierce rivals FC Barcelona and Real Madrid in La Liga, on 2 April 2016. Unsurprisingly, Lionel Messi, Luis Suarez, and Neymar occupy the first three spots for FC Barcelona, while Gareth Bale, Cristiano Ronaldo, and Karim Benzema occupy the first three spots for Real Madrid.

4.3 Related Work

This section discusses related work in soccer as well as other sports.

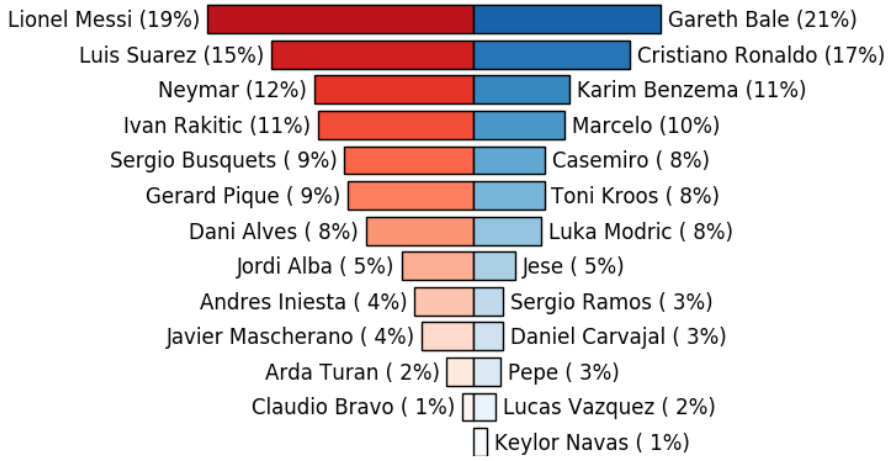


Figure 4.2: The shares of the player ratings in the team ratings for El Clásico, a match between fierce rivals Real Madrid and FC Barcelona in La Liga, on 2 April 2016. FC Barcelona is shown at the left, while Real Madrid is shown at the right.

STARSS is related to the work on expected-goals models, which have been a hot topic in the soccer-analytics community for the past few years. Expected-goals models aim to objectively quantify the quality of goal attempts and several different models have been proposed in recent years [89, 27, 108]. However, our work differs from these existing approaches in two crucial aspects. First, our approach is not restricted to shots and rates players based on all actions contributing to the team's offensive output. Second, our approach explicitly takes the spatio-temporal context of the actions into account, as suggested by Altman et al. [6].

There are a number of websites such as WhoScored.com and Squawka.com that provide player ratings for soccer on a match-by-match basis. These websites have hand-crafted formulas that simply compute a weighted sum of frequencies for a hand-picked set of actions (e.g., shots, tackles, etc.). The weights associated with each action are set by hand according to domain knowledge. The importance of some of the defensive statistics such as the number of tackles is debatable. A high percentage of successful tackles is often considered a good thing but can also be the result of poor positioning. Our method differs from these approaches in that we avoid hand-crafting and use an automated data-driven approach to assign rankings. Furthermore, we consider the spatio-temporal context in which the actions were performed which the hand-crafted models ignore.

While virtually unexplored to date for soccer, the task of objectively quantifying player actions has been investigated for other sports. Cervone et al. [32] propose the Expected Possession Value (EPV) model for basketball, which estimates the number of points a team is expected to score during a possession. uses a multiresolution semi-Markov stochastic model that defines a probability distribution over what the ballhandler is likely to do next, given the spatial configuration of the players on the court. Hence, this approach requires optical tracking data for all players. Routley and Schulte [146] introduce a conceptually similar model for ice hockey. They note that valuing actions can be posed within a reinforcement learning framework, which is challenging as in sports there is only access to a fixed data set, not a dynamic environment in which we can run new trials. Their approach considered a discrete state space and ignored locational information, which is highly important in soccer. Pettigrew [133] assesses the offensive productivity of hockey players based on the context in which they score goals. Schuckers and Curro [151] introduce the Total Hockey Rating (THoR) which goes beyond shots and goals to rate hockey players by taking all game events into account.

4.4 Conclusions

This chapter introduced STARSS, which is an approach for automatically rating the actions performed by soccer players. Viewing a soccer match as a sequence of actions performed by players, the approach proceeds in three steps to rate these actions. First, it splits the match into phases of related actions. Second, it assigns a rating to each phase, indicating how likely it is that the phase will end in a goal. Third, it distributes the assigned rating over the individual actions that constitute the phase.

Unlike more simple approaches for rating soccer players, our approach goes beyond shots and goals. It considers all the actions that contribute to a team's offensive output and accounts for the spatio-temporal context in which these actions were performed. Several case studies show that our approach is able to identify top-performing players in individual matches as well as throughout the course of an entire season.

Chapter 5

VAEP: A Framework for Valuing Actions

This chapter proposes VAEP, a novel data-driven framework for valuing actions in a soccer game. Unlike most existing work, the framework can value all types of actions (e.g., passes, crosses, dribbles, take-ons, and shots), considers not only the offensive, but also the defensive value of actions, and accounts for the circumstances under which each of these actions happened as well as their possible longer-term effects. Intuitively, an action value reflects the action's expected influence on the scoreline. That is, an action valued at +0.05 is expected to contribute 0.05 goals in favor of the team performing the action, whereas an action valued at -0.05 is expected to yield 0.05 goals for their opponent.

In summary, this chapter makes the following four contributions:

1. a framework for valuing player actions and rating players based on their impact on the game;
2. a model for predicting short-term scoring and conceding probabilities at any moment in a game;
3. a number of use cases showcasing our most interesting results and insights;
4. a Python package[1] that (a) converts existing event stream data to our language, (b) implements our framework, and (c) constructs a model that estimates scoring and conceding probabilities.

[1] `https://github.com/ML-KULeuven/socceraction`

This chapter is based on the following publications:

Decroos, T., Bransen, L., Van Haaren, J., and Davis, J. Actions Speak Louder than Goals: Valuing Player Actions in Soccer. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2019), KDD '19, ACM, pp. 1851–1861

Decroos, T., and Davis, J. Interpretable Prediction of Goals in Soccer. In *AAAI 2020 Workshop on Artificial Intelligence in Team Sports* (2020)

Decroos, T., Bransen, L., Van Haaren, J., and Davis, J. VAEP: An Objective Approach to Valuing On-the-Ball Actions in Soccer (Extended Abstract). In *Proceedings of the 29th International Joint Conference on Artificial Intelligence* (2020), AAAI Press

5.1 VAEP: A Framework for Valuing Actions

This section introduces the VAEP (**V**aluing **A**ctions by **E**stimating **P**robabilities) framework for valuing actions performed by soccer players. First, we show how to use scoring and conceding probabilities to compute objective action values. Next, we show how to convert a set of action values to a player rating that represents the player's total offensive and defensive contribution to their team.

5.1.1 Converting Scoring and Conceding Probabilities to Action Values

Broadly speaking, most actions in a soccer game are performed with the intention of (1) increasing the chance of scoring a goal, or (2) decreasing the chance of conceding a goal. Given that the influence of most actions is temporally limited, one way to assess an action's effect is by calculating how much it alters the chances of both scoring and conceding a goal in the near future. We treat the effect of an action on scoring and conceding separately as these effects may be asymmetric in nature and context dependent.

We view a soccer game as a sequence of on-the-ball actions $[a_1, \ldots, a_n]$. The effect of every on-the-ball action is that it alters the game state. That is, an action a_i moves the game from game state S_{i-1} to game state S_i. Each game state S_i can be represented by all the actions $[a_1, \ldots, a_i]$ that have happened up to that point in the game.

Suppose that for each game state S_i, we have access to the probabilities of scoring and conceding in the near future for the home team h and the visiting team v. Let $P_{scores}(S_i, h)$ and $P_{concedes}(S_i, h)$ denote the probability of the home team h respectively scoring and conceding in the near future. Similarly, let $P_{scores}(S_i, v)$ and $P_{concedes}(S_i, v)$ denote the probability of the visiting team v respectively scoring and conceding in the near future.

Valuing an action for a team then requires assessing the *change* in probability for both scoring and conceding as a result of action a_i moving the game from state S_{i-1} to state S_i. The change in probability for team x scoring, where x can be either the home team h or the visiting team v, can be computed as:

$$\Delta P_{scores}(a_i, x) = P_{scores}(S_i, x) - P_{scores}(S_{i-1}, x). \tag{5.1}$$

This change will be positive if the action increased the probability that team x will score a goal. We call this change $\Delta P_{scores}(a_i, x)$ the *offensive value* of an action a_i for team x. Similarly, the change in probability for team x conceding a goal can be computed as:

$$\Delta P_{concedes}(a_i, x) = P_{concedes}(S_i, x) - P_{concedes}(S_{i-1}, x). \tag{5.2}$$

This change will be positive if the action increased the probability that team x will concede a goal. However, all actions should always aim to decrease the probability of conceding. That is why we call the negation of this change $-\Delta P_{concedes}(a_i, x)$ the *defensive value* of an action a_i for team x.

We combine Equations 5.1 and 5.2 to derive an action's total VAEP value.

Definition 1 (VAEP Value)**.** *The total VAEP value of an action is the sum of that action's offensive value and defensive value.*

$$V(a_i, x) = \Delta P_{scores}(a_i, x) + (-\Delta P_{concedes}(a_i, x)) \tag{5.3}$$

Given that we are usually interested in the value of an action for the team of the player performing the action, we use $V(a_i)$ to denote $V(a_i, x_i)$, where x_i is the team of the player performing action a_i.

The VAEP framework provides a simple approach to valuing actions that is independent of the representation used to describe the actions. The framework's strength is that it transforms the subjective task of valuing an action into the objective task of predicting the likelihood of a future event in a natural way.

5.1.2 Converting Action Values to Player Ratings

Our method assigns a value to each individual action. We can aggregate the individual action values into a player rating for multiple time granularities as

well as along several different dimensions. A player rating could be derived for any given time frame, where the most natural ones would include a time window within a game, a full game, or a full season. Regardless of the time frame, we compute a player rating in the same manner. Since spending more time on the pitch offers more opportunities to contribute, we compute the player ratings per 90 minutes of game time. Given a time frame T and player p, we compute the player's VAEP rating as

$$rating(p) = \frac{90}{m} \sum_{a_i \in A_p^T} V(a_i), \tag{5.4}$$

where A_p^T is the set of actions the player p performed during time frame T, $V(a_i)$ is computed according to Definition 1, and m is the number of minutes the player played during T. This player rating captures the average net goal difference contributed to the player's team per 90 minutes.

Additionally, instead of summing over all actions, a player's rating can be computed per action type. This allows constructing a player profile, which may enable identifying different playing styles. In general, player ratings can be computed along different dimensions, depending on the use case.

5.2 Estimating Scoring and Conceding Probabilities

This section describes our method for estimating the scoring and conceding probabilities required by the VAEP framework. Let $goal(h)$ denote a goal scored by the home team h, and $goal(v)$ denote a goal scored by the visiting team v. Our task can then be defined as:

Given: game state $S_i = [a_1, \ldots, a_i]$;

Estimate: the probability of scoring and conceding in the near future for the home team h and the visiting team v, which we denote by:

$$\begin{aligned}
P_{scores}(S_i, h) &= P(goal(h) \in F_i^k | S_i) \\
P_{concedes}(S_i, h) &= P(goal(v) \in F_i^k | S_i) \\
P_{scores}(S_i, v) &= P(goal(v) \in F_i^k | S_i) \\
P_{concedes}(S_i, v) &= P(goal(h) \in F_i^k | S_i)
\end{aligned}$$

where $F_i^k = [a_{i+1}, \ldots, a_{i+k}]$ is the sequence of k actions that follow action a_i, and k is a user-defined parameter.

Because $P_{conceding}(S_i, h) = P_{scoring}(S_i, v)$ and $P_{scoring}(S_i, h) = P_{conceding}(S_i, v)$, we only have to estimate the probability of scoring and conceding for one team, and we get the probabilities of the other team for free. We leverage this fact by only estimating the scoring and conceding probabilities for the team that possessed the ball in game state S_i. Hence, our task simplifies to two separate binary probabilistic classification problems with identical inputs but different labels.

Given: game state S_i, where x_i is the team in possession of the ball during S_i;

Estimate: (1) $P_{scores}(S_i, x_i)$, and (2) $P_{concedes}(S_i, x_i)$.

Applying a standard machine learning algorithm requires converting the sequence of actions $[a_1, a_2, \ldots, a_m]$ describing an entire game into examples in the feature-vector format. Thus, one training example is constructed for each game state S_i. In the remainder of this section, we describe (1) the features of each game state, (2) the label we assign to each game state, (3) probabilistic classifiers to estimate scoring and conceding probabilities, and (4) evaluation metrics to evaluate the performance of the learned models.

5.2.1 Features

For each example, instead of defining features based on the entire current game state $S_i = [a_1, \ldots, a_i]$, we only consider the most recent three actions $[a_{i-2}, a_{i-1}, a_i]$. Approximating the game state in this manner offers several advantages. First, most machine learning techniques require examples to be described by a fixed number of features. Converting game states with varying numbers of actions, and hence different amounts of information, into this format would necessarily result in a loss of information. Second, considering a small window focuses attention on the most relevant aspects of the current context. The number of actions to consider is a parameter of the approach, and three actions was empirically found to work well. From these three actions, we define features that impact the probability of a goal being scored in the near future. Using the SPADL representation for actions introduced in Chapter 3, we consider three categories of features.

SPADL features. For each of the three actions, we define a set of categorical and real-valued features based on information explicitly included in the

SPADL representation. We consider categorical features for an action's type and result, and the body part used by the player performing the action. Similarly, we consider real-valued features for the (x, y)-coordinates of the action's start and end locations, and the time elapsed since the start of the game.

Complex features. The complex features combine information within an action and across consecutive actions. Within each action, these features include (1) the distance and angle to the goal for both the action's start and end locations, and (2) the distance covered during the action in both the x and y directions. Between two consecutive actions, we compute the distance and elapsed time between them and whether the ball changed possession. These features provide some intuition about the current speed of play.

Game context features. The game context features are (1) the number of goals scored in the game by the team possessing the ball after action a_i, (2) the number of goals scored in the game by the defending team after action a_i, and (3) the goal difference after action a_i. We include these features because teams often adapt their playing style to the current scoreline (e.g., a team that is 1-0 ahead will play more defensively than a team that is 0-1 behind).

5.2.2 Labels

For the first classification problem of estimating $P_{scores}(S_i, x_i)$, we assign a game state S_i a positive label (= 1) if the team possessing the ball after action a_i *scored* a goal in the subsequent k actions, and a negative label (= 0) in all other cases. Similarly, for the second classification problem of estimating $P_{concedes}(S_i, x_i)$, we assign a game state S_i a positive label (= 1) if the team possessing the ball after action a_i *conceded* a goal in the subsequent k actions, and a negative label (= 0) in all other cases.

In both binary classification problems, k is a user-defined parameter that represents how far ahead in the future we look to determine the effect of an action. In this chapter, we chose $k = 10$ based on domain knowledge and preliminary experiments.

5.2.3 Probabilistic Classifiers

To estimate scoring probabilities, we need a probabilistic classifier that can predict the labels from the features. We discuss three models: logistic

regression [129], XGBoost [35], and the lesser known generalized additive models (GAMs) [85].

Logistic Regression Logistic regression is a statistical model that uses a logistic function to model a binary target variable that depends on the input features [129]. Given an input feature vector $x = [x_1, x_2, \ldots, x_m]$ and a target variable y, logistic regression will learn the following function:

$$g(E[y]) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_m x_m = \alpha_0 + \sum_i^m \alpha_i x_i$$

where g is the *logit* link function and the weights α_i are learned from the training data. This formula illustrates how logistic regression is a linear model. It is also interpretable, as the sign of α_i shows whether there is a positive or negative correlation between feature x_i and the target variable y while the magnitude of α_i hints how big its impact is.

XGBoost XGBoost is a popular gradient boosting decision trees model that solves many data science problems in a fast and accurate way [35]. Due to its excellent performance, it has become the de facto standard model of choice for many data science practitioners when classifying tabular data. One of the reasons XGBoost works so well out of the box is that by using decision trees and the boosting mechanism, it can learn complex non-linear decision boundaries. The downside is that these complex non-linear decision boundaries are often too complex for humans to grasp and thus XGBoost is in practice a black box model.

Generalized Additive Model Generalized additive models (GAMs) are statistical models that model the target variable as a sum of univariate functions [85]. Standard GAMs have the form

$$g(E[y]) = f_1(x_1) + f_2(x_2) + \cdots + f_m(x_m) = \sum_i^m f_i(x_i)$$

where g is the *logit* link function and the functions f_i are learned from the training data. This model is interpretable in the sense that users can visualize the relationship between the univariate terms of the GAM and the dependent variable through a plot $f_i(x_i)$ vs. x_i. The formula illustrates how GAMs are a more powerful generalization of logistic regression models, replacing the linear functions $\alpha_i x_i$ with more complex non-linear functions $f_i(x_i)$.

One of the reasons why GAMs have been less popular up until now is lack of a mature and widely available implementation. However, in summer

2019, Microsoft released InterpretML [124], an open-source Python package which exposes interpretable machine learning algorithms to practitioners and researchers. One of the contributions in InterpretML is the first implementation of the Explainable Boosting Machine (EBM). EBM is an implementation of a GAM that uses a boosting mechanism to learn the univariate functions and can also model the most important pairwise interactions among features. EBM is interpretable, yet can be nearly as accurate as many blackbox models such as XGBoost.

5.2.4 Evaluation Metrics

There exist three popular metrics to evaluate probabilistic classifiers: area under the ROC curve (AUROC), Brier score, and logarithmic loss. We first discuss the three evaluation metrics in detail and then offer our insights on when to use which metric.

AUROC The area under the receiver operator curve (AUROC) evaluates how well a classifier can differentiate positive examples from negative examples. Intuitively, AUROC answers the following question: "Given a positive example and a negative example, how likely is it that our classifier will correctly rank the positive example ahead of the negative example?". Note that even random guessing will achieve an AUROC of 50%. This offers a naive baseline any probabilistic classifier should always beat. One crucial aspect of AUROC that often goes ignored is that it is in essence a ranking metric. AUROC only considers the relative ranking of examples and ignores the actual predicted probabilities. This means that a classifier can be poorly calibrated, yet still achieve great AUROC.

Brier score The Brier score (BS) is a proper scoring rule that measures the accuracy of probabilistic predictions. A proper scoring rule is a metric that can only be minimized by reporting the true class distribution. It is essentially the mean squared error between the predictions and the labels and has the following formula:

$$BS = \frac{1}{N}\sum_{i}^{N}(p_i - y_i)^2$$

in which N is the number of examples, p_i is the probability that was predicted for example i and y_i is the label (0 or 1) of example i.

Logarithmic loss The logarithmic loss (LL) is also a proper scoring rule that measures the accuracy of probabilistic predictions. The biggest difference

with Brier score lies in the way that it weighs individual prediction errors. Logarithmic loss has strong foundations in information theory and its formula is:

$$LL = \frac{1}{N}\sum_{i}^{N} y_i \log p_i + (1 - y_i)\log(1 - p_i).$$

A common oversight in soccer analytics research is building predictive models by optimizing and evaluating on one of the above metrics without critical thought on why to use a specific evaluation metric. Some works [127, 47, 108] overlook the fact that the choice of evaluation metric should depend on the specific use case in which the predictive model will be used. We now discuss our insights on when each metric is applicable.

Choosing whether or not to use AUROC is relatively straightforward. AUROC is the best metric for (a) classification tasks or (b) ranking examples based on how likely they are to be positive or negative. An example use case is using the outputted probabilities of a learned model to determine the top-k game states that are most likely to result in a goal.

When we care about using the actual values of the probabilities, the choice is between the Brier score and logarithmic loss as AUROC is not suitable. Unfortunately, it is less clear when one should use the Brier score versus logarithimc loss. Brier score and logarithmic loss are similar in the sense that they are both proper scoring rules and can both only be minimized by reducing the individual prediction errors. However, they differ in how they aggregate the individual prediction errors.

To illustrate this difference and to more easily compare the two metrics, let $e_i = |p_i - y_i|$ be the prediction error for example i. Using this definition and the multiplication rule for logarithms, we can simplify the formulas for the Brier score and logarithmic loss to:

$$BS = \frac{1}{N}\sum_{i}^{N} e_i^2$$

and

$$LL = \frac{1}{N}\sum_{i}^{N} \log(1 - e_i) = \frac{1}{N}\log(\prod_{i}^{N} 1 - e_i).$$

This rewrite illustrates howthe Brier score is simply the mean squared error. Moreover, the Brier score combines individual prediction errors by summing them while the logarithmic loss combines individual prediction errors by multiplying them.

This insight is the reason we recommend to use Brier score to build a predictive model if the resulting probabilities will be summed or subtracted. For example, in this chapter we construct player ratings by summing the deltas between game state probabilities and thus use Brier Score to evaluate the probabilistic classifiers. We recommend to use logarithmic loss if the resulting probabilities from the predictive model are more likely to be used in multiplications, such as in [47] and [60], where the resulting probabilities are multiplied with the probabilities of predictive models of other tasks. Other use cases where probabilities are often used in multiplications are simulations, reinforcement learning, and recommender systems.

In summary, which evaluation metric to use depends on what the probabilities outputted by the predictive model will be used for. We recommend to use AUROC when *ranking* probabilities or *classifying* examples, Brier score when *summing* or *subtracting* probabilities, and log loss when *multiplying* or *dividing* probabilities.

5.3 Experiments

Evaluating the VAEP framework is challenging as no objective ground truth action values or player ratings exist. Therefore, our experiments address three main questions: (1) providing intuitions into how our framework behaves and compares to other metrics, (2) presenting use cases revolving around player acquisition and characterization, and (3) evaluating several of our design decisions.

In most of this section, we focus our analysis on Wyscout data in the SPADL format for the English, Spanish, German, Italian, French, Dutch, and Belgian top divisions. We apply the VAEP framework to 11565 games played in the 2012/2013 through 2017/2018 seasons. We only consider league games and thus ignore all friendly, cup, and European games.

To produce scoring and conceding probabilities, action values and player ratings, we trained two classification models using the features and labels detailed in section 5.2 and the CatBoost algorithm, which is a gradient boosting decision tree algorithm extremely similar to XGBoost [35, 139]). We trained the first model on the 2012/2013 through 2015/2016 seasons to produce the outcomes for the 2016/2017 season. Similarly, we trained the second model on the 2012/2013 through 2016/2017 seasons to produce the outcomes for the 2017/2018 season.

In the last two parts of this section (5.3.6 and 5.3.7), we train and evaluate a model on StatsBomb data in the SPADL format of the 2017/18 and 2018/19

	TIME	PLAYER	ACTION	P_{scores}	VALUE
1	92m4s	S. Busquets	pass	0.03	0.00
2	92m6s	L. Messi	pass	0.02	- 0.01
3	92m8s	S. Busquets	pass	0.03	+ 0.01
4	92m11s	L. Messi	take on	0.08	+ 0.05
5	92m12s	L. Messi	pass	0.17	+ 0.09
6	92m14s	A. Vidal	shot	1.00	+ 0.83

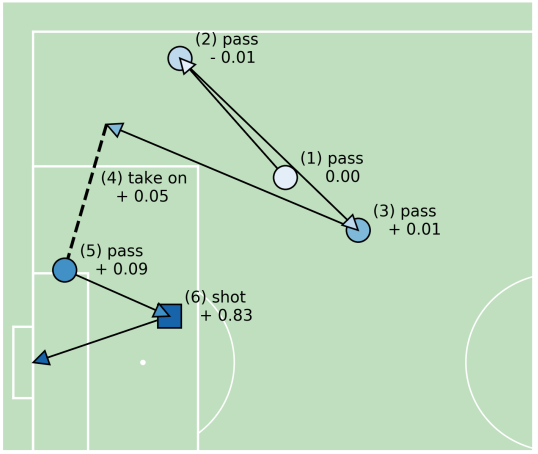


Figure 5.1: The attack leading up to Barcelona's final goal in their 3-0 win against Real Madrid on December 23, 2017.

English Premier League.

5.3.1 Intuition Behind the Action Values

Figure 5.1 illustrates how our framework works by visualizing the actions and their corresponding values that led to Barcelona's goal in the 93rd minute of their game away against Real Madrid on December 23, 2017.

The attack consists of six actions and starts with Sergio Busquets passing the ball towards the right flank (1), which receives a neutral action value of 0.00 since it neither improves nor worsens the situation. The subsequent pass from Lionel Messi back to Busquets (2) is penalized with an action value of -0.01 since it moves the ball backwards to a less favorable position than before. Busquet's excellent through ball to Messi (3), which finally moves the ball closer to the goal, receives an action value of +0.01. Messi receives the ball and dribbles

past a Real Madrid defender into the box (4), which receives an action value of +0.05 for significantly raising the scoring odds from 0.03 to 0.08.

Messi's next action showcases his genius, passing the ball backwards and away from the crowded six yard box (5). Our framework awards this pass a value of +0.09 for raising the scoring odds from 0.08 to 0.17. This action shows the power of our framework, which rewards Messi for moving the ball away from the opponent's goal. In a purely data-driven way, our framework identifies this action to be a good choice given the circumstances. To our knowledge, no other method for valuing actions using event stream data would reward Messi for this action. Finally, Aleix Vidal shoots the ball in (6). For converting a 0.17 scoring chance to a goal, our framework rewards Vidal with an action value of +0.83. If Vidal had missed his shot, he would have been penalized with an action value of -0.17.

5.3.2 Comparing our VAEP Player Ratings to Traditional Player Performance Metrics

Currently, players' offensive contributions are usually quantified by counting goals and assists, as those events directly influence the score line.[2] Therefore, we compare our VAEP player ratings against the following three baseline metrics: goals per 90 minutes, assists per 90 minutes, and goals + assists per 90 minutes. We investigate these metrics' capabilities to identify top players by producing each metric's top-10 list for the 2017/2018 English Premier League season, which are shown in Table 5.1. The top-10 in terms of goals per 90 minutes consists of strikers who focus on finishing rather than creating scoring chances. Similarly, the top-10 in terms of assists per 90 minutes mostly consists of midfielders who primarily specialize in setting up chances for their teammates. The ranking in terms of goals + assists per 90 minutes appears to strike a balance between both archetypes.

However, our framework also identifies impactful players who do not rate high on these traditional metrics. First, the top-10 list produced by our VAEP framework features Kevin De Bruyne (Manchester City), Eden Hazard (Chelsea), and Riyad Mahrez (Leicester City). Although considered Premier League stars, they do not appear in any of the traditional top-10s. Second, the combined market value[3] for the players in our top-10 list (1,110 million euro) is considerably higher than that for goals (862 million euro), assists (760 million euro), and goals + assists (947 million euro).

[2] `https://www.squawka.com/en/news/every-player-with-10-goals-and-10-assists-in-europes-top-five-leagues-this-season-ranked-by-contribution-per-90/1031863`

[3] According to `https://www.transfermarkt.co.uk/` on February 1st, 2019.

These observations suggest that our VAEP framework captures players' contributions to their teams' performances better than the traditional player performance metrics.

5.3.3 Identifying Promising Young Players and Minor League Talent

The English and Spanish leagues are the toughest and wealthiest by far. Hence, young players struggle to earn playing time, which forces the clubs to sign promising youngsters from smaller leagues such as the French, Dutch, and Belgian leagues. Typically, it is easier and especially cheaper for English and Spanish clubs to acquire promising youngsters from these leagues than from direct rivals. Therefore, we investigate the top-ranked young talents (i.e., players born after January 1, 1997 who played at least 900 minutes) separately for the 2017/18 season in the English and Spanish leagues (Table 5.2a), and the French, Dutch and Belgian leagues (Table 5.2b).

Marcus Rashford, who was linked with a € 110 million move to Real Madrid in January 2019,[4] and Ousmane Dembélé, who moved to Barcelona in August 2017 for a fee of € 120 million, are the most notable players in Table 5.2a. In contrast, the fourth-ranked but lesser-known Jonjoe Kenny has a much lower estimated market value than both of these players due to two reasons. First, Kenny is a defensive player, who are typically valued lower than offensive players by clubs and fans. Second, Kenny plays for mid-table club Everton, where he is surrounded by only a few world-class players. Nevertheless, our player ratings suggest a much higher valuation than his estimated market value of €5 million on February 1st, 2019.

David Neres tops Table 5.2b. In the summer of 2017, the winger became the fourth most expensive incoming transfer in the Dutch league when Ajax acquired him for € 15 million. He was a transfer target for top clubs Liverpool, Chelsea, and Arsenal, who all wished to sign him in the summer of 2019. Second-ranked Mason Mount was with Dutch side Vitesse on a season-long loan from Chelsea and received Vitesse's Player of the Year Award. Fourth-ranked Kylian Mbappé won the Best Young Player Award at the 2018 World Cup, while both Malcom (summer 2018, fee € 41 million) and Frenkie de Jong (summer 2019, fee € 75 million) have signed with Barcelona.

Tables 5.2a and 5.2b demonstrate our framework's ability to serve as a useful tool for talent scouts. Our framework can generate rankings for each league in

[4] `https://www.thesun.co.uk/sport/football/8318008/real-madrid-marcus-rashford-transfer-man-utd/`

Table 5.1: The top-10 players who played at least 900 minutes in the 2017/2018 English Premier League season in terms of (g) goals, (a) assists, (g+a) goals + assists, and (vaep) our VAEP player ratings. R_m denotes the rank of the player out of 305 players for metric m. The market value denotes the player's market value on February 1st, 2019 according to Transfermarkt.

(a) Top-10 players in terms of goals per 90 minutes (g/90)

R_g	Player	g/90	R_{vaep}	Market Value
1	M. Salah	0.986	2	€ 150m
2	S. Agüero	0.960	14	€ 75m
3	P. Aubameyang	0.851	42	€ 75m
4	H. Kane	0.847	9	€ 150m
5	G. Jesus	0.700	204	€ 70m
6	O. Niasse	0.666	17	€ 7m
7	R. Sterling	0.625	7	€ 120m
8	C. Austin	0.612	117	€ 10m
9	A. Lacazette	0.570	49	€ 65m
10	P. Coutinho	0.565	1	€ 140m

(b) Top-10 players in terms of assists per 90 minutes (a/90)

R_a	Player	a/90	R_{vaep}	Market Value
1	H. Mkhitaryan	0.484	114	€ 30m
2	P. Coutinho	0.484	1	€ 140m
3	L. Sané	0.482	47	€ 100m
4	K. De Bruyne	0.467	3	€ 150m
5	D. Silva	0.369	13	€ 25m
6	R. Sterling	0.347	7	€ 120m
7	P. Aubameyang	0.340	42	€ 75m
8	M. Özil	0.333	15	€ 35m
9	P. Pogba	0.332	8	€ 80m
10	C. Brunt	0.327	73	€ 2m

(c) Top-10 players in terms of goals + assists per 90 minutes (g+a/90)

R_{g+a}	Player	g+a/90	R_{vaep}	Market Value
1	S. Agüero	1.235	14	€ 75m
2	M. Salah	1.232	2	€ 150m
3	P. Aubameyang	1.191	42	€ 75m
4	P. Coutinho	1.049	1	€ 140m
5	R. Sterling	0.972	7	€ 120m
6	H. Kane	0.905	9	€ 150m
7	L. Sané	0.853	47	€ 100m
8	G. Jesus	0.808	204	€ 70m
9	A. Martial	0.795	6	€ 60m
10	O. Niasse	0.749	17	€ 7m

(d) Top-10 players in terms of our VAEP player ratings

R_{vaep}	Player	Rating	R_g	R_a	R_{g+a}	Market Value
1	P. Coutinho	0.899	10	2	4	€ 140m
2	M. Salah	0.817	1	23	2	€ 150m
3	K. De Bruyne	0.641	72	4	15	€ 150m
4	E. Hazard	0.636	21	122	34	€ 150m
5	R. Mahrez	0.635	34	11	16	€ 60m
6	A. Martial	0.607	13	13	9	€ 60m
7	R. Sterling	0.579	7	6	5	€ 120m
8	P. Pogba	0.549	55	9	28	€ 80m
9	H. Kane	0.545	4	140	6	€ 150m
10	S. Heung-Min	0.539	19	36	17	€ 50m

Table 5.2: The top-5 players born after January 1, 1997 in terms of our VAEP player ratings during the 2017/2018 season in (a) the tougher English and Spanish leagues, and (b) the smaller French, Dutch, and Belgian leagues.

(a) Young talents in the English and Spanish leagues.

Rank	Name	Team	Age	Rating	Market Value
1	M. Rashford	Man United	20	0.406	€ 65m
2	T. Alexander-Arnold	Liverpool	19	0.405	€ 45m
3	O. Dembélé	Barcelona	20	0.360	€ 80m
4	J. Kenny	Everton	21	0.344	€ 5m
5	M. Oyarzabal	Real Sociedad	21	0.337	€ 40m

(b) Young talents in the French, Dutch, and Belgian leagues.

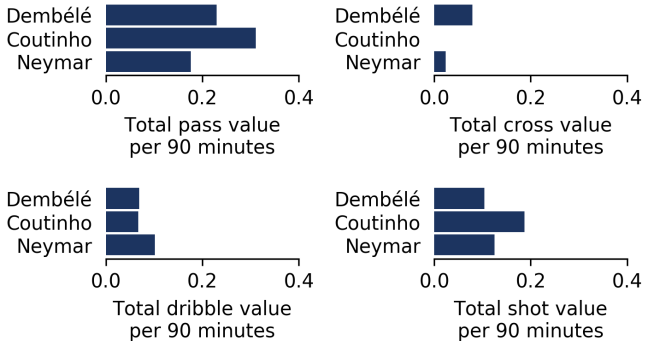
Rank	Name	Team	Age	Rating	Market Value
1	D. Neres	Ajax	21	0.620	€ 25m
2	M. Mount	Vitesse	19	0.616	€ 4m
3	Malcom	Bordeaux	21	0.567	€ 40m
4	K. Mbappé	PSG	19	0.507	€ 200m
5	F. de Jong	Ajax	20	0.495	€ 60m

the world (e.g., second divisions or leagues in North America, South America, and Asia) given that the required event stream data is available.

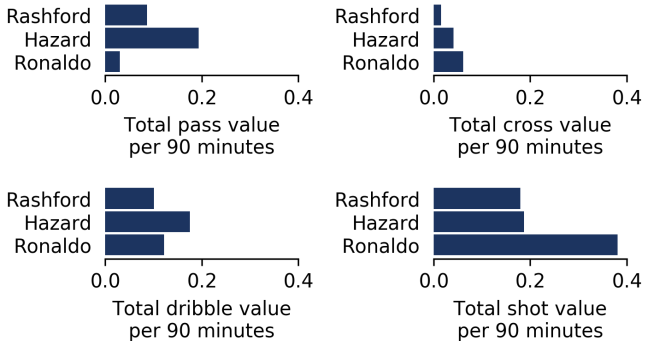
5.3.4 Characterizing Playing Style

Clubs are increasingly considering player styles during the recruitment process to identify players who best suit their team's preferred style of play (e.g., short passes and high defending vs. long balls and defensive play). Currently, scouts are typically tasked with judging playing styles with the naked eye. However, these scouts' time is often the limiting resource, which makes it difficult to consider the entire pool of candidate reinforcements. Therefore, metrics that assess a player's ability to perform different types of actions can help select a relevant set of players who are worth extra attention. Using our VAEP framework, addressing this task boils down to computing a player's rating per 90 minutes for each type of action.

As a concrete use case, consider Barcelona's attempts in the summer of 2017 to offset the loss of Neymar by acquiring Borussia Dortmund's Ousmane Dembélé and Liverpool's Philippe Coutinho. Figure 5.2a compares Dembélé, Coutinho



(a) Playing style of replacement players for Neymar



(b) Playing style of replacement players for Ronaldo

Figure 5.2: Overview of the total contribution per 90 minutes for different types of actions for (a) Neymar, Ousmane Dembélé, and Philippe Coutinho during the 2016/2017 season, and (b) Cristiano Ronaldo, Marcus Rashford, and Eden Hazard during the 2017/2018 season.

and Neymar's total ratings per 90 minutes for four action types. According to our metric, both Dembélé and Coutinho's passes receive a higher value than Neymar's while Neymar is a superior dribbler. From a stylistic perspective, this breakdown suggests that both Dembélé and Coutinho were reasonable targets as not many players come close to replicating Neymar's signature skill of dribbling. Dembélé and Coutinho are decent dribblers and better passers than Neymar. In addition, Dembélé outperforms Neymar in crossing, while Coutinho outperforms him in shooting.

Similarly, Real Madrid lost their all-time top scorer Cristiano Ronaldo in the summer of 2018. In the subsequent season, the struggling club appeared in desperate need of a suitable replacement. Manchester United's Marcus Rashford

and Chelsea's Eden Hazard were both linked with moves to Madrid. However, Figure 5.2b shows that neither came close to replicating Ronaldo's incredible finishing skill. Moreover, Ronaldo exhibits a higher total shot value per 90 minutes than Rashford and Hazard combined. While Hazard outperforms Rashford in every aspect, Rashford is closer to Ronaldo in terms of style as both rate similarly for passing and dribbling. Given that Real Madrid acquired Eden Hazard in summer 2019, we can assume that they preferred a more experienced player that could immediately strengthen their team, even if he was less compatible with their playing style, over the younger Rashford who would allow them to stick with the same playing style as in the Ronaldo-era.

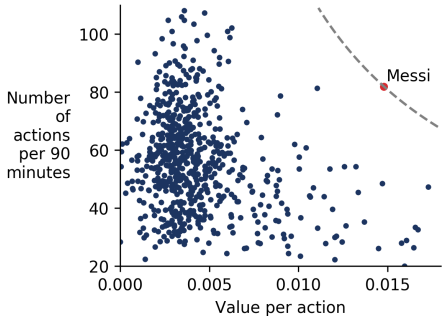
5.3.5 Trading Off Action Quality and Quantity

A natural tension exists between the quality and quantity of actions. If a player performs a high number of actions, then it is harder for each action to have a high value. Figure 5.3a shows the number of actions that players execute on average per 90 minutes (quantity) and the average value of these actions (quality) for those players who played at least 900 minutes during the 2017/18 season in the Spanish and English leagues. The grey-dotted isoline shows the gap in VAEP rating between top-ranked Lionel Messi and the rest. The isoline is curved as a player's rating is obtained by multiplying the average value per action (*x-axis*) and the average number of actions (*y-axis*). As shown by the isoline and more traditional statistics,[5] Messi is clearly in a class of his own.

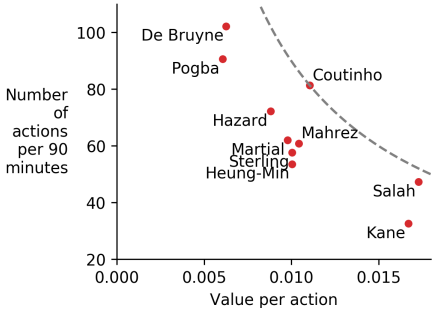
Zooming in on Figure 5.3a, Figure 5.3b shows the top-10 players in the 2017/18 English Premier League season. Strikers Harry Kane and Mohammed Salah perform a relatively low number of actions but their actions are highly valued on average. Midfielders Kevin De Bruyne and Paul Pogba perform more actions albeit with a lower average value per action. Philippe Coutinho, Eden Hazard, Riyad Mahrez, Anthony Martial, Raheem Sterling, and Son Heung-min fall in between these two archetypes, hitting a sweet spot between the quality and quantity of their actions.

Similarly, Figure 5.3c shows the top-10 players in the Spanish league. We observe the same archetypes as for the English league. Strikers Cristiano Ronaldo, Antoine Griezmann, Gareth Bale, Enis Bardhi, Iago Aspas, and Cédric Bakambu perform a low number of highly valuable actions. Real Madrid midfielders Toni Kroos and Isco perform more actions that are less valuable. Philippe Coutinho, who appears in both figures following his move from Liverpool to Barcelona in January 2018, again hits the sweet spot between action quality and quantity.

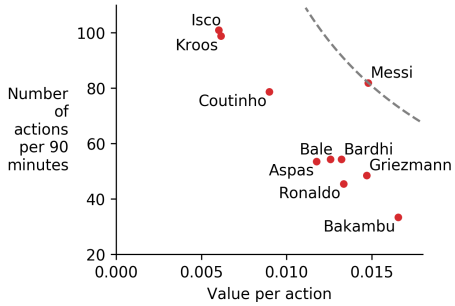
[5] `https://fivethirtyeight.com/features/lionel-messi-is-impossible/`



(a) All players



(b) Top-10 players in the English league



(c) Top-10 players in the Spanish league

Figure 5.3: Scatter plots of players in the 2017/18 season who played at least 900 minutes in the Spanish or English league. The plots contrast the average number of actions performed per 90 minutes with the average value of the actions of the player. As shown by the grey-dotted isoline in (a) and (c), Lionel Messi is clearly in a class of his own.

Lionel Messi is an outlier in the sense that he rates high on action quality and quantity at the same time.

5.3.6 Interplay between Feature Sets and Classifiers

A common challenge in data science is to evaluate the performance of a system. While it can be easy to define a high-level task such as assigning values to actions, one underappreciated aspect of evaluating the solution is that usually no ground truth exists and standard evaluation metrics such as accuracy, precision and recall thus cannot be used. As a result, the only way to evaluate a system is to evaluate the components it consists of. In our case, we evaluate the action values by evaluating the underlying scoring probabilities for which ground truth

labels are available. The goal of this subsection is to understand the effect of the interplay between the complexity of the feature set used to describe the game state and the selected probabilistic classifier used to estimate scoring probabilities for each game state.

We evaluate the performance of three classifiers: Logistic Regression [129], XGBoost [35], and Generalized Additive Models [124]. For all three classifiers, we perform no tuning and set all parameters to the default values of their respective implementations.[6][7][8] Some examples of these default parameters are Logistic Regression using the L2 regularization penalty and L-BFGS as the optimization problem solver, XGBoost using 100 trees of maximum depth 6, and the GAM using 16 estimators to construct each univariate function. The only exception is that we allowed the GAM to learn three pairwise interaction terms rather than its default value of zero. This parameter was changed to leverage the full capabilities of the underlying implementation of the GAM.

For each classifier, we also consider the following three sets of features:

Location only The (x, y)-coordinates of the last action in game state S_i.

VAEP The set of 151 features detailed in Section 5.2.

Top-10 The top-10 most important features from the VAEP feature set. These features were selected using the built-in ordering of feature importance available in the implementations of the XGBoost and GAM models.

Our data set for this experiment consists of event stream data of 760 matches from seasons 2017/18 and 2018/19 of the English Premier League. The data was provided to us by StatsBomb and then converted to the SPADL format. We trained each classifier-feature set combination on 747,813 game states in the 2017/18 Premier League season and evaluated them using the Brier score on 789,108 game states in the 2018/19 Premier League season.

We evaluate the performance of each approach using the Brier score, which measures the accuracy and calibration of the predictions and is minimized when the true underlying probability distribution of the data is reported [63]. This property is important because we sum and subtract the predicted probabilities to generate action values.

Table 5.3 reports the Brier scores for each classifier-feature set combination. From these results, we can infer the following conclusions:

[6] `https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html`

[7] `https://github.com/dmlc/xgboost`

[8] `https://github.com/interpretml/interpret`

Features	Classifier		
	LogReg	XGBoost	GAMs
Location only	0.01112	0.01087	0.01087
VAEP	0.01010	0.00966	0.00968
Top-10	0.01030	0.00970	0.00971

Table 5.3: Brier score (lower is better) of three different feature sets using three different probabilistic classifiers: Logistic Regression (LogReg), XGBoost, and Generalized Additive Models (GAMs).

Only considering location is insufficient. As can be seen in the first row of Table 5.3, Logistic regression achieves a Brier score of 0.01112, while both XGBoost and GAMs achieve a Brier score of 0.01087. These scores only slightly improve upon the Brier score obtained when using the class prior as the predicted probability (0.01128), a common baseline in probabilistic classification tasks. Furthermore, regardless of the learning method, only using the location does not come close to matching the performance of using a more expansive and expressive feature set.

Having a model that captures non-linearities helps. Regardless of the feature set, XGBoost and GAMs offer substantial improvements on predictive performance compared to using logistic regression. For the location only feature set, Figure 5.4 clearly demonstrates how GAMs are capable of capturing non-linear correlations between the features and the target variable, while Logistic Regression is not.

A small feature set can yield excellent performance. Compared to seeing the full set of 151 features (second row in Table 5.3), using the top 10 features only slightly decreases performance (third row in Table 5.3). However, using fewer features highly enhances the interpretability of these models. The Brier score of the GAM (0.00968) is again similar to the performance of XGBoost (0.00966).

In summary, regardless of the feature set GAMs achieve a performance that is better than Logistic Regression and comparable to that of XGBoost, while remaining interpretable.

5.3.7 Discussion of the Top-10 Features

Figure 5.5 details the ten univariate functions that make up the GAM that almost matches XGBoost's performance and provides some insights in what

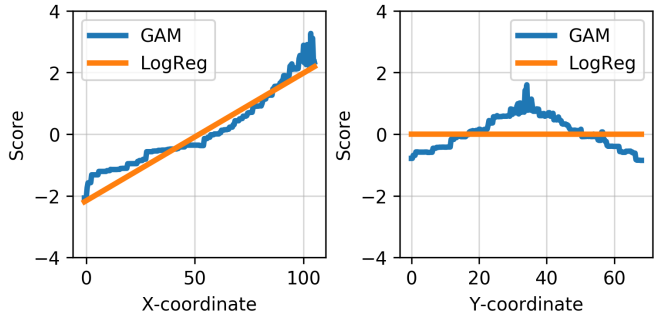


Figure 5.4: A Generalized Additive Model (GAM) consisting of two univariate models using (a) the x-coordinate and (b) the y-coordinate of a game state. The GAM predicts the probability of scoring a goal from a given x, y-location by summing the scores of the univariate models per feature and converting the resulting sum to a probability $P \in [0, 1]$ using the *logit* linking function. The straight orange line represents the weight given to the feature by a Logistic Regression model (LogReg) and illustrates the difference in predictive power.

makes a game state (and an action) likely to result in a scored goal. The figure also includes the scores of an example game state to illustrate how a GAM can be used to understand the reasoning behind individual predictions.

Panels (a-c) capture location-based aspects of the game state. In Panel (a), we see that as the ball gets closer and closer to the center of the opponent's goal, the chances of scoring increase. This correlation increases dramatically when the ball is very close to the center of the goal. In Panel (b), we can see that being aligned to the center of the goal slightly increases the chance of scoring whereas being at a tight angle decreases it. This makes sense, as a shooter likely has more places to aim when positioned in front of the goal. Finally, Panel (c) shows that when the ball enters the final third, there is a strong positive correlation with scoring. This increases as the ball gets closer to the endline behind the opponents goal, but not as dramatically as in Panel (a).

Panels (d-f) capture contextual aspects of the game state. Panel (d) shows how the probability of scoring is dramatically reduced if the last action was not successful. This makes sense, as in the SPADL representation an action not being successful means that the team has lost possession of the ball and therefore cannot attempt to score without first regaining possession.

Panel (e) shows an even stronger impact on goal scoring probability if the last action was a foul. The effect captured here is that while there might still be a

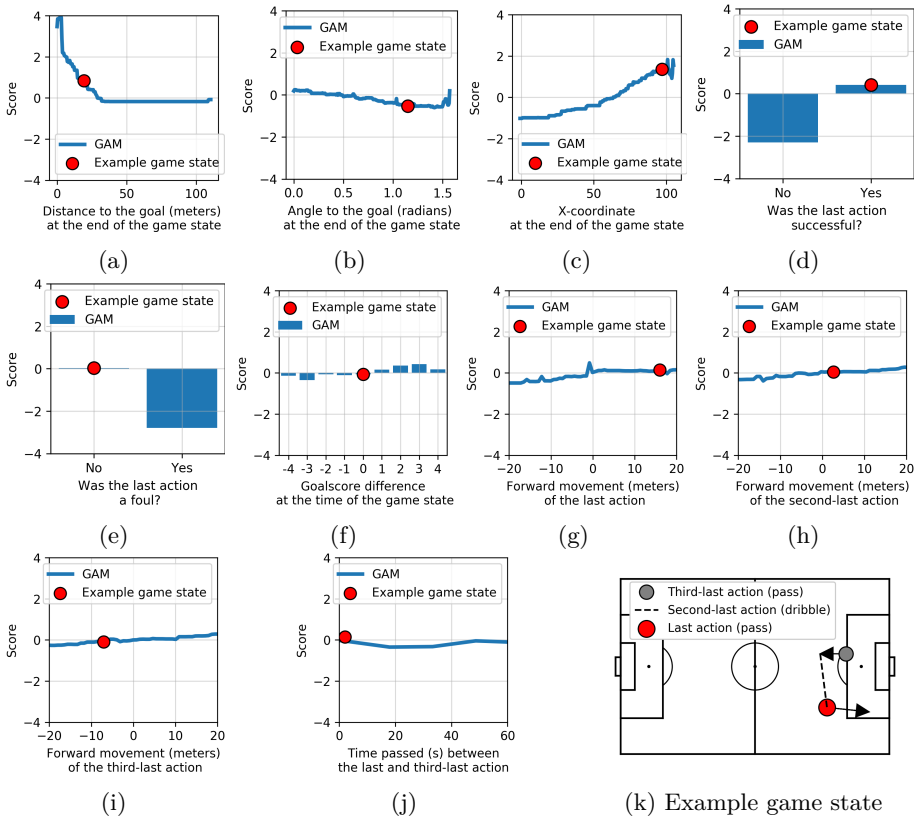


Figure 5.5: Univariate functions of the ten features (a-j) that make up the GAM that predicts the short-term goal-scoring probability from a given game state. The red dots denote the scores of an example game state (k) with a short-term goal-scoring probability of 4.9%. This goal-scoring probability is computed using the formula $g^{-1}(\sum_i^m f_i(x_i))$ where $f_i(x_i)$ are the scores in Panels (a-j) and g^{-1} is the inverse logit function. The two features that have the biggest positive impact on the goal-scoring probability of the example game state are (a) the small distance to the goal and (c) the high X-coordinate. The feature with the biggest negative impact is (b) the bad angle to the goal.

slim chance that a team can quickly recover the ball following an unsuccessful action, this becomes virtually impossible if the team committed a foul. The reason is that after a foul the game temporarily suspends and is no longer in open play. This allows the players of the opposing team who are now in possession of the ball enough time to position themselves such that they obtain the maximum tactical advantage.

Panel (f) shows that the probability of scoring varies based on the score difference, with teams leading by ≥ 2 being more likely to score. Robberechts et al. [144] showed that the probability of scoring a goal changes with the goalscore difference. However, we currently are unsure of whether this is a causal effect (i.e., being three goals ahead or tree goals behind has an effect on the mental state of a player, making them perform better or worse actions [21]) or whether this is simply a correlation (i.e., a team that is three goals ahead in a match is a good team with an above average finishing rate, therefore its games states are more valuable). Researching this further is an interesting direction for future work.

Panels (g-j) capture aspects about the speed of play. On Panels (g-i) as the values on the x-axis increase, it indicates that the ball is moving longer distances and hence getting closer and closer to your opponent's goal. In Panel (j), as the value on the x-axis decreases, this indicates that there is less time between consecutive actions. This may be a proxy for the ball moving more rapidly. Hence, in combination, these last four features hint towards the speed of play during the game state. This can be an important factor to decide goal-scoring probability, i.e., the odds of scoring are usually higher during a quick counter-attack than during slow build-up play.

For each of the three location-based features in Panels (a-c), the GAM also learns a pairwise interaction term where it combines each feature with the successfulness of the action in Panel (d). These interaction terms help fine-tune the performance of the GAM for specific examples, but are also more challenging to interpret than the simple univariate functions in Figure 5.5.

5.4 Related Work

While valuing player actions in soccer is an important task, it has remained virtually unexplored due to the challenges resulting from the dynamic and low-scoring nature of soccer. The approaches from [125], [21] and [60] for soccer, [146] and [107] for ice hockey, and [32] for basketball come closest to our framework. Most of these approaches address the task of valuing individual actions by modeling a game as a Markov game [105]. In contrast to [125] and

[146], which divide the pitch into a fixed number of zones, our approach models the exact locations of each action. Unlike [32], which values only three types of on-the-ball actions, our approach considers any relevant on-the-ball action during a game. However, our definitions of player actions, game states, and action values are similar to those used by these works as well as earlier research for soccer [147, 86], American football [76], and baseball [170].

Most of the related work on soccer either focuses on a limited number of player action types like passes and shots or fails to account for the circumstances under which the actions occurred. [51], [98], and [79] address the task of valuing the actions leading up to a goal attempt, whereas [24], [22], and [84] address the task of valuing individual passes. The former approaches naively assign credit to the individual actions by accounting for a limited amount of contextual information only, while the latter approaches are limited to a single type of action.

Furthermore, this work is also related to expected-goals models, which estimate the probability of a goal attempt resulting into a goal [108, 27, 6, 110, 47]. In our VAEP framework, computing the expected-goals value of a goal attempt boils down to estimating the value of the game state prior to the goal attempt.

5.5 Discussion of Remaining Challenges

One limitation of our VAEP framework is that we only value on-the-ball actions. That is, the model only values actions with the ball, while defending is often more about preventing your opponent from gaining possession of the ball by clever positioning and anticipation.

Another challenge is that it is hard to accurately compare players across leagues, as it is easier to perform highly valuable actions in minor leagues (e.g., French, Dutch, and Belgian) than in tougher leagues (e.g., English and Spanish). This can be clearly observed in Section 5.3.3 where the young talents in the minor leagues receive a higher rating than those in the English and Spanish leagues.

Similarly, it can even be hard to accurately compare players across clubs in the same league as it is generally easier to perform valuable actions in a top club with strong teammates, than in a mid-table club with weaker teammates.

The final challenge for deploying our framework in the real world is building trust in the ratings as traditional scouts are unfamiliar with our way of rating soccer players. In addition, our ratings are slightly less intuitive than traditional metrics such as goals per 90 minutes, which complicates the task for analytically less inclined scouts to understand what our ratings measure precisely.

5.6 Conclusion

This chapter introduced VAEP, a framework for assigning a value to each individual player action during a soccer game. The advantages of VAEP over most existing works are that it (1) values all action types (e.g., passes, crosses, dribbles, shots), (2) considers both the offensive and defensive value of actions, (3) bases its valuation on a more complete view of the game context, and (4) reasons about an action's possible effects on the subsequent actions. Intuitively, the player actions that increase a team's chance of scoring receive positive values while those actions that decrease a team's chance of scoring receive negative values.

Chapter 6

Automatic Discovery of Team Tactics

In this chapter, we propose an approach to find patterns in event stream data of professional soccer matches. On a high level, our approach performs the following five steps. First, the algorithm splits the match into phases, which are uninterrupted sequences of events where one team is in possession of the ball. Second, we cluster these phases by their spatio-temporal characteristics. Third, we rank each cluster according to their expected relevance to the user. Fourth, we search for frequently occurring patterns (i.e., sequences of events) within each cluster. Fifth, based on domain knowledge, we develop a ranking function that orders the discovered patterns in each cluster according to their expected relevance to the user.

We evaluate our approach on Opta event stream data from the 2015/2016 season of the English Premier League. We let a domain expert inspect the discovered patterns and find that our approach is capable of identifying interesting tactics. Furthermore, we evaluate how each of our design choices contributes to the overall performance of the system.

The content of this chapter is based on the following publication [50]:

Decroos, T., Van Haaren, J., and Davis, J. Automatic Discovery of Tactics in Spatio-Temporal Soccer Match Data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2018), pp. 223–232

Table 6.1: Special types of passes and their frequencies

Event type	Frequency
pass	100.00%
normal pass	67.72%
long ball	16.52%
head pass	8.03%
throw in	4.78%
cross	4.24%
free kick	2.47%
corner	1.11%
through ball	0.28%

6.1 Dataset

Our dataset consists of raw event stream data for the English Premier League for the 2015/16 season.[1] The dataset contains 652,907 events of 39 different types, which are related to either the flow of the match (e.g., a player substitution, a yellow card, an awarded corner) or the action on the pitch (e.g., a shot, a clearance, or a foul).[2] This corresponds to an average of 1,718 events per match. The most frequent event types in our dataset are "pass" (368,426), "out" (48,046), and "ball recovery" (41,448).

Passes are a special type of event. These can sometimes be of a more specific type, such as a "cross". This is stored as additional information. Table 6.1 shows an overview of the type of special passes, along with their frequency. Note that sometimes passes can have multiple specific types. For example, a pass can be both a "corner" and a "cross" at the same time.

Another special type of event are shots. These are registered in the data as four different event types based on their outcome: goal, miss, attempt saved or post (i.e., the ball hit the post). Hence, if we want to analyze the shots in our database, we have to aggregate over these event types.

[1] Note that in this chapter we do not use the SPADL format of Chapter 3, but instead use raw event stream data of a prominent vendor. The reason for this is that the contributions in this chapter chronologically precede the creation of the SPADL format. In fact, some of the challenges encountered in this chapter directly inspired the creation of SPADL.

[2] Because of the nature of raw Opta event stream data, we adopt the terminology of *events* and *event sequences* in this chapter, rather than *actions* and *action sequences* as in all other chapters in this dissertation.

6.2 Approach

The goal of our approach is to identify common attacking tactics employed by a specific team. Formally, our task can be defined as:

Given: A set of matches, where each match is represented as an event sequence, about a team of interest.

Find: Relevant spatio-temporal patterns that characterize attacking tactics.

Soccer is a highly dynamic game with many movements and interactions among players across time and space. In addition to the nine challenges outlined in Section 3.4, three other specific challenges arise when detecting tactics:

Challenge 10: Tactics involve coping with both a spatial component, as the location where an event occurs is important, as well as a temporal component, as the order of events is important.

Challenge 11: Identifying interesting and relevant patterns is a highly subjective decision. Certain sequences of play, such as high-quality attempts on goal occur infrequently, but are of high interest.

Challenge 12: Teams do not employ a single tactic. Each team has several different tactics during a game. Furthermore, each tactic has minor variations.

To tackle these challenges, we perform the following five steps.

1. Divide the event stream of each match into phases.
2. Cluster the phases based on their spatio-temporal component.
3. Rank the clusters based on the preferences of the user.
4. Mine each of the obtained clusters to identify frequent sequential patterns.
5. Rank the discovered patterns based on the preferences of the user.

The following subsections discuss each of these five steps in more detail.

6.2.1 Dividing a Match Event Stream into Phases

Formally, each match is represented as a sequence of events $[e_1, \ldots, e_n]$, where each e_i is an event and n is the total number of events. Each event e_i is a tuple $e_i = (t, l, p, et)$, where t is a timestamp, l is the location on the pitch where the event took place as given by its x and y coordinates, p is the set of players involved in the event, and et is the event type. We use ET to refer to the set of all possible event types.

In terms of tactics, a sequence representing an entire match represents too coarse of a granularity to consider for analysis. Tactics will manifest themselves as short, consecutive sequences of actions on the pitch such as attacking through the middle or playing a through pass. Therefore, a more natural unit to analyze is what a domain expert may call a "soccer gameplay phase" or simply a phase (e.g., a corner, an attack from the left flank, a turnover). A phase is a sequence of consecutive events that fit together. An added benefit of a phase representation is that it is easier to find patterns in multiple, shorter event sequences than one long event sequence. Therefore, we split the event stream of a match into phases in a similar manner as we did in Chapter 4. Figure 6.1 shows an example of a phase.

Each match is subdivided into phases, where each phase P is subsequence of consecutive events $[e_a, \ldots, e_b]$. A new phase starts if there

1. Is a pause of at least 10 seconds between events; or
2. Possession switches from one team to the other (e.g., a successful tackle, the ball goes out of play for a throw-in or corner kick, a goal is scored, or a free kick is awarded).

We only consider phases that have at least three events. Phases with only one or two events are usually not very informative about the playing style of a team. Figure 6.2 shows the distribution of phase lengths for Manchester City in the 2015-2016 English Premier League season. Finally, like in many sports, teams switch which goal they are attacking at halftime. Comparing phases that attack different goals is difficult, so we normalize each phase such that the team of interest is always playing left to right.

Another approach to divide the match event stream into phases is to divide the event stream into subsequences of constant length (e.g., windows of length 10 seconds) [47]. While this approach is more straightforward, it has two important drawbacks. First, the time between two consecutive events can differ greatly from one match to another due to a difference in intensity and the unreliability of human annotators. This would lead to many uninformative phases being

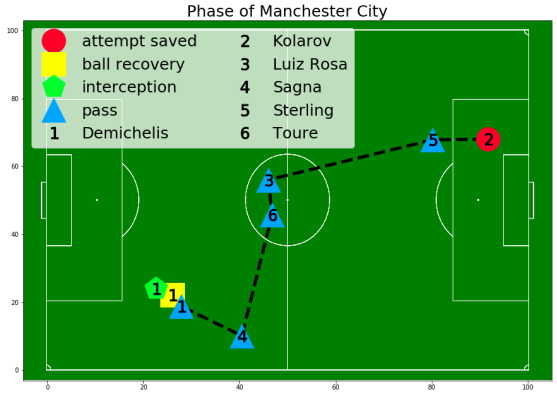


Figure 6.1: An example phase in our data. A phase is a sequence of consecutive events that fit together according to a domain expert.

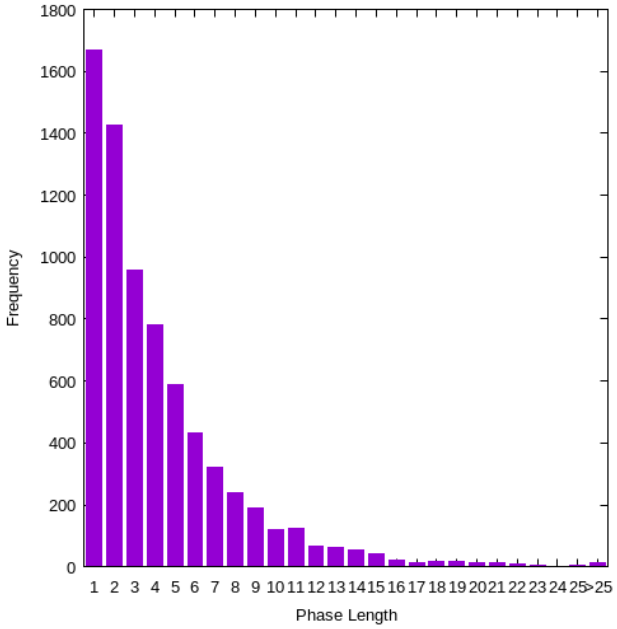


Figure 6.2: Distribution of phase lengths for Manchester City in the 2015-2016 English Premier League season.

constructed. Second, using this approach would result in many phases where both teams possessed the ball and performed actions during the phase, making it more difficult to infer the tactics of one specific team.

6.2.2 Clustering Phases on their Spatio-Temporal Component

The goal of the second step is to identify similar spatio-temporal phases via clustering. We do this for two reasons. One, this helps reduce the space of possible patterns that we need to search in step four. Two, a team is likely to employ multiple different attacking tactics, such as corners, attacking through the middle, down the flank, each of which will be characterized by different spatial characteristics. Clustering gives us a natural way to divide the data along these lines.

In this chapter, we use hierarchical agglomerative clustering, which is a popular and simple approach for clustering data [58]. To measure the distance between two clusters, we use the complete-linkage metric. Complete-linkage clustering tends to find compact clusters of approximately equal diameters but not necessarily equal number of objects [58], which is precisely the type of clustering we want.

The clustering works as follows. First, each element is assigned its own cluster. Next, clusters are iteratively merged together until a stop criteria is met. In each iteration, the two clusters separated by the shortest distance are combined. Complete-linkage clustering computes the distance between two clusters as the distance between those two elements (one in each cluster) that are farthest away from each other. In this work, we stop once there are k clusters remaining, where k is a user-defined parameter.

A crucial step in clustering objects is choosing the right distance function. In our setting, we have to cope with the fact that phases are of varying length. Furthermore, we want to identify spatially similar phases. One well known way to cope with these two desiderata is to use Dynamic Time Warping (DTW) [123]. DTW computes a cost by finding a one-to-many matching between two sequences. The matching allows for a non-linear warping effect when aligning the two sequences. Consequently, DTW is able to cope with minor mismatches between sequences, such as delays or shifts. A drawback to DTW is that it is not a distance function as it does not satisfy the triangle inequality.

The most natural way to explain and compute the DTW cost is via dynamic programming. In our case, given a phase P of length m and a phase P' of length n, the DTW cost can be computed as:

$$C_{i,j} = \delta(P_i, P'_j) + \min\{C_{i-1,j-1}, C_{i,j-1}, C_{i-1,j}\} \tag{6.1}$$

where C is the $m \times n$ cost matrix and $\delta(\cdot, \cdot)$ computes the cost of aligning the i^{th} event of P with the j^{th} event of P' as:

$$\delta(P_i, P'_j) = \sqrt{(P_{i,x} - P'_{j,x})^2 + (P_{i,y} - P'_{j,y})^2} \tag{6.2}$$

where $P_{i,x}$ ($P'_{j,x}$) and $P_{i,y}$ ($P'_{j,y}$) denote the x and y coordinates of the i^{th} (j^{th}) event in P (P'). That is, the cost function is only considering the spatial proximity of the events in two phases. The final DTW cost between the two phases P and P' is then given by $C_{m,n}$.

In earlier work [175], Van Haaren et al. clustered the phases based on their possession maps. A possession map is a grid overlaying the field that shows how often each area of the field was occupied by the players and the ball during a phase. Using DTW as a distance measure has two benefits over this earlier work: DTW is simpler and, unlike the possession map, it reasons about the sequential aspect of the phase.

6.2.3 Ranking Clusters

Next, we rank clusters according to their expected relevance to the user. Typically, the quality of clusters is judged by statistics such as average pairwise distance, maximal pairwise distance and minimal pairwise distance [57]. However, these evaluation functions are less likely to be relevant to a domain expert [177]. A soccer coach might be most interested in a cluster with phases that frequently lead to shots and goals. An opponent might be most interested in the clusters with the most phases, in order to identify and anticipate the most frequent patterns of play. Finally, a journalist might be interested in the clusters with the longest phases, as those can be the most interesting for sports fans. In this chapter, we went with the viewpoint of a soccer coach and rank clusters based on the number of shots that they contain.

6.2.4 Mining Patterns

The fourth step involves identifying frequent sequential patterns, that is, time-ordered sequences of events, within each cluster. One approach to mining patterns is applying techniques from inductive logic programming, which allows us to search for a rich set of patterns [174]. However, these techniques currently do not scale well to a large volume of data and are thus ill-suited for the event data analyzed in this chapter. We employ the CM-SPADE pattern miner [66], which is a more conventional sequential pattern mining algorithm found in the SPMF toolbox [67]. This pattern miner is more restrictive in the type of

learned patterns, but offers better scalability in terms of speed and memory. An important benefit of our approach is that our choice of pattern miner can be easily swapped for an alternative pattern miner such as GSP [163], PrefixSpan [131], etc.

Typically, sequential pattern miners take as input sequences, where each element in the sequence is an itemset (i.e., unordered set). An event contains a lot of information, and the key representational challenge is how to convert an event into an itemset. Deciding on an itemset representation requires considering two key questions:

Q1: What information to consider? For example, an end-user may care about knowing which players often play together, in which case the players involved in each event are important. However, teams rotate players between games, players may change positions during a game, and players are substituted within a game. Thus, some users may be interested in abstracting away from the specific players involved when considering a team's tactics. In this case, omitting the players' names from the representation is desirable.

Q2: How to encode the information? Each piece of information contained in an event could be represented in a multitude of different ways. For example, a player's position and the type of pass could be encoded hierarchically. The location could be represented as an exact position or as occurring in a specified zone of the pitch, and furthermore the zones could be organized hierarchically.

When thinking about how to encode the various components of an event, we pay special attention to two aspects of an event: its location and its type. Both aspects require some engineering to obtain good results.

Most pattern mining algorithms are designed to work with discrete data or to convert continuous attributes to discrete ones using a threshold (e.g., checking if the value is less than a threshold). The x and y coordinates of an event are real values, so we discretize the location. Rather than using a standard discretization method such as a grid [175], we divide the soccer pitch in zones based on domain knowledge as shown in Figure 6.3.

As mentioned in Section 6.1, shots and passes require some special care as they are important events and each one has multiple different types. For passes, we augment the itemset by adding any special type of pass as an extra event that is happening simultaneously. We treat shots in an analogous manner. Effectively, this introduces an extensional hierarchy in the data where an itemset can match

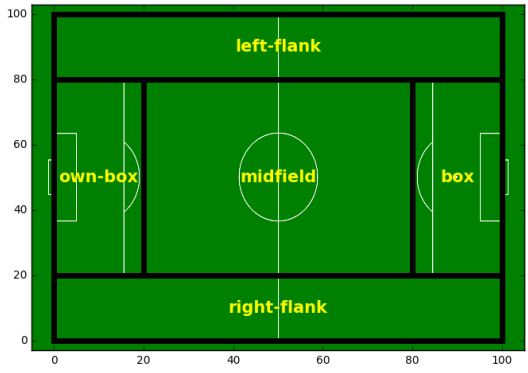


Figure 6.3: The zones used to discretize (x, y)-locations on the pitch.

Table 6.2: Sequence representation for the phase in Figure 6.1

1. An interception AT the right flank
2. A ball recovery AT the right flank
3. A pass FROM the right flank TO the right flank
4. A pass FROM the right flank TO the midfield
5. A pass FROM the midfield TO the midfield
6. A pass OR long ball FROM the midfield TO the box
7. A pass FROM the box TO the box
8. A shot OR attempt saved AT the box

on either the more generic event (e.g., a pass) or the more specific event (e.g., a through ball).

We want our patterns to be as readable as possible to a domain expert, therefore we encode them in natural language format. An event of type A at location X is encoded as `[A] AT [X]`, while an event of type B that moves the ball from location Y to location Z is encoded as `[B] FROM [Y] TO [Z]`. Table 6.2 shows the sequence that corresponds to the phase in Figure 6.1.

Using the discussed processing of locations and events, we consider the following five ways to represent an event as an itemset:

Location. This representation only considers the location of the event as determined by the zone in which it occurred.

Event Type. This representation only considers the type of the event, with the extra processing for passes and shots.

Player. This representation only considers players. Conceptually, each player is represented by a binary variable which takes on a value of 1 if the player participates in the event and 0 otherwise.

Location and Event Type. This representation constructs an itemset by combining (i.e., concatenating) the Location and Event Type representations.

Location, Event Type, and Players. This representation constructs an itemset by combining (i.e., concatenating) the Location, Event Type and Player representations.

Our primary evaluation will focus on the fourth representation and consider Locations and Event Types. The evaluation will explore how each of the representations affects the found patterns.

6.2.5 Ranking Patterns

Finally, we rank the discovered frequent sequential patterns with respect to their expected relevance to a user. Typically, frequent patterns are ranked according to their support in the data. However, this evaluation function is less relevant to soccer coaches. Given that most of the action during a soccer match typically happens in the middle of the pitch and that 90% of all events are passes, the top of the ranking is likely to be dominated by patterns describing passing sequences in the midfield.

We propose an alternative evaluation function that considers the types of the events appearing in a pattern, the length of the pattern and the pattern's support to determine its relevance. More specifically, we first assign a weight to each event type. Higher weights indicate higher relevance. This approach allows the user to define a bias towards a particular type of patterns. Specifically, we use a ranking function of the form:

$$Score(FS) = Supp(FS) \cdot \sum_{et \in ET} \lambda_{et} \cdot \#et \in FS$$

where FS is a frequent sequence, $Supp(SP)$ is the support count of the sequential pattern FS, λ_{et} is the weight assigned to event type et, and $\#et \in FS$ is the number of occurrences of event type et in SP. Given that we are mostly interested in goal attempts, we assign a high weight ($\lambda_{shot} = 2$) to shots, a low weight to normal passes ($\lambda_{pass} = 0.5$), and average weights ($\forall et \in ET \setminus \{shot, pass\} : \lambda_{et} = 1$) to all other types of events in our experiments.

6.3 Experimental Study

Our empirical evaluation on the dataset presented in Section 6.1 addresses the following four research questions:

Q1: Do we discover interesting and relevant patterns?

Q2: Can we characterize the tactics of teams?

Q3: What is the effect of the clustering step?

Q4: What is the best representation for phases?

The first two questions focus on evaluating the quality of our results, whereas the last two questions focus on assessing the impact of our design decisions on the overall results. Next, we discuss the methodology and present the results.

6.3.1 Methodology

The analysis is performed on a team-by-team basis. That is, all 38 league matches for a given team are used as input to the algorithm. The discussion will focus on the found patterns for Manchester City, Arsenal, and Leicester City. These patterns were evaluated qualitatively by a domain expert, as we do not have access to any ground truth data we can compare our patterns against. Tactics are often kept confidential by soccer clubs, so getting this ground truth data is nearly impossible.

We focus our attention on the top 10 clusters as ranked by the number of shots the cluster contains. We employ the CM-SPADE algorithm [66] in the SPMF toolbox [67] to discover frequent maximal sequential patterns in each cluster. We used a support threshold of 10, and then rank the found patterns according to our score metric. As a default, we consider 100 clusters and use the Location and Event Type itemset representation. For **Q3**, we consider 1, 10, 100, and 500 clusters. For **Q4**, we consider all five ways to convert an event into an itemset discussed in Subsection 6.2.4.

All experiments were run on a desktop with an Intel i7-6700 3.40GHz processor with four cores, each having two CPUs. The machine had 32 GB of memory.

6.3.2 Discovering Interesting and Relevant Patterns

Figure 6.4 shows the top 12 ranked clusters for Manchester City. The phases that appear within the same cluster exhibit a reasonable degree of spatial coherence. There are identifiable commonalities, such as that the top-right and top-middle clusters contain phases beginning in the opposition's right and left flank. Figure 6.5 shows a zoomed in version of the top-ranked cluster. In this cluster, several patterns were found that show a clear attacking pattern starting from the right flank. These involve actions such as passes followed by a cross, attacks from a corner, and set pieces. Similar patterns were found in the second-ranked cluster.

Figure 6.6 shows the fourth-ranked cluster, which is also interesting. The highest-ranked pattern in this cluster involved a ball recovery on the right flank, followed by a pass to the midfield, followed by a pass to the left flank. As seen from the cluster, this pattern is capturing a diagonal movement of the ball from the right side of Manchester city's own half to left side of their opponent's half. In the 2015/16 season, Mauricio Pellegrini commonly employed a formation that aligned Kevin De Bruyne on the right, David Silva in a central role, and Raheem Sterling on the left in support of striker Sergio Agüero. De Bruyne recovers many balls, especially for someone in that role. Sterling is very fast, and hence offers an outlet on the left side for a possible attack.

6.3.3 Identifying Team Tactics

To answer **Q2**, we compare the patterns found for three different teams: Arsenal, Leicester City, and Manchester City. We discuss this from both a quantitative and a qualitative perspective.

From a more quantitative perspective, Arsenal had 3,884 phases in the season containing three or more events and 480 shots occurred in these phases. Leicester City had 4,099 phases in the season containing three or more events and 439 shots occurred in these phases. Manchester City had 3,828 phases in the season containing three or more events and 512 shots occurred in these phases. The number of phases for Arsenal and Manchester City are very similar whereas Leicester has slightly more. One possible explanation could be that Leicester City matches typically involved a lot of duelling in midfield, which can lead to more possession changes. Additionally, Leicester generated around 10% fewer shots than Arsenal, and Arsenal generated about 6.5% fewer shots than Manchester City.

Table 6.6 gives the number of phases (P), number of shots (S), and the number

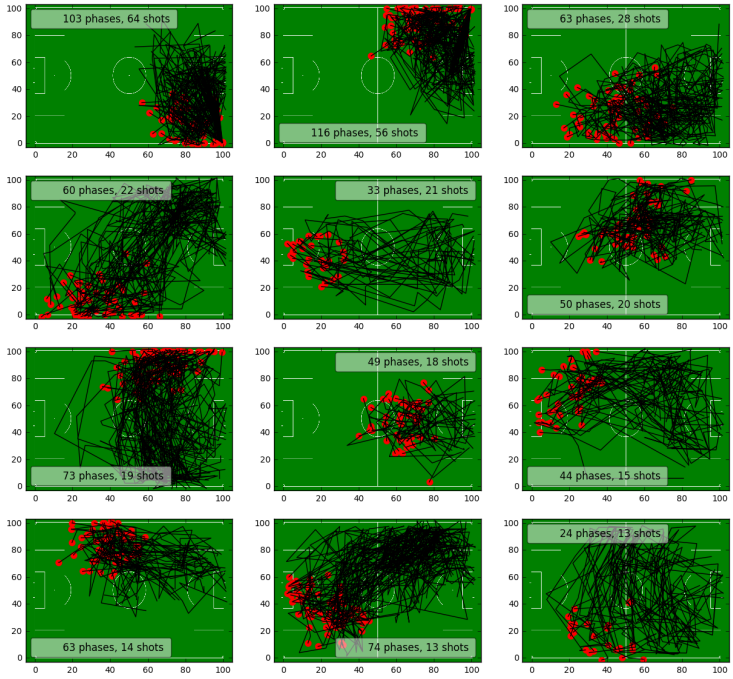


Figure 6.4: All phases assigned to each of the top 12 clusters for Manchester City. The red dots indicate where the phase begins. Manchester City is attacking to the right. The top-ranked cluster is in the upper left hand corner, and the clusters are ordered from left to right.

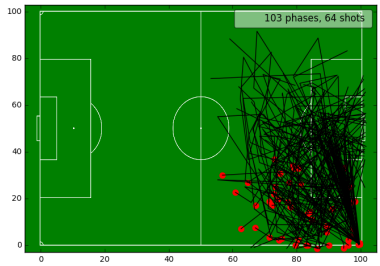


Figure 6.5: All phases assigned to the top-ranked cluster for Manchester City. The red dots indicate where the phase begins. Manchester City is attacking to the right. This shows a clear attacking pattern starting from the right flank.

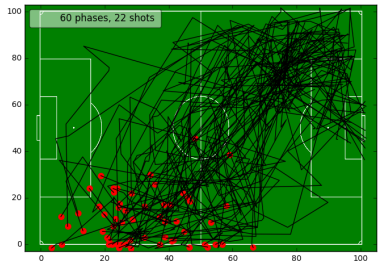


Figure 6.6: All phases assigned to the fourth-ranked cluster for Manchester City. The red dots indicate where the phase begins. Manchester City is attacking to the right.

of frequent sequential patterns (FS) contained within each of the top 10 ranked clusters for each team. 59.2% of Leicester City's shots occur in the top 10 clusters, with 16.5% appearing in the top cluster. In contrast, 54.1% of Manchester City's shots occur in the top 10 clusters with 12.5% in the top cluster. For Arsenal, only 50.4% of the shots occur in the top 10 clusters and 9.4% in the top cluster.

For Arsenal, Figure 6.7 shows the phases assigned to each of the top 12 clusters. Arsenal's play exhibits spatial diversity in how attacks are generated. Like most teams, the top two clusters capture attacks from either flanks (e.g., from corners or crosses). However, the clusters ranked 3 through 5 all capture various phases that start near Arsenal's own goal line, with a large number originating from

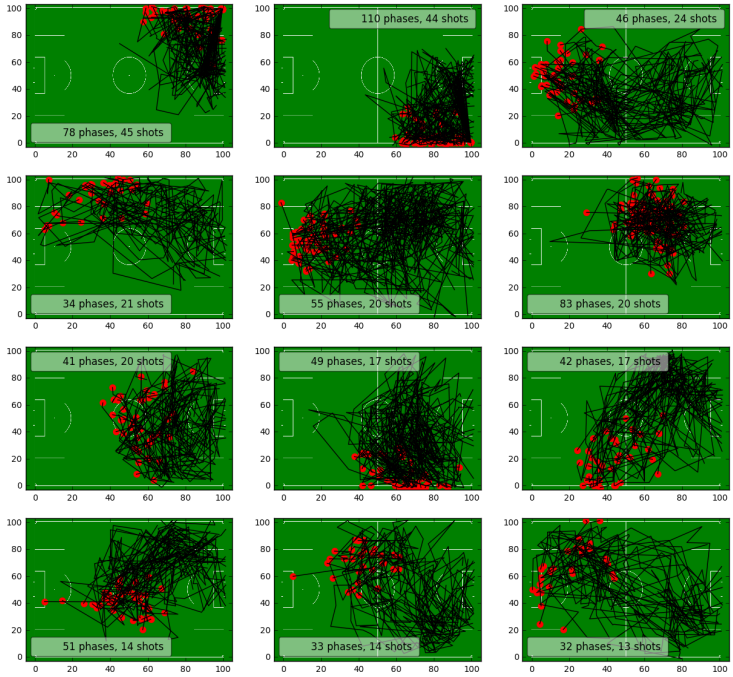


Figure 6.7: All phases assigned to each of the top 12 clusters for Arsenal. The red dots indicate where the phase begins. Arsenal is attacking to the right.The top-ranked cluster is in the upper left hand corner, and the clusters are ordered from left to right.

goal kicks. These account for 65 shots. Their play involves long sequences of passing the ball around, with lots of action through the midfield. Table 6.3 shows the top-ranked frequent sequences within three clusters for Arsenal.

For Leicester City, Figure 6.8 shows the phases assigned to each of the top 12 clusters. Leicester City's top two clusters capture attacks from the right or left flank. However, Leicester generates many more shots (71) from the left flank than the right flank (46) whereas other teams are more balanced. The fact that Leicester was more prolific from the left is a bit surprising, as Riyad Mahrez, who won one of the player of the year awards and had a large number of goals

Table 6.3: The top-ranked frequent sequences found in the second, third, and ninth ranked clusters for Arsenal.

Cluster	Sequential Pattern
2^{nd} Cluster	1. A pass OR cross FROM the left flank TO the box 2. A shot
3^{rd} Cluster	1. A pass FROM the midfield TO the midfield 2. A pass FROM the midfield TO the midfield 3. A pass FROM the midfield TO the midfield
9^{th} Cluster	1. A pass FROM the midfield TO the midfield 2. A pass FROM the midfield TO the left-flank 3. A pass FROM the left flank TO the midfield 4. A pass FROM the midfield TO the midfield 5. A pass FROM the midfield TO the midfield

Table 6.4: The top-ranked frequent sequences found in the first, second, and seventh ranked clusters for Leicester City.

Cluster	Sequential Pattern
1^{st} Cluster	1. A pass OR cross FROM the left flank TO the box 2. A shot
2^{nd} Cluster	1. A pass OR cross FROM the right flank TO the box 2. A shot AND a Miss 3. Ball goes out of bounds
7^{th} Cluster	1. A ball recovery IN the midfield 2. A shot

(14 from open play) and assists (11), operated on the right. Unlike Arsenal, Leicester City has very few sequences in the top 10 clusters that start with a goal kick. Additionally, Leicester has four clusters where most phases start in the opponent's half of the midfield, and these generate 64 shots. This indicates a direct, counter-attacking style with shorter sequences. Table 6.3 shows the top-ranked frequent sequences within three clusters for Leicester City.

Manchester City generates a lot of shots from phases that start on the left or right flank, and the distribution is nearly even with 64 coming from the right and 56 from the left. More generally, Manchester City's style falls somewhere in between Arsenal and Leicester City. On the one hand, there are several clusters with phases starting near midfield that are short and direct. On the

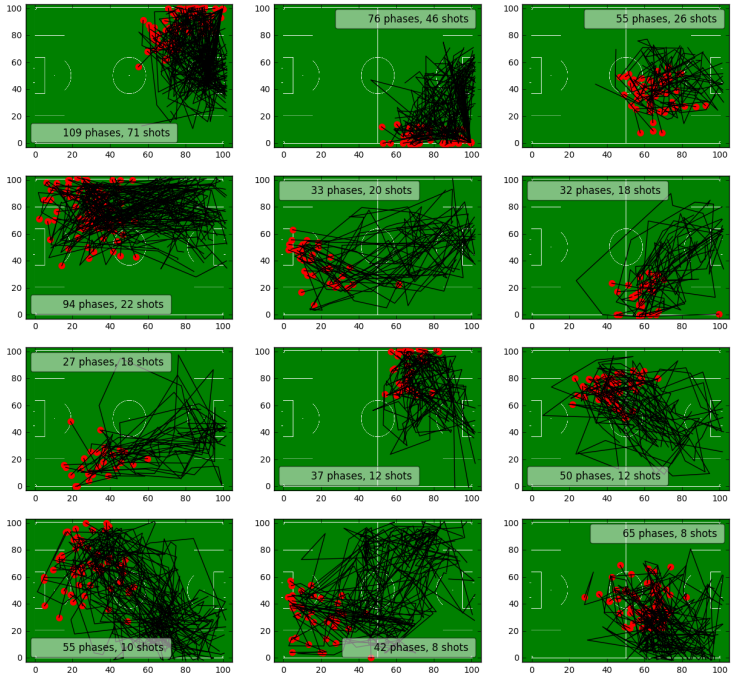


Figure 6.8: All phases assigned to each of the top 12 clusters for Leicester City. The red dots indicate where the phase begins. Leicester City is attacking to the right. The top-ranked cluster is in the upper left hand corner, and the clusters are ordered from left to right.

other hand, like Arsenal, there are some clusters that show groups of phases starting in Manchester City's half of the field between the penalty box and midfield. However, there are fewer phases initiated with a goal kick. Table 6.5 shows the top-ranked frequent sequences within three clusters for Manchester City.

Table 6.5: The top-ranked frequent sequences found in the first, third, and seventh ranked clusters for Manchester City.

Cluster	Sequential Pattern
1^{st} Cluster	1. A pass OR cross OR corner FROM the right flank TO the box 2. A shot
3^{rd} Cluster	1. A pass FROM the left flank TO the left flank 2. A pass FROM the left flank TO the midfield 3. A pass FROM the midfield TO the right flank
7^{th} Cluster	1. A ball recovery IN the left flank 2. A pass FROM the left flank TO the midfield

Table 6.6: A comparison of the clusterings found for Arsenal, Leicester City and Manchester City. The clusters are sorted by the number of shots each one contains and focuses on the 10 clusters that contain the most shots. Within each cluster, the number of phases (P), number of shots (S), and the number of frequent sequential patterns (FS) are shown.

Cluster	Arsenal			Leicester City			Manchester City		
Number	P	S	FS	P	S	FS	P	S	FS
1	78	45	143	109	71	227	103	64	141
2	110	44	159	76	46	134	116	56	127
3	46	24	47	55	26	11	63	28	82
4	34	21	20	94	22	34	60	22	72
5	55	20	187	33	20	17	33	21	19
6	83	20	26	32	18	7	50	20	27
7	41	20	40	27	18	13	73	19	165
8	49	17	150	37	12	12	49	18	10
9	42	17	116	50	12	16	44	15	12
10	51	14	40	55	10	41	63	14	18

6.3.4 The Effect of the Clustering Step

To answer **Q3**, we compare 1 cluster (i.e., no clustering), versus 10, 100, and 500 clusters. We focus the analysis on Manchester City. Table 6.7 provides statistics on the number of phases (P), number of shots (S), and the number of frequent sequential patterns (FS) contained within each of the top 10 ranked clusters when considering 10, 100, and 500 clusters. From a quantitative standpoint, considering only 10 clusters resulted in 225,118 frequent sequences which is substantially more than for 100 clusters (4,557) or 500 clusters (676). When looking at 100 clusters, 54.1% of all shots appear in the top 10 clusters, and for 500 clusters this number drops to 30.5%.

When performing no clustering, essentially all found patterns involve passing patterns of differing length within the midfield, with an occasional pass to one of the flanks. There are no patterns involving shots or the box in the top 100 ranked frequent sequences.

When clustering into 10 clusters, seven of the clusters generate patterns that contain almost only passes within midfield. There is one cluster for attacks from the right flank and one from the left. Finally, one cluster contains phases starting near Manchester City's own box. Hence there is little diversity in the found patterns.

Figure 6.9 shows the top 12 ranked clusters when the phases are clustered into 500 clusters. The clusters are typically very spatially coherent, but contain very few phases. This makes it difficult to find interesting patterns that have enough support in the data. Consequently, looking at 100 clusters seems to be a good tradeoff between diversity, spatial coherence, and sufficient data to find patterns with the desired support.

6.3.5 The Best Phase Representation for Mining Patterns

Next, we consider the effect of the five different ways to convert an event into an itemset representation. We focus on the top 10 ranked patterns within each of the top 10 clusters. The two approaches (location and event type; player, location, and event type) that include multiple pieces of information result in very similar patterns. In both cases, around two thirds of the patterns are length one or two and one third are length three or greater. A drawback to including the player information is that it greatly expands the search space of possible patterns that need to be considered.

Only considering one aspect of an event in the itemset representation is quite limiting. Particularly, for just the location or just the event type, the patterns

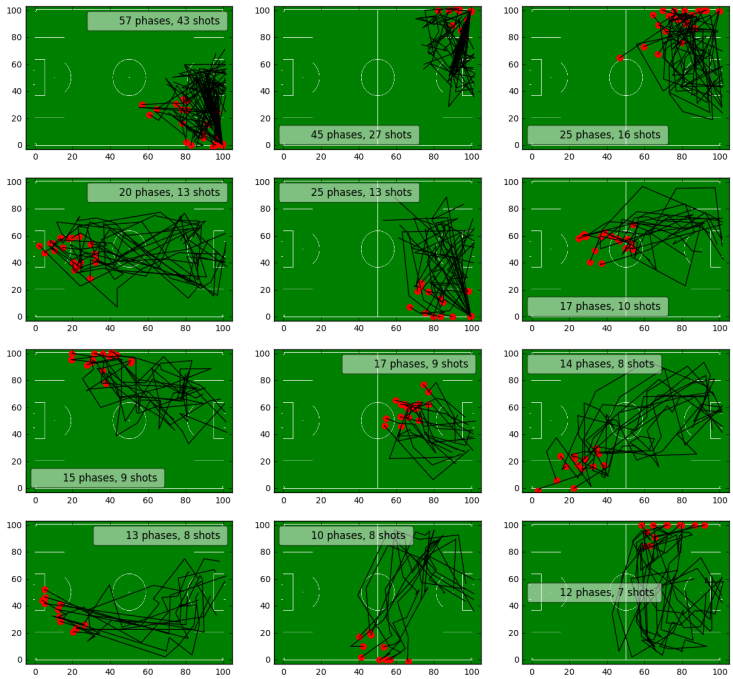


Figure 6.9: All phases assigned to each of the top 12 clusters from Manchester City when considering 500 clusters. The red dots indicate where the phase begins. Manchester City is attacking to the right.

Table 6.7: The effect of the number of clusters with three cluster sizes considered: 10, 100, and 500. The clusters are sorted by the number of shots each one contains and focuses on the 10 clusters that contain the most shots. Within each cluster, the number of phases (P), number of shots (S), and the number of frequent sequential patterns (FS) are shown.

Cluster	10 Clusters			100 Clusters			500 Clusters		
Number	P	S	FS	P	S	FS	P	S	FS
1	424	108	488	103	64	141	57	43	86
2	353	89	314	116	56	127	45	27	88
3	344	85	77,699	63	28	82	25	16	8
4	539	56	6,812	60	22	72	20	13	12
5	144	47	136,212	33	21	19	25	13	3
6	305	46	1,702	50	20	27	17	10	3
7	199	46	908	73	19	165	15	9	3
8	355	35	353	49	18	10	17	9	0
9	763	0	520	44	15	12	14	8	1
10	402	0	110	63	14	18	13	8	0

contain little context about what is happening. For just location or just event type, it is possible to find quite long sequences. For locations, it is common to find sequences of length five, with the longest being of length eight. For event type, sequences of up to length 13 are found. When considering just players, most patterns are very short, most are length one or two, and nothing longer than length three. A common finding is a pattern where the same player is involved in multiple consecutive events. These most likely indicate a dribbling sequence followed by a pass. But again, more context is needed to obtain a good understanding.

6.4 Related Work

This chapter falls within the emerging area of work that looks at analyzing spatio-temporal sports data. Knauf et al. [96, 97] proposed a novel spatio-temporal kernel for clustering player trajectory data. Their kernel is able to consider multiple different trajectories simultaneously, which is important for capturing tactics. Furthermore, it is based on the solid theoretical foundations of kernels. In several context-specific scenarios, such as play initiation and attacks, the approach identified interesting clusters that are illustrative of differences between two team's playing styles. Another trajectory-based approach, which focuses

on scoring opportunities [62], clusters together different scoring opportunities based on hand-crafted features as well as trajectory information about players on both teams. The goal is to assess how effective a team is at creating chances from certain types of situations (e.g., corners). Another approach to characterizing scoring chances is based on inductive logic programming [174], which allows representing rich, relational structure in a domain. This work focuses on discriminative mining of event streams to find patterns of play that are more likely than not to lead to shots on goal, but primarily focuses on capturing (hierarchical) spatial relations. Van Haaren et al. [173] used a similar approach in volleyball to detect patterns of play that occur frequently in won rallies.

Another way to analyze tactics is to build occupancy maps based on ball movements [109]. The occupancy maps are then used to assess how predictable the ball's movement is within a given region of the field. While the approach does provide a characterization of team behavior, it does not yield insights into specific trajectories or patterns of movement that a team employs to generate attacks. Another line of work looks at trying to characterize playing style and tactics by looking at passing patterns. One approach is to look at the passing graph and look for common passing sequences, that is, sequences of a given length that involve mostly the same players passing in mostly the same order [83]. However, it ignores the spatial component. Another approach looks at identifying frequent passing patterns by applying dynamic time warping [81]. Beyond these, other analyses of tactics include recognizing team formations in soccer (e.g., [15]) or identifying specific plays in American football (e.g., [167]).

In terms of data mining tasks, related areas include trajectory mining [73] and finding frequent spatio-temporal patterns [29]. In contrast to trajectory mining, the transition time between different events is not as important in our case. Both of these approaches also take the typical pattern mining approach of focusing on identifying frequent patterns. For example, a team's defenders may pass the ball among themselves, simply because they are trying to kill time. While this would result in a frequent pattern that represents a tactic ("kill time"), it does not provide significant insight into the more important strategic decisions such as how does a team build up the attack. Furthermore, frequency also ignores the fact that certain sequences are inherently more interesting. In soccer, events like goals, shots, and getting the ball in a dangerous area are very infrequent, but incredibly important. Hence, it is natural to assign more or less weight to a sequence based on how interesting and valuable the individual events within the pattern are. Importantly, spatio-temporal patterns in sports are not the result of a single object moving around. Instead, they arise from a complex, dynamic environment where many factors such as interactions among multiple different players across space and time, and features of the game state

(e.g., score, field position, time left, and team quality) influence decision making and tactics. Our approach is able to account for some of these factors.

6.5 Conclusions

Advanced data collection techniques are becoming more and more commonplace in sports and they generate rich, complex spatio-temporal data. Automatically analyzing team tactics from these data is an interesting and challenging problem. This chapter tackled one aspect of this task by trying to automatically discover interesting attacking tactics from event data collected from professional soccer matches. We proposed a five-step pipeline to analyze such data. An analysis of the 2015/16 English Premier League season identified several differences in style of play between different teams. It also identified some relevant, reoccurring patterns of play.

There are several important directions for future work. One is to continue to tackle the representational issues associated with performing pattern mining in a mixed discrete and continuous space. In conjunction with this, the ability to generalize to nearly identical commonly occurring sequences could also allow finding additional interesting patterns. Another direction is evaluating whether our results are consistent over time, i.e., if the tactics inferred for a given team after a set of matches, carry over to a subsequent set of matches. Finally, it would be interesting and much more informative to have full optical-tracking data for all players and the ball. However, tackling such a setting would require radically different techniques.

Chapter 7

Player Vectors: Characterizing Playing Style of Soccer Players

In this chapter, we attempt to characterize a player's playing style in an objective and data-driven manner based on analyzing event stream or play-by-play match data. While playing style is a somewhat subjective concept, our working definition is that playing style manifests itself as where on the pitch a player tends to perform specific actions with the ball. Our goal is to summarize this playing style in a fixed-length player vector. Characterizing playing style from event stream data is challenging as we have to reason about spatial locations, discrete actions, and a variable number of events. We cope with these challenges by overlaying a grid on the pitch and counting how often each player performs a specific action in a given location. Then, to reduce the dimensionality we perform non-negative matrix factorization. We repeat this for several types of actions. Finally, we concatenate together a player's compressed vectors for each action type to construct his player vector. To evaluate the quality of our player vectors, we propose a retrieval task. Given anonymous event data for a player, we show that we can accurately predict the player's true identify. Moreover, we show how to interpret player vectors and present several qualitative use-cases related to scouting and monitoring player development.

The content of this chapter is based on the following publication [44]:

Decroos, T., and Davis, J. Player Vectors: Characterizing Soccer Players' Playing Style from Match Event Streams. In *Joint European Conference on*

Machine Learning and Knowledge Discovery in Databases (2019), Springer

7.1 Applications of Playing Style at Soccer Clubs

Characterizing playing style can contribute to the following three processes at soccer clubs.

Scouting Soccer clubs can search the market more intelligently if they know the type of player they are looking for and how well prospective targets match that type. Transfers are expensive, and clubs are always looking for bargains and ways to mitigate risks in player recruitment.

Monitoring player development The coach can inspect the playing style of a player in a human-interpretable player vector. If the player vector matches the expectations of the coach, then the coach can monitor that this player vector remains stable and unchanged. If the player vector does not match the expectations of the coach, then he can give his player some pointers and afterwards monitor how well the player is implementing the advice.

Match preparation Understanding the playing style of your opponent can offer certain tactical advantages. The defenders of a team will wish to know what type of attackers they are up against. Similarly, the attackers will be interested in the playing style of the defenders they need to score against.

7.2 How to Define and Evaluate Playing Style

Characterizing a soccer player's playing style requires reaching a consensus on what constitutes a playing style. While this is an inherently subjective concept, our hypothesis is that a player's playing style arises from the interplay between his skills and the tactics employed by the team. Hence, a style of play will manifest itself in the player's behavior during the game.

Definition 2 (Playing style)**.** *A player's playing style can be characterized by his preferred area(s) on the field to occupy and which actions he tends to perform in each of these locations.*

In our work, we also make the following two assumptions.

Assumption 1: Most players exhibit differences in playing style and can be differentiated on this. While it is possible that two players exhibit such a similar playing style that they cannot be discerned from each other, this is not the case for most pairs of players.

Assumption 2: A player's playing style will not drastically change in a short period of time. That is, in a sequence of consecutive games in a season, each player will exhibit the same playing style. This seems justifiable for two reasons. First, while players' skills and playing style evolve over the course of their career, these changes occur gradually rather than abruptly. Second, while the tenure of managers, who influence tactics, do not tend to be overly long in professional soccer, the majority of teams in a league do not change manager mid-season.

Based on this definition and these assumptions, any system that successfully characterizes playing style from match event stream data can be used to retrieve players from anonymized event stream data. This player retrieval task can be more formally defined as follows:

Given: Anonymized event stream data describing actions of player p

Retrieve: The identity of player p

The quality of a system that characterizes playing style can be evaluated by its performance on this player retrieval task, as this task measures how well a system can recognize players and differentiate between them purely based on their actions on the field.

In the next section, we describe our system for solving this task. In addition to characterizing each player's playing style, our system also allows human analysts to interpret our representation of playing style and to automatically compare players' playing style on their similarity.

7.3 Building Player Vectors

In this section we address the following task:

Given: Event stream data describing actions of player p.

Build: A fixed-size player vector that characterizes player p's playing style and can be interpreted both by human analysts and machine learning systems.

At a high-level, our approach works as follows. First, we select relevant action types for characterizing playing style.

Second, for each player p and relevant action type t, we overlay a grid on the field and count how many times player p performed action t in each grid cell. This transform helps address some of the challenges listed in Chapter 3 because it (1) captures the spatial component, (2) fuses discrete (action type) with continuous (location) features in a unified representation, and (3) converts a variable length set of actions into a fixed size. We end up with one matrix per player per action type.

Third, we reshape each matrix into a vector and group it together with all other vectors of the same action type to form new, bigger matrices per action type detailing the playing styles of all players for that specific action type. We then perform non-negative matrix factorization (NMF) to reduce the dimensionality of these matrices. NMF automatically clusters together similar grid cells into a coherent group, which is more informative and intuitive (e.g., for scouting) than looking at individual grid cells where a player operates.

Finally, we construct a player vector for each player by concatenating his compressed vectors of each action type. We also show how to compute the similarity of player vectors (to be used in machine learning algorithms such as clustering and nearest neighbors).

7.3.1 Selecting Relevant Action Types

Our hypothesis is that the type and location of the actions a player performs are informative of that player's playing style. Our event stream data contains 19 different types of actions. However, we only consider offensive actions that are performed during open play for two reasons.

First, defense is primarily about positioning, and often this involves picking a position to prevent certain actions from occuring. Hence, by definition, characterizing defensive style requires off-the-ball location data, which we do not have access to. Furthermore, most on-the-ball defensive actions (e.g. tackles, clearances) are usually performed out of necessity rather than because they are indicative of a player's playing style. One effect of this criterion is that all keeper-specific actions are automatically ignored.

Second, the open play criteria means we exclude set piece actions (e.g., free-kicks and throw-ins) from our analysis. Teams typically have set-piece specialists (e.g., for free kicks). Similarly, actions like throw-ins are often performed by a pre-defined position (e.g., fullbacks or wingers), so this is more an artefact of

Table 7.1: Each action type must fit two criteria to be considered relevant for characterizing playing style: it must be offensive and it must occur during open play. The relevant action type are passes, dribbles, crosses, and shots.

Action type	Frequency	Offensive	Open play
pass	53.1%	✓	✓
dribble	25.2%	✓	✓
clearance	3.8%		✓
throw_in	2.8%	✓	
interception	2.6%		✓
tackle	2.3%		✓
cross	1.8%	✓	✓
shot	1.5%	✓	✓
bad_touch	1.4%		✓
foul	1.3%		✓
freekick_short	1.3%	✓	
keeper_pick_up	0.8%		✓
keeper_save	0.8%		✓
corner_crossed	0.6%	✓	
freekick_crossed	0.2%	✓	
keeper_claim	0.2%		✓
corner_short	0.1%	✓	
shot_freekick	0.1%	✓	
keeper_punch	0.1%		✓

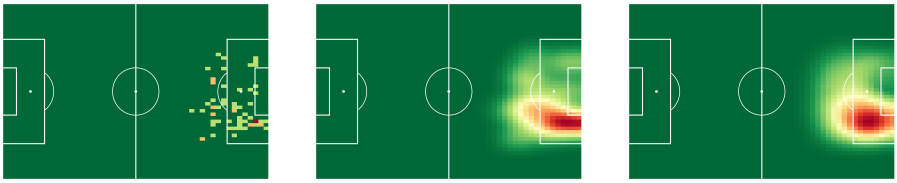
position than style. While we believe analyzing set pieces is extremely interesting and important, a proper study of these actions would require a different type of analysis than we perform in this chapter.

When applying these two criteria, the remaining relevant action types are passes, dribbles, crosses, and shots (Table 7.1).

7.3.2 Constructing Heatmaps

A heatmap is a summary of the locations where player p performs actions of type t. For each player and action type, we execute the following three steps.

1) Counting We overlay a grid of size $m \times n$ on the soccer field. Next, we select all of player p's actions of type t in our data set. Per grid cell X_{ij}, we count the number of actions that started in that cell. Hence, we have



(a) The raw shot heatmap obtained by overlaying a grid and counting shot locations

(b) The smoothed shot heatmap obtained after normalizing and smoothing the raw shot heatmap.

(c) The same shot heatmap reconstructed from a length-4 feature vector.

Figure 7.1: Example of a heatmap detailing the shot playing style of Riyad Mahrez, winger at Leicester City in the 2016/2017 season.

transformed a variable-size set of actions to a fixed-size matrix $X \in \mathbb{N}^{m \times n}$ containing the raw counts per cell.

2) Normalizing Two players p_1 and p_2 can have an identical playing style, but if player p_1 played more minutes than player p_2, then player p_1's matrix X will contain higher raw counts than the matrix of player p_2. To combat this, we normalize X such that each cell contains its count if p had played 90 minutes (1 game). For example, if player p played 1600 minutes in total in our data set, then we construct the normalized matrix $X' = \frac{90}{1600} X$.

3) Smoothing We would expect some spatial coherence, or smoothness, in the locations where the actions were performed. However, this coherence can be disrupted by laying a high granularity grid (i.e., high values for parameters m and n) over the pitch as the boundaries between grid cells are abrupt and somewhat arbitrary. Hence, the counts for nearby cells may exhibit more variance than they should. To promote smoothness in the counts of nearby cells, a Gaussian blur is applied to matrix X'. A Gaussian blur is a standard image processing technique [155] that involves convolving X' with a Gaussian function. Specifically, the value of each cell in X' is replaced by a weighted average of itself and its neighborhood, leading to the blurred matrix $X'' \in \mathbb{R}_+^{m \times n}$.

X'' is the heatmap detailing where player p performs actions of type t (Figure 7.1). For some action types, e.g., passes, we are not just interested in their start locations, but also in their end locations. For these action types, we construct separate heatmaps X''_{start} and X''_{end} using respectively the start and end locations of the actions in the counting step.

7.3.3 Compressing Heatmaps to Vectors

The goal is to capture the information available in a heatmap (i.e., the locations where a player p performs actions of type t) in a small vector. We detail our approach for compressing heatmaps to vectors per action type t.

First, we reshape each heatmap X'' to a 1-dimensional vector x of length mn. In the case of action types where we are interested in both the start and end location, we reshape the heatmaps X''_{start} and X''_{end} to vectors x'_{start} and x'_{end} and concatenate them in a single 1-dimensional vector x of length $2mn$. More generally, let $s = 1$ if we are only interested in the start location of an action type and $s = 2$ if we are interested in both the start and end location of an action type. The length of x is then smn.

We then construct the matrix $V = [x_0 x_1 \dots x_l]$ that contains as columns the reshaped heatmaps of all l players in our data set for action type t. Next, we compress matrix V by applying non-negative matrix factorization (NMF) [102], which is a form of principal component analysis where the resulting components contain only positive numbers. This results in two matrices W and H such that:

$$V \approx WH, \tag{7.1}$$

where $V \in \mathbb{R}_+^{smn \times l}$, $W \in \mathbb{R}_+^{smn \times k}$ and $H \in \mathbb{R}_+^{k \times l}$. Here, k is a user-defined parameter that refers to the number of principal components for action type t.

The columns of W are the principal components that represent basic spatial groups of action type t. These principal components can be visualized as heatmaps (Figure 7.2). The rows of H are the small vectors that are the compressed versions of the heatmaps in V. In other words, if the reshaped heatmap x was the i-th column in matrix V, then the i-th row of H is its compressed vector. Each compressed vector can be visualized by multiplying it with the principal component matrix W. The result of this multiplication is a heatmap similar to the original, but reconstructed from only k features (Figure 7.1). In addition, each feature in a compressed vector is interpretable in the sense that its numeric value quantifies how often the player executes actions of type t with locations in the spatial group of a specific principal component.

7.3.4 Assembling Player Vectors

The player vector v of a player p is the concatenation of his compressed vectors for the relevant action types: passes, dribbles, crosses, and shots. The total length of a player vector v is equal to $k_{pass} + k_{dribble} + k_{cross} + k_{shot}$ where k_t is the number of principal components chosen to compress heatmaps of action

type t. In this chapter, we set k_t as the minimal number of components needed to explain 70% of the variance in the heatmaps of action type t. This parameter setting was empirically found to work well because of the high variability of players' positions in their actions (Challenge 2 in Section 3.4). Ignoring 30% of the variance allows us to summarize a player's playing style only by his dominant regions on the field rather than model every position on the field he ever occupied. This design choice lead us to use 4 shot components, 4 cross components, 5 dribble components, and 5 pass components, adding up to form length-18 player vectors.

We can now quantify two player's playing style similarity by computing the Manhattan distance between their player vectors. Manhattan distance works well because the value of each feature in each player vector is a meaningful quantity. The Manhattan distance does not alter this meaning and simply computes the sum of the absolute differences per feature, unlike Euclidean distance which tends to unfairly penalize large differences in a few features. We also empirically confirm that the Manhattan distance works best in Section 7.4.4.

7.4 Experiments

Evaluating our method is challenging as no objective ground truth exists for characterizing playing style. Therefore our experiments address three main questions: (1) providing intuitions into what information our player vectors capture, (2) demonstrating how our approach could be used for scouting and monitoring player development by substantiating a number of claims in popular media about professional soccer players, and (3) measuring our performance at the player retrieval task, which we argue in Section 7.2 is an effective proxy for how well our approach characterizes playing style.

Our data set consists of Opta event stream data from 9155 matches in the five major soccer competitions in Europe: the English Premier League, the German Bundesliga, the Spanish Primera Division, the Italian Serie A and the French Ligue Un. Our data spans almost all matches between the 2012/2013 and 2016/2017 seasons. Our match event stream data is encoded in the SPADL format (Chapter 3).

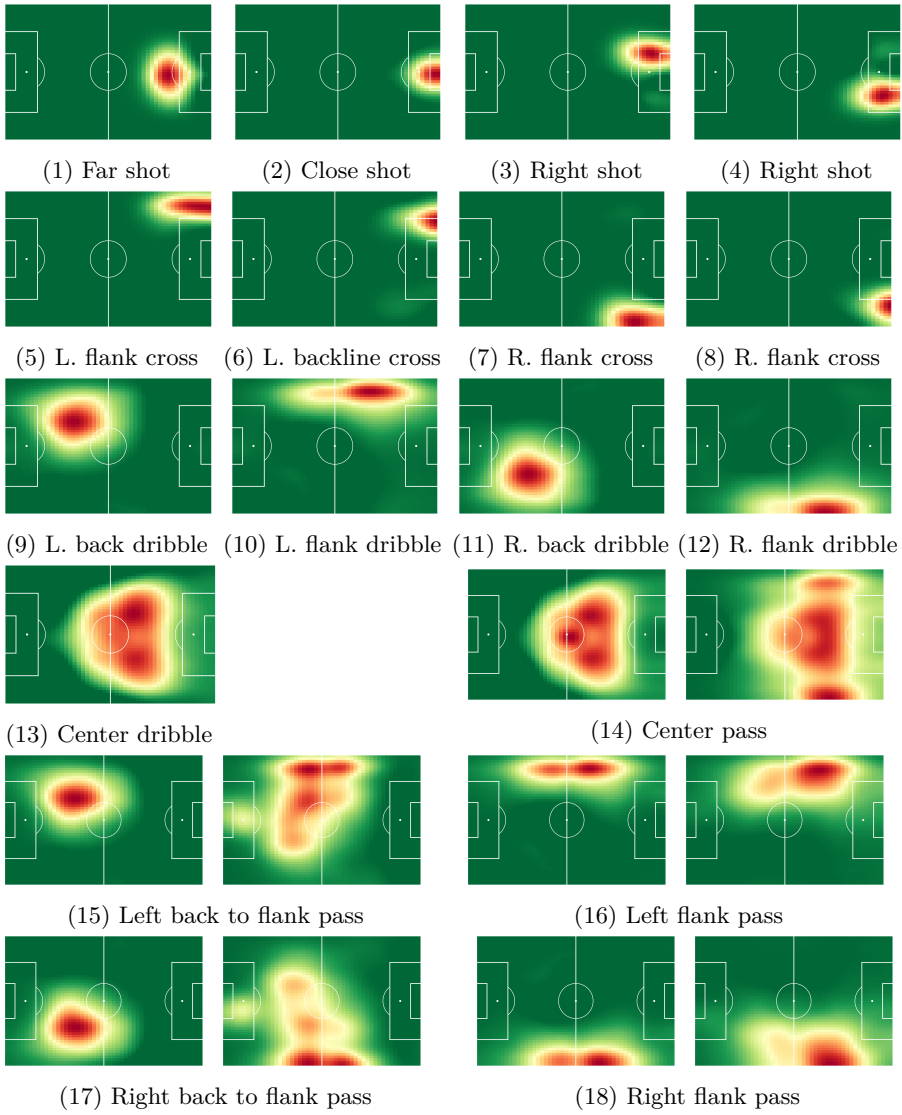


Figure 7.2: The 18 components of our player vectors constructed by compressing heatmaps of shots (1-4), crosses (5-8), dribbles (9-13), and passes (14-18) with non-negative matrix factorization.

7.4.1 Intuition

Figure 7.2 illustrates all 18 components (4 shots, 4 crosses, 5 dribbles, 5 passes) corresponding to the weights in our length-18 player vectors. The shot, cross, and dribble components only describe where groups of actions start, while the pass components descibe where groups of passes start and end. This is because the end locations of shots, crosses, and dribbles are not informative of a player's playing style. Shots and crosses all end in roughly the same location, while dribbles are usually short and vary in direction such that there is no noticeable difference between their start and end heatmaps.

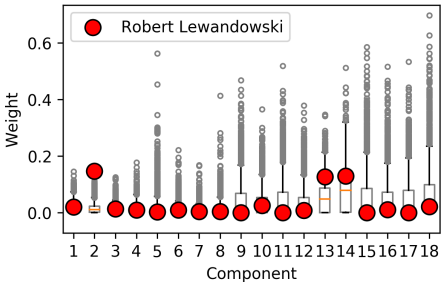
Figure 7.3 shows the player vectors of four archetypical players in their 2016/2017 season.

Robert Lewandowski: Striker at Bayern Munich. He shows high weights for three components: *C2: Close shot*, *C13: Center dribble*, and *C14: Center pass*. These are the actions central strikers are expected to focus on.

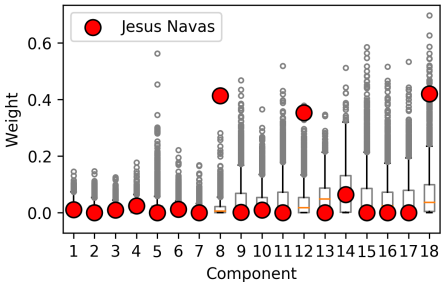
Jesus Navas: Winger at Manchester City. He shows high weights for three components that are typical of an offensive right winger: *C8: Right backline cross*, *C12: Right flank dribble*, *C18: Right flank pass*.

Kevin De Bruyne Midfielder at Manchester City. De Bruyne's player vector seems less pronounced than the others at first glance, but is actually very informative. First, we can deduce that De Bruyne plays mostly on the opponent's half due to the non-existent weights for components *C9/C11: Left/Right back dribble* and *C15/C17: Left/Right back to flank pass*. Second, his player vector shows similar values for (almost) all mirroring components (e.g., *C16/C18: Left/Right flank pass*). The exception is shots: his weight for *C4: Right shot* is high, while almost non-existent for *C3: Left shot*. De Bruyne's player vector suggests that he is an offensive central midfielder with no preference towards the left or the right when it comes to passing, dribbling or crossing, but attempts to score only from the right.

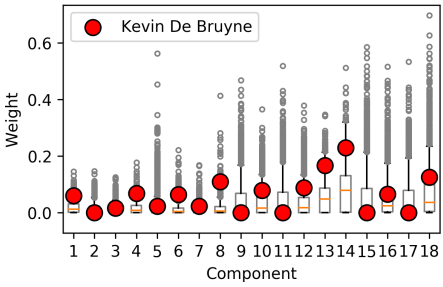
Sergio Ramos Defender at Real Madrid. Two of his components stand out: *C9: Left back dribble* and *C15: Left back to flank pass*. While less notable than his defensive components, Ramos shows an unusually high weight for *C2: Close shot*. This is because Ramos often attempts to head the ball in the goal at corner kicks, as proven by his nine goals in the 2016/2017 season headed in from corner kicks. Ramos is a left-most central defender with a very defensive playing style, except when it comes to corners.



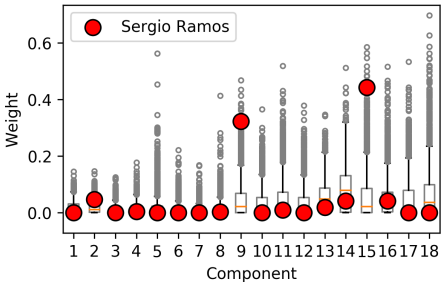
(a) Robert Lewandowski, central striker at Bayern Munich, shows high weights for *C2: Close shot*, *C13: Center dribble*, and *C14: Center pass*.



(b) Jesus Navas, right winger at Manchester City, shows high weights for *C8: Right backline cross*, *C12: Right flank dribble*, and *C18: Right flank pass*.



(c) Kevin De Bruyne, central offensive midfielder at Manchester City, shows high weights for all offensive components, favoring neither left nor right.



(d) Sergio Ramos, left-most central defender at Real Madrid, shows high weights for *C9: Left back dribble* and *C15: Left back to flank pass*.

Figure 7.3: Visualized player vectors for an archetypical (a) striker, (b) winger, (c) midfielder, and (d) defender in the 2016/2017 season. The boxplots in the background show the distribution of the weights per component.

Player vectors can characterize playing style in an intuitive manner that can make sense to domain experts (e.g., scouts and coaches), yet the interpretable components upon which the player vectors are built are constructed in a completely data-driven manner.

7.4.2 Scouting

We investigate three claims in popular media about similar players. We computed and compared player vectors for all 1480 players who played at least 900 minutes in the 2016/2017 season of the five major soccer competitions

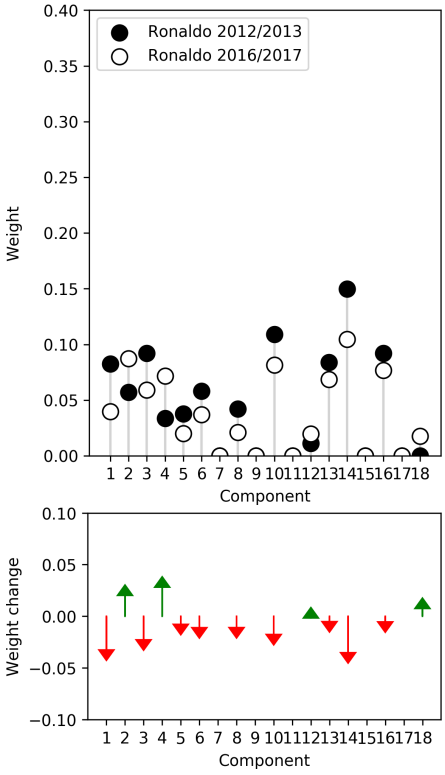
in Europe. Lionel Messi is regarded by many as the best soccer player in the world. One player who has been deemed to play similarly to Messi is Paulo Dybala, a fellow Argentinian attacker [160, 74]. When ranked using our player vectors, Dybala is the 2^{nd} most similar player (out of 1479) to Lionel Messi. Idrissa Gueye (midfielder at Everton FC) is often hailed as the new N'golo Kante (midfielder at Chelsea FC) by many journalists [1, 95, 28]. Gueye is the 2^{nd} most similar player to Kante in our data set. Aymeric Laporte is a 24-year-old defender playing for Manchester City FC, who was deemed to be the long-term replacement for 33-year-old Real Madrid defender Sergio Ramos [138, 38], who was named best defender in the world in 2017 by UEFA.[1] Laporte is the 29^{th} most similar player to Ramos using our player vectors. While 29^{th} out of 1479 is not bad, this example does illustrate that our approach is better at characterizing offensive playing style than defensive playing style, as defensive playing style is often more about positioning than on-the-ball actions (Challenge 6 in Section 3.4).

7.4.3 Monitoring Player Development

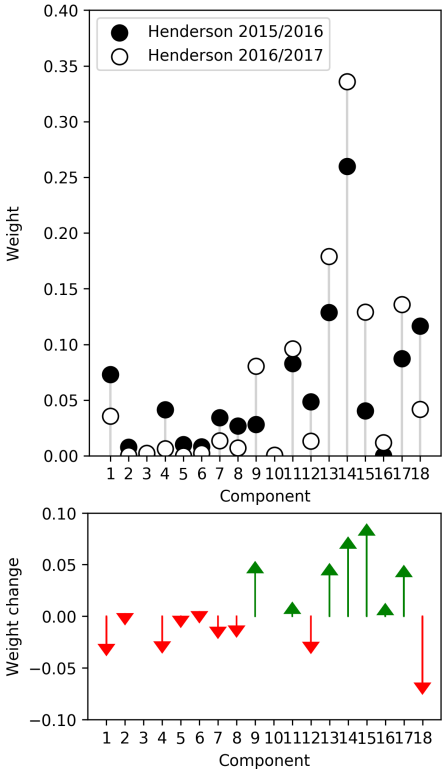
Journalists agree that Cristiano Ronaldo (ex-Real Madrid) evolved from his role as a left winger to a role as a central striker [156, 145]. Our player vectors capture this transition (Figure 7.4). In the 2012/2013 season, Ronaldo's most common shot types were *C1: Far shot* and *C3: Left shot*. In the 2016/2017 season however, his shot playing style is completely different with *C2: Close shot* as his most common shot type and no significant difference in output between *C3: Left shot* and *C4: Right shot*. Ronaldo also executed fewer crosses, dribbles and passes in the 2016/2017 season (see the drops in components 5-18), focusing more on finishing scoring chances than setting them up.

Jordan Henderson is a midfielder at Liverpool. In the 2016/2017 season, coach Jürgen Klopp instructed Henderson to play more defensively, transitioning his playing style from a box-to-box midfielder to a defensive midfielder [184, 134]. When comparing Henderson's 2015/2016 player vector to his 2016/2017 player vector (Figure 7.4), we notice that his output in terms of passes and dribbles (components 9-18) has significantly increased, while his output in terms of shots and crosses has completely disappeared (components 1-8).

[1] `http://www.uefa.com/insideuefa/awards/previous-winners/newsid=2495000.html`



(a) Ronaldo evolved from a left winger in the 2012/2013 season to a central striker in the 2016/2017 season. Note the drop of *C1: Far shot* and *C3: Left shot* and the rise of *C2: Close shot* and *C4: Right shot*.



(b) Henderson transitioned to a more defensive playing style after the 2015/2016 season. Note the almost complete disappearance of shots and crosses (components 1-8) and the rise of passes and dribbles (components 9-18).

Figure 7.4: Player vectors illustrating the development of (a) Cristiano Ronaldo, former striker at Real Madrid, and (b) Jordan Henderson, midfielder at Liverpool.

Table 7.2: Top-k results and mean reciprocal rank (MRR) when trying to retrieve 741 players from anonymized event stream data of season 2016/2017 using labeled event stream data from season 2015/2016.

Distance function	Top-1	Top-3	Top-5	Top-10	MRR
Manhattan distance	**38.2%**	**49.8%**	**54.9%**	**64.4%**	**0.469**
Euclidean distance	33.0%	47.0%	52.9%	61.8%	0.429

7.4.4 Player Retrieval from Anonymized Event Stream Data

Our approach has many parameters: (a) the size of the grid to construct the heatmaps (50×50), (b) the algorithm to smooth the heatmaps (Gaussian blur), (c) the algorithm to compress the heatmaps (non-negative matrix factorization), (d) the number of components to use (4 shots, 4 crosses, 5 dribbles, and 5 passes), and (e) the distance function to compare the player vectors (Manhattan). Normally we would have no experimental way to tune these parameters as playing style is a subjective concept with no ground truth. However, as explained in Section 7.2, we can use player retrieval from anonymized match event stream data as a proxy for characterizing playing style.

We solve the player retrieval task as follows. First, we construct a set of labeled player vectors V using a training event stream data set that has not been anonymized. Second, we obtain a set of anonymous actions performed by a target player p_t and construct a player vector v_t based on these actions. Third, we compare v_t to all $v \in V$ and construct a rank-ordered list of the most similar players to p_t. The quality of this ranking is then the position of the unknown player in the ranking. In other words, if most players appear at the top of their own rankings, then we have successfully characterized playing style. If most players do not appear near the top of their own rankings, then we have failed.

To illustrate this idea, we provide the results of an experiment to test whether Manhattan distance or Euclidean distance is the best distance function for comparing player vectors in Table 7.2. In our experiment, our training data was labeled event stream data from season 2015/2016 of the five top soccer competitions in Europe and the test data was anonymized event stream data from season 2016/2017 of the same competitions. We only considered players that have played 900 minutes in the same team in both seasons. This left us with 741 anonymized players in the test data which we de-anonymized using 741 labeled players in the training data.

The Manhattan distance outperforms the Euclidean distance at retrieving players from anonymized event stream data. We can successfully retrieve 38.2%

of all players with only one attempt and retrieve 64.4% of all players in the top-10 of our rankings. Hence, we conclude that Manhattan distance is the better distance function to use to compare players' playing style.

7.5 Related Work

Danneels et al. [40] predict a player's position (i.e., attacker, midfielder, defender) based on their actions. While similar to our research, our goal and approach is more broad and ambitious, as our player vectors are much more detailed than only three distinct labels. Gyarmati et al. [82] construct movement vectors to characterize a player by his movement on the field. Van Gool et al. [172] analyze the playing style of teams instead of players. Their approach is different from ours, but their goal is similar as they also try to capture a subjective concept like playing style in a more objective and data-driven way. STATS introduced *STATS Playing Styles* [64], which are eight different styles (e.g., fast tempo, direct play, counter attack) teams use to create shooting opportunities. Fernandez et al. [61] also categorize different styles of play for teams in professional soccer.

In other sports, Franks et al. [69] used spatial information to categorize shots in professional basketball. In this work, data from the NBA was collected and analyzed using non-negative matrix factorization (NMF). This paper was a huge influence on our work, as our approach on soccer event stream data is largely inspired by their approach on basketball event stream data.

7.6 Conclusion

Objectively characterizing the playing style of professional soccer players has important applications in scouting, player development monitoring, and match preparation. We showed how to construct player vectors by transforming sets of actions from match event stream data to fixed-size players vectors using non-negative matrix factorization. These player vectors offer a complete view of a player's playing style (within the limits of the data source), are constructed in a purely data-driven manner, are human-interpretable and can be used in machine learning systems such as clustering and nearest neighbor analysis.

Chapter 8

SoccerMix: Representing Soccer Actions with Mixture Models

In this chapter, we introduce SoccerMix, a novel mixture-model approach for analyzing on-the-ball soccer actions that addresses a number of shortcomings of grid-based approaches. On the one hand, it alleviates the problem of sparsity by grouping actions in a data-driven manner. On the other hand, SoccerMix's probabilistic nature alleviates the issues of the arbitrary and abrupt boundaries imposed by grid cells. More uniquely, SoccerMix also considers the direction that actions tend to move the ball in, which is an important property for capturing style of play that has received little attention thus far. For example, it allows distinguishing among players or teams that play probing forward passes versus those that play safer lateral passes in a specific zone of the pitch. Intuitively, the action groups produced by SoccerMix can be thought of as describing *prototypical* actions of a certain type, location, and direction.

We provide a number of use cases that illustrate how SoccerMix can aid in scouting and match analysis by capturing the playing styles of both teams and players. In contrast to existing approaches which solely focus on the offensive style of a team, SoccerMix can also yield insights into a team's defensive style. Specifically, we model how a team can force its opponent to deviate from its typical style of play.

A publicly available implementation of SoccerMix is available at `https://github.com/ML-KULeuven/soccermix`.

The content of this chapter is based on the following publication [52]:

Decroos, T., Van Roy, M., and Davis, J. SoccerMix: Representing Soccer Actions with Mixture Models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (2020), Springer

8.1 Shortcomings of Grid-based Approaches

Analyzing the playing style of a team or player based on event stream data often involves constructing a so-called *fingerprint* of that team or player which summarizes their actions and captures distinguishing behaviors such as where on the field they tend to perform certain actions. This is often done by dividing actions into groups of similar actions and counting how often players or teams perform actions within each group. However, assessing similarity is difficult because actions are described by various attributes (e.g., type, location) which lay in different domains (e.g., discrete, continuous).

One approach is to lay a grid over the field and proclaim two actions to be similar when they are of the same type and fall in the same grid cell [37, 44, 174, 175]. However, this approach has three downsides. First, the somewhat arbitrary and abrupt boundaries between grid cells can make certain spatially close actions appear dissimilar. Second, choosing the best resolution for the grid is non-trivial as a coarse grid ignores important differences between locations, while a more fine-grained grid will drastically increase the sparsity of the data as a smaller number of actions will fall in a single grid cell. Third, ideally we would like to group actions on additional attributes such as ball direction, but considering more attributes makes each action more unique, which increases the sparsity in the data. Hence, most approaches only include one or two attributes in their analysis [12, 44, 132]. Rarely do approaches consider three or more attributes [50, 175].

8.2 Methodology

Our goal is to capture the playing style of either a player or a team. As in past works [44, 83, 174, 175], our intuition is that playing style is tied to where on the pitch a player (or team) tends to carry out certain types of actions. Most playing style analysis techniques follow the same two-step approach:

Step 1: Partition all on-the-ball actions into groups of similar actions and represent each action by its membership to one or more of these groups.

Step 2: Transform the group membership counts of a player's or team's actions into a human-interpretable summary of playing style.

Traditionally, most research has focused on the second step [37, 44, 50, 174]. However, picking sub-optimal groups in the first step can introduce significant problems such as sparsity down the line. In fact, many sophisticated data aggregation methods such as pattern mining [50] and matrix factorizaton [44] are often only used in step two to combat the problems introduced by the sub-optimal groups established in the first step.

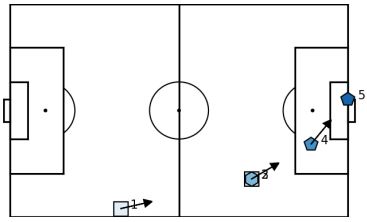
In this chapter, we attempt to tackle the first step in a more intelligent manner than before in order to greatly simplify the second step. More specifically, we aim to find groups of similar actions such that players' or teams' group membership counts are already human-interpretable and informative of playing style. This way, no additional sophisticated transformation is needed in step two. Finding these groups of similar actions involves answering four questions:

1. Which properties of actions are relevant for capturing playing style?
2. How can we group actions based on both discrete and continuous properties?
3. How can we prevent sparsity (many groups with little or no actions in them)?
4. How can we group actions based on properties with different notions of similarity (e.g., linear data vs. circular data)?

8.2.1 Describing Actions

Various companies provide event stream data and each one uses a different format, has varying definitions of events, and records different sets of events. Moreover, the data also contains extraneous information such as changes in weather that are not crucial for analysis. The SPADL representation [42] addresses these concerns by converting event streams to a uniform representation designed to facilitate analysis. Additionally, we wish to treat the initiation and completion of certain actions separately (e.g. giving and receiving a pass), as these are typically performed by different players. Hence, we began by transforming our data into the Atomic-SPADL format (Chapter 3).

Typically, playing style analysis focuses on action types and locations. One piece of data that is important for style of play and which has received little attention is the direction of actions. For example, it is important to differentiate



	Player	Type	Location	Direction
1	T. Alexander-Arnold	pass	(34.4m, 2.6m)	0.23 rad
2	M. Salah	receival	(75.0m, 12.1m)	0.00 rad
3	M. Salah	dribble	(75.0m, 12.1m)	0.54 rad
4	M. Salah	shot	(93.5m, 23.2m)	0.89 rad
5	M. Salah	goal	(105.0m, 37.5m)	0.00 rad

Figure 8.1: This phase of Liverpool scoring a goal illustrates the event stream data used in this chapter. Actions are described by their type t, location (x, y), and direction θ.

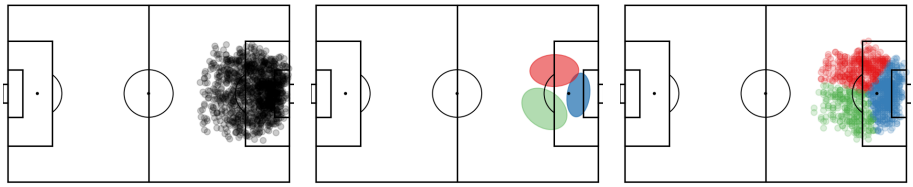
among players who tend to play probing forward passes versus those that tend to play safer, lateral passes. Therefore, in this chapter, we represent each action as a tuple (t, x, y, θ) where t is the type of the action (e.g., shot, tackle, pass, receival), $x \in [0, 105]$ (meters) and $y \in [0, 68]$ (meters) denote the location on the field where the action happened, and $\theta \in [-\pi, \pi]$ (radians) denotes the direction the ball travels in following the action (Figure 8.1).

8.2.2 Grouping Actions with Mixture Models

Grouping actions on multiple attributes is non-trivial as it requires fusing together both discrete attributes (i.e., the action type) and continuous attributes (i.e., the location and direction). Past work has mostly ignored direction and focused on fusing action type and location. The most common approach is to lay a grid over the field and for each action type count the number of times it occurs in each zone [44, 174, 175]. However, this approach has two significant problems. First, this approach ignores the fact that some actions only ever occur in certain areas of the pitch (e.g., throw-ins only occur on the outer edges of the field, shots typically only occur on the attacking half of the field). Second, the boundaries between grid cells are arbitrary and abrupt, which can disrupt the spatial coherence. This can make some actions that occurred in nearby locations appear dissimilar because they fall in different location groups.

This chapter takes a different approach and uses mixture models to cluster actions. Mixture models are probabilistic models that assume that all the data points are generated from a mixture of a finite number of distributions with unknown parameters [117]. Formally, a mixture model calculates the probability of generating observation x as:

$$p(x) = \sum_{j=1}^{k} \alpha_j \cdot F_j(x|\Theta_j) \tag{8.1}$$



(a) Shot locations

(b) The 2D Gaussian distributions learned on shot locations

(c) Shots colored according to the distributions that most likely generated them.

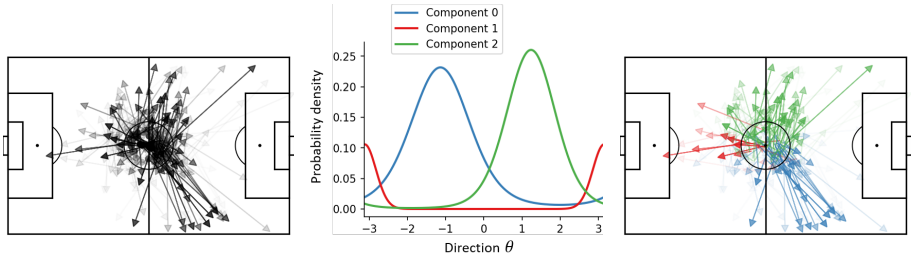
Figure 8.2: Stage 1 of SoccerMix: a mixture model with three 2D Gaussian distributions is fitted to shot locations.

where k is the number of components in the mixture model, α_j is the probability of the j^{th} component, and F_j is a probability distribution or density parameterized by Θ_j for the j^{th} component. Intuitively, mixture models can be thought of as a soft clustering variant of k-means clustering. Mixture models address all the drawbacks of the grid-based approach. First, they perform a more data-driven as opposed to a hand-crafted partitioning of the pitch. This results in a more nuanced partitioning as the mixture model can learn a more fine-grained representation in zones where lots of actions take place and a more course-grained one in zones where actions are less frequent. Second, by performing a soft grouping each action has a probability of belonging to each cluster, which alleviates the arbitrariness of grid boundaries.

SoccerMix hierarchically groups actions with mixture models in two stages:

Stage 1 For each action type, fit a mixture model to the locations (x, y) of the actions of that type. This allows SoccerMix to model that certain action types usually occur in specific areas of the field (e.g., shots only occur close to the goal, see Figure 8.2)

Stage 2 For each component of each mixture model in stage 1, fit a new mixture model to the directions θ of the actions in that component. This allows SoccerMix to model that the direction that a specific action tends to move the ball in, depends on the location where the action occurred (e.g., passes in central midfield are usually lateral or backwards, rarely forwards, see Figure 8.3).



(a) Passes that start in the central midfield

(b) The Von Mises distributions learned on pass directions

(c) The passes colored according to the distribution that most likely generated them.

Figure 8.3: Stage 2 of SoccerMix: a mixture model with three Von Mises distributions is fitted to a group of passes that start in the central midfield. In Figure 3b, component 1 (red) illustrates how a single Von Mises distribution can be fitted to observations close to $-\pi$ and π and is thus essential for describing backwards passes.

8.2.3 Distributions of Locations and Directions

The next question to consider is which distributions to use as the components of the mixture models. Locations and directions require a different notion of similarity. In the spatial domain, nearby locations are similar, which we can naturally model using a 2D Gaussian distribution (Figure 8.2) [141]:

$$pdf(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^2|\mathbf{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right) \tag{8.2}$$

where $\boldsymbol{\mu}$ is the mean and $\mathbf{\Sigma}$ is the covariance matrix of the distribution.

When viewed as directions, $-\pi + \epsilon_1$ and $\pi - \epsilon_2$ are similar because directions can be seen as values on a circle rather than on a line. However, a Gaussian distribution would not consider these directions to be similar. Therefore, we model the directions using a Von Mises distribution which arises in the directional statistics literature [13, 112]. Unlike a Gaussian, Von Mises distributions allow for the possibility that observations close to $-\pi$ and observations close to π can be generated by the same distribution (Figure 8.3). The probability density function of a Von Mises distribution is:

$$pdf(\theta) = \frac{1}{2\pi I_0(\kappa)} \exp\left(\kappa \cos(\theta - \mu)\right) \tag{8.3}$$

where μ is the mean direction (the distribution is centered around μ) and κ is a measure of concentration ($\kappa = 0$ means that the distribution is uniform

over the circle while a high value for κ means that the distribution is strongly concentrated around the angle μ). Finally, $I_0(\kappa)$ is the modified Bessel function of order 0, whose exact definition lies beyond the scope of this chapter [112].

8.2.4 Fitting a Mixture Model to the Data

Fitting the parameters of a mixture model to a data set is typically done using the Expectation Maximization algorithm [9]. Given n observations $\{x_1, \ldots, x_n\}$, k distributions $\{F_1, \ldots, F_k\}$, and nk latent variables r_{ij} which denote how likely it is that distribution F_j generated observation x_i, the algorithm iteratively performs the following two steps:

Expectation For each observation x_i and distribution F_j, compute the responsibility r_{ij}, i.e., how likely it is that F_j generated x_i:

$$r_{ij} = \alpha_j \cdot F_j(x_i|\Theta_j).$$

Maximization For each distribution F_j, compute its weight α_j and its parameter set Θ_j. α_j is the prior probability of selecting component j and can be computed as follows:

$$\alpha_j = \frac{\sum_{i=1}^{n} r_{ij}}{\sum_{j=1}^{k}\sum_{i=1}^{n} r_{ij}}.$$

Θ_j is the parameter set that maximizes the likelihood of distribution F_j having generated each observation x_i with probability r_{ij}. To update Θ_j, we employ the distribution-specific update rules detailed below.

It is straightforward to compute the maximum likelihood estimates for the Gaussian distribution's parameter set $\Theta_j = \{\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\}$:

$$\boldsymbol{\mu}_j = \frac{1}{\sum_{i=1}^{n} r'_{ij}} \sum_{i=1}^{n} r'_{ij} \cdot x_i \tag{8.4}$$

$$\boldsymbol{\Sigma}_j = \frac{1}{\sum_{i=1}^{n} r'_{ij}} \sum_{i=1}^{n} r'_{ij} \cdot (x_i - \boldsymbol{\mu}_j)(\boldsymbol{\mu}_j - x_i)^T \tag{8.5}$$

where r'_{ij} is a normalized responsibility computed as:

$$r'_{ij} = \frac{r_{ij}}{\sum_{j=1}^{k} r_{ij}}.$$

Computing the maximum likelihood estimates for the Von Mises distributions is more challenging for two reasons. First, we use the output of the learned location mixture models as input for the direction mixture models. More specifically, each observation x_i has a respective weight $w_i = \alpha_{loc} \cdot F_{loc}(x_i|\Theta_{loc})$ (where F_{loc} is the location distribution we wish to further decompose) that represents the probability of observation x_i being part of the input set of observations for the direction mixture model. These weights w_i necessitate slightly altering how the responsibilities r_{ij} are normalized. Second, learning the parameters for a Von Mises distribution is inherently harder than for Gaussians. Directly estimating κ_j is impossible as its exact equations cannot be analytically solved. Luckily, an approximation using the mean result distance R_j exists that works remarkably well for many practical purposes (Equation 8.7) [112]. We first construct normalized responsibilities r''_{ij} that pretend that each observation x_i in the data set was generated by the mixture model with a probability of w_i and then update the parameter set $\Theta_j = \{\mu_j, \kappa_j\}$ as follows:

$$\mu_j = \text{atan2}\left(\mu_j^{sin}, \mu_j^{cos}\right) \tag{8.6}$$

$$\kappa_j \approx \frac{R_j(2 - R_j^2)}{(1 - R_j^2)} \tag{8.7}$$

where

$$\mu_j^{sin} = \frac{1}{\sum_{i=1}^{n} r''_{ij}} \sum_{i=1}^{n} r''_{ij} \cdot \sin x_i \qquad \mu_j^{cos} = \frac{1}{\sum_{i=1}^{n} r''_{ij}} \sum_{i=1}^{n} r''_{ij} \cdot \cos x_i$$

$$R_j = \sqrt{(\mu_j^{sin})^2 + (\mu_j^{cos})^2} \qquad r''_{ij} = w_i \cdot \frac{r_{ij}}{\sum_{j=1}^{k} r_{ij}}.$$

One of the contributions in this chapter is that we publicly release our implementation of mixture models at `https://github.com/ML-KULeuven/soccermix`. This implementation supports learning a mixture of any type of distribution from a weighted input set of observations.

8.2.5 Practical Challenges

When applying SoccerMix to real-world event stream data, three practical challenges arise. First, the locations in event stream data are approximations. For some actions, such as goal kicks, annotators use a set of predefined start locations instead of its actual location. Therefore we add random noise to the locations and directions of actions to ensure that we do not simply recover the annotation rules for some actions. Second, the mixture models are

sensitive to outliers (e.g., actions with highly irregular locations). Therefore, we preprocess the event stream data to remove outliers using the Local Outlier Factor algorithm [25]. Third, we need to select the number of components used in each mixture model. The number of components needed depends on the action type. For example, passes need more components than corners; a team can perform passes anywhere on the field, but they can take corners from only two locations (the corner flags). We select the number of components in each mixture model by formulating an integer linear programming problem where the goal is to optimize the total Bayesian Information Criterion (BIC) of the entire set of mixture models.[1]

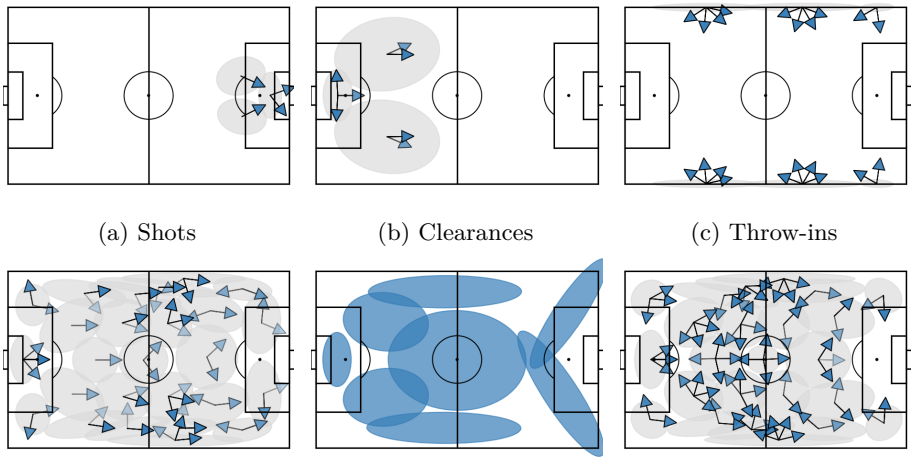
8.2.6 Capturing Playing Style with SoccerMix

Our goal is to construct a vector that describes a specific player's or team's style. Intuitively, SoccerMix discovers groups of similar actions, where each group describes a *prototypical* action of a certain type, location, and direction. Hence, we can use the learned mixture models to encode each action as a probability distribution over all prototypical actions and encode this in a weight vector. We can then build a style vector for a player (team) by summing the weight vectors of all actions performed by that player (team) in a specific time frame (e.g., a game or a season). In the style vector, the weight of an action group can be interpreted as how often a player (team) performed that prototypical action.

8.3 Experiments

In our experiments, we use event stream data provided by Statsbomb for the 2017/18 and 2018/19 seasons of the English Premier League (EPL). Using 400,000 actions sampled from the 2017/18 season, we fitted 2D Gaussian mixture models to the locations of the 23 action types to produce 147 location groups. Next, we fitted Von Mises mixture models to the directions of the actions in those groups to produce 247 groups that describe prototypical actions of a certain type, location, and direction (Figure 8.4). Learning all mixture models took approx. 30 minutes on a computer with 32GB RAM and an Intel i7-6700 CPU @ 3.40GHz with 8 cores. We used these mixture models to produce weight vectors for $\pm$ 2,300,000 actions in 760 games and used those to construct style vectors for 676 players and 23 teams.

[1] More details on our approach to select the number of components used in each mixture model can be found in the public implementation.



(a) Shots (b) Clearances (c) Throw-ins

(d) Dribbles (e) Interceptions (f) Passes

Figure 8.4: Examples of the prototypical actions discovered by SoccerMix. Ellipses denote 2D Gaussian distributions that describe locations. Arrows denote the mean directions of the Von Mises distributions that describe ball directions. Some action types do not directly move the ball and are thus only grouped on location (e.g., interceptions in Figure 4e).

In this section, we first show how the style vectors produced by SoccerMix can be used to identify players based on their playing style. Next, we show how to compare the playing styles of teams and players, along with an approach for capturing the defensive style of teams. Finally, we use our style vectors to take a closer look at the game that cost Liverpool the title to Manchester City in the 2018/19 season and investigate what exactly went wrong.

8.3.1 De-anonymizing Players

No objective definition of playing style exists, which creates challenges. Intuitively, one would expect that in the short-term (i.e., across consecutive seasons) a player's style will not change substantially. Based on this insight, we proposed the following evaluation setup in Chapter 7: "Given anonymized event stream data for a player, is it possible to identify the player based on his playing style in the previous season?"

We perform the exact same player de-anonymization experiment as in Chapter 7 and compare SoccerMix to our player vectors based on non-negative matrix

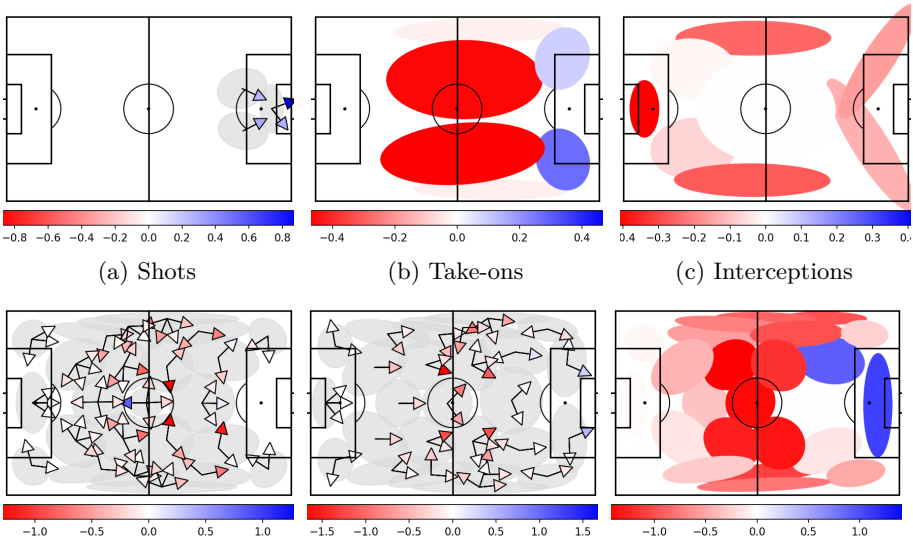
Table 8.1: The top-k results (i.e., the percentage of players whose 2017/18 style vectors are one of the k most similar to their 2018/19 style vectors) and the mean reciprocal rank (MRR) when retrieving 193 players in the English Premier League from anonymized (season 2018/19) and labeled (season 2017/18) event stream data.

Method	Top-1	Top-3	Top-5	Top-10	MRR
SoccerMix	**48.2%**	**62.7%**	**71.5%**	80.8%	**0.589**
Player Vectors (NMF)	36.5%	53.2%	66.5%	**83.2%**	0.505

factorization (NMF). For both approaches, we used the actions of 193 players that appeared and played at least 900 minutes in both seasons. Then, for each player, we compared the distance between the style vector constructed over the 2018/19 season to the style of all players in the 2017/18 season to create a rank-ordered list of the most similar players. Table 8.1 shows how SoccerMix is more successful than the NMF-based player vectors on nearly all ranking metrics. In 48.2% of the cases, SoccerMix correctly identifies a player's style for the current season as being most similar to his previous season's style, which is a 33% relative improvement over the NMF-based approach. Moreover, SoccerMix has a substantially better mean reciprocal rank than the prior approach for this task, which suggests that the style vectors of SoccerMix offer a more complete and accurate view of players' playing style.

8.3.2 Comparing the Playing Style of Players

The style vectors produced by SoccerMix can be used to illustrate the differences in playing style between two players. As an illustrative use case, consider comparing the playing style of Manchester City forward Sergio Agüero and Liverpool forward Roberto Firmino who are both world-class center forwards playing for top teams. Figure 8.5 illustrates the differences in their style vector for shots, take-ons, interceptions, passes, dribbles, and receivals during the 2018/19 EPL season. Spatially, Agüero is more active in the penalty box as he performs more take-ons, dribbles, and ball receivals in that area than Firmino. In contrast, Firmino performs these actions more in the midfield. Finally, the interception map shows that while Agüero does not completely neglect his defensive duties, Firmino plays a more expansive role that sees him also intercept the ball on the flanks and near the penalty box. These insights correspond to Agüero's reputation of being an out-and-out striker who camps out near the opponent's penalty box whereas Firmino often drops deep to facilitate for his attacking partners Mohammed Salah and Sadio Mané. SoccerMix allows generating such figures for any two players which has the potential to aid clubs



(a) Shots (b) Take-ons (c) Interceptions

(d) Passes (e) Dribbles (f) Receivals

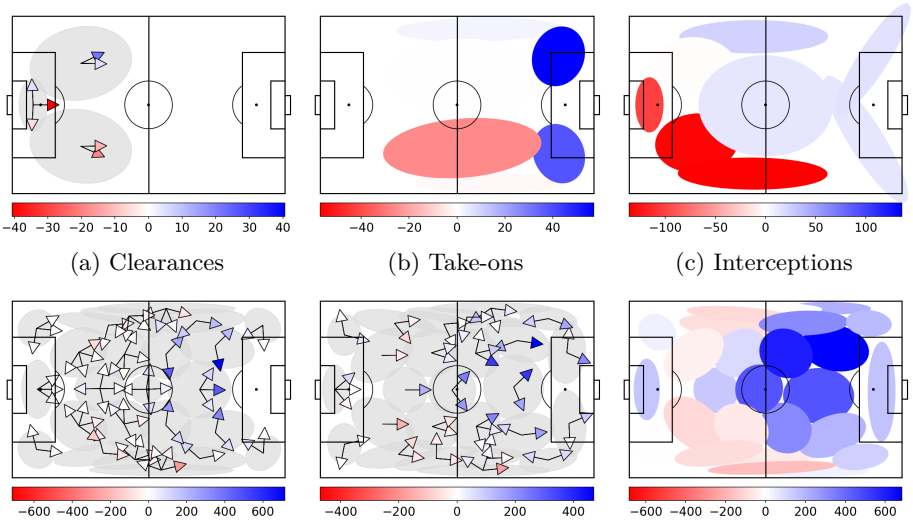
Figure 8.5: Differences in playing style between Manchester City forward Sergio Agüero and Liverpool forward Roberto Firmino during the 2018/19 EPL season. Blue (red) actions indicate that Agüero (Firmino) performed more of these actions than the other. Both players are shown as playing left to right (→). Agüero is more active in the penalty box, while Firmino's actions are more spread out over the midfield.

in player scouting as they can identify players whose style fits how they wish to play.

8.3.3 Comparing the Playing Style of Teams

SoccerMix's style vectors can also be used to compare the playing style of teams. To illustrate this use case, we compare the playing styles of Manchester City and Liverpool, who both completely dominated the 2018/19 English Premier League, finishing at the top of the table with 98 and 97 points respectively with a large 25-point gap to distant third contender Chelsea.

Figure 8.6 shows how Manchester City performs noticeably more take-ons, passes, dribbles, and receivals in the heart of the opponent's half compared to Liverpool. This illustrates how the coaches of both teams have shaped their team's playing style to their own soccer philosophy. Under Jürgen Klopp,



(a) Clearances (b) Take-ons (c) Interceptions

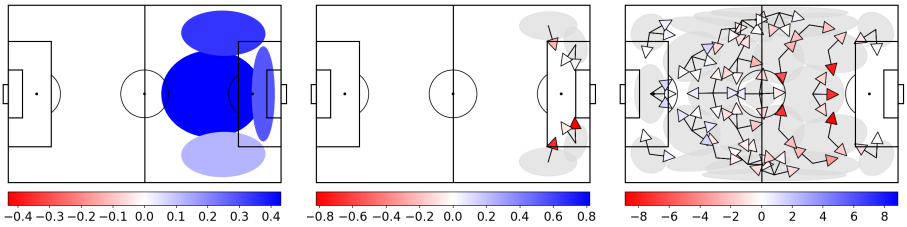
(d) Passes (e) Dribbles (f) Receivals

Figure 8.6: Differences in playing style between Manchester City and Liverpool during the 2018/19 EPL season based on the prototypical action groups obtained with SoccerMix. Blue (red) actions indicate that Manchester City (Liverpool) performed more of these actions than the other team. Both teams are shown as playing left to right (→). Liverpool funnels play towards their right side, while Manchester City generally plays higher up the field.

Liverpool have perfected the art of frequent counter-pressing and speedy counter-attacks. Under Pep Guardiola, Manchester City at times mimics the possession-based, tiki-taka style of its coach's ex-club (FC Barcelona), passing and moving the ball high up on the field.

Additionally, Liverpool seems to funnel the play towards their right side, performing noticeably more clearances, take-ons, and interceptions on their right flank. The most likely source of this uptick is Trent Alexander-Arnold, a right-back at Liverpool who is widely regarded as one of the best attacking full-backs in professional soccer and is a spearhead of Liverpool's transitional, counter-attacking style of play.[2]

[2] `https://sport.optus.com.au/articles/os6422/trent-alexander-arnold-is-changing-the-full-back-position`



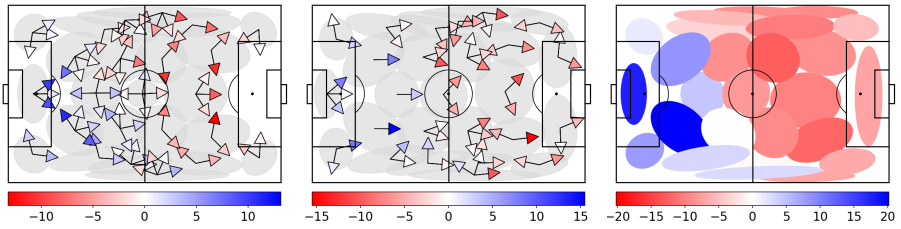
(a) Opponents' offsides (b) Opponents' crosses (c) Opponents' passes

Figure 8.7: Illustrations of how Liverpool (a) employs a good offside trap, (b) has a weaker defense at their right flank when it comes to preventing their opponents from crossing the ball, and (c) forces other teams to play more on their own half. Blue (red) indicates that teams perform more (fewer) of these actions when playing against Liverpool.

8.3.4 Capturing the Defensive Playing Style of Teams

Approaches that capture playing style usually focus on offensive playing style, i.e., what does a team do when in possession of the ball? Analyzing defensive style is much harder as it involves off-the-ball actions such as correct positioning and putting pressure on attackers, which are not recorded in event streams. Our insight is that these off-the-ball actions are often performed with the intention of *preventing certain actions from occurring.* This suggests that we can gain a partial understanding of defensive style by measuring the effects that a team's off-the-ball actions have on what on-the-ball actions their opponent performs. More precisely, we analyze how a team forces its opponents to deviate from their usual playing style.

To illustrate this, we measure the mean difference between teams' style vectors constructed using (1) only the matches against Liverpool and (2) all other matches (i.e., those not involving Liverpool). Figure 8.7 shows how Liverpool causes their opponents, playing left to right, to be flagged more for offside than is typical. This indicates a well-synchronized line of defense that employs a very effective offside trap. The crosses show that, although Liverpool limits the number of crosses its opponents perform, this restriction is not symmetric: they allow fewer crosses from the left of defense (the offense's right) than the right. Lastly, as a combination of both offensive and defensive playing style, Liverpool generally forces the other teams to play more on their own half than on Liverpool's half.



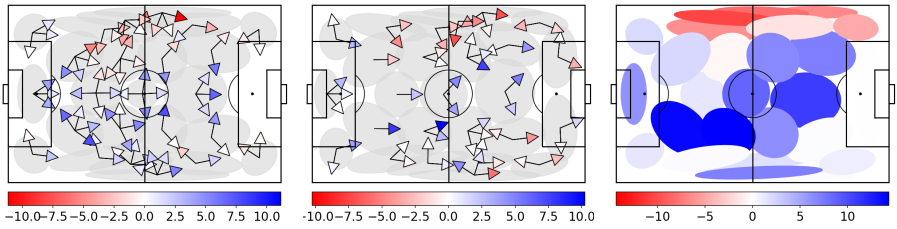
(a) Passes (b) Dribbles (c) Receivals

Figure 8.8: Differences in Liverpool's playing style during their lost away game against Manchester City compared to their style when playing against all other teams in the 2018/19 EPL. Blue (red) indicates Liverpool performing more (fewer) of these actions in their away game against Manchester City. The direction of play is left to right (→).

8.3.5 Case Study: How Liverpool Lost the Title to Manchester City in a Single Game

On January 3rd, 2019, Liverpool held a 6 point lead atop the EPL table when they traveled to play Manchester City in a highly anticipated match. Alas, in their only league loss of the season, Liverpool fell 2-1 and ended up missing out on the title to Manchester City by a single point. It is not a stretch to say that this was the game that cost them the title. Using the concept of style difference vectors from the previous section, Figure 8.8 illustrates how Liverpool's playing style in this game drastically deviated from how they played against other teams. In short, Manchester City maintained their typical high defensive line and forced Liverpool to remain on their own side of the field. This is apparent in both the higher number of passes, dribbles, and receivals Liverpool had to perform deep in their own half as well as the fact that they performed significantly fewer actions than normal in Manchester City's half.

While interesting, it is not completely surprising that Liverpool's offensive output suffered against its only decent rival that season, Manchester City. To dig deeper, we adjust for the level of the opponent and compare Liverpool's playing style in their away game (loss) and home game (draw) against Manchester City in 2018/19 (Figure 8.9). In its away game, Liverpool made noticeably less use of its left flank, performing fewer passes, dribbles, and receivals in that area. This suggests that Liverpool's left flank players were not functioning very well that game, which is further evidenced by midfielder James Milner and winger Sadio Mané on Liverpool's left flank being substituted out in the 57th and 77th minute of the game.



(a) Passes (b) Dribbles (c) Receivals

Figure 8.9: Differences in Liverpool's playing style between their away game and their home game against Manchester City in the 2018/19 EPL. Red (blue) indicates fewer (more) of these actions in the away game than in the home game. Liverpool's left flank players were having a bad day in the away game, as evidenced by the fewer passes, dribbles, and receivals in that area.

8.4 Related Work

Many approaches group actions by overlaying a grid on the field [174, 44, 50]. How they differ is in how they combat the challenges associated with this grid. In Chapter 6, we avoided the sparsity issues of a fine-grained grid by dividing the field into only four zones (left-flank, midfield, right-flank, and penalty box), as the performance of the pattern mining algorithm rapidly declined when using a more fine-grained grid. However, those patterns can then only describe ball movements between these four zones and are thus too broad and simple to be able to identify unique characteristics related to playing style. Van Haaren et al. [174] attempted to combine the advantages of both coarse and fine-grained grids by encoding action locations on multiple granularity levels. However, they found that this multi-level representation of actions blew up the search space of their inductive logic programming approach and led to heavy computational costs.

In Chapter 7, we applied a post-processing step to the counts of a fine-grained grid. More specifically, the count of each grid cell is replaced by a weighted mean of itself and its neighboring grid cells, which promotes the spatial coherence between grid cells and combats issues such as sparsity and abrupt boundaries. However, there are two downsides to this approach. First, a new technique with its own parameters (that are non-trival to tune) is added to the analysis pipeline. Second, this approach encourages dividing actions into a number of groups that is excessive for representing the characteristics of the data, which makes it difficult for automated systems to numerically process the new data representation and for humans to interpret the end results. For example, in

Chapter 7, we use $50 \times 50 = 2500$ grid cells to represent shot behavior of players (of which most will be empty), while SoccerMix only needs 3 location groups to represent shot behavior.

8.5 Conclusion

Capturing the playing style of teams and players in soccer can be leveraged in areas such as player scouting and match preparation. In this chapter we introduced SoccerMix: an approach to intelligently partition player actions into groups of similar actions. Intuitively, each group describes a prototypical action with a specific type, location, and direction. We have shown how SoccerMix can be used to capture the playing style of both teams and players. Additionally, we introduced a new way to capture the defensive playing style of a team by using deviations in the actions of that team's opponents. Finally, we have publicly released SoccerMix's implementation at `https://github.com/ML-KULeuven/soccermix`.

Chapter 9

Conclusions

This chapter summarizes the main contributions and conclusions of this dissertation. It also offers some general "lessons learned" from the perspective of both soccer analytics and artificial intelligence and provides possible directions for future work.

9.1 Summary

The objective of this dissertation was to learn value and style from event stream data using a number of techniques from the research field of artificial intelligence. Towards this end, we identified a number of shortcomings in the existing literature and made contributions in three broad areas of soccer analytics: (1) representing event stream data, (2) valuing on-the-ball actions, and (3) capturing the playing style of teams and players.

First, to better represent event stream data, we defined a new language that simplifies and unifies the data of different event stream data vendors, thereby encouraging the reproducibility of the soccer analytics research field. Additionally, we released a software package to convert event stream data from prominent vendors into this new language that, as of September 2020, has been downloaded approximately 9000 times.

Second, to assign values to on-the-ball actions, we created a framework that goes beyond simple metrics and possession-based approaches, which do not recognize the value of defensive actions or consider the full context in which actions are performed. Our framework uses a simple and elegant formula that

formalizes the intuition that all actions in a match are performed with the intention of increasing the chance of scoring a goal and/or decreasing the chance of conceding a goal.

To capture the playing style of teams and players, we used AI techniques such as non-negative matrix factorization and mixture models to model the spatial component of event stream data. These techniques overcome a number of issues (e.g., data sparsity, abrupt boundaries) that tend to plague approaches for capturing playing style that manually divide the pitch into simple zones to model the locations of actions. Additionally, we also introduced a new way to capture the defensive playing style of a team by using deviations in the playing style of that team's opponents.

We summarize our contributions below and discuss the strengths and weaknesses of each contribution.

9.1.1 Representing Event Stream Data

SPADL [42] is a language to describe player actions that alleviates a number of engineering challenges commonly found in raw event stream data produced by vendors such as Opta Sports [126], Wyscout [185], and StatsBomb [165]. SPADL offers three advantages over raw event stream data. First, SPADL focuses exclusively on on-the-ball actions, ignoring less relevant event such as weather changes. Second, SPADL unifies event stream data such that analysis projects are transferable between event stream data vendors. Third, SPADL always stores the same eight attributes per action, which makes actions much easier to automatically parse compared to event stream data formats that allow the presence of optional information snippets.

We also introduced Atomic-SPADL [48], which is a variant of SPADL that removes the "result" attribute and adds a few new actions and event types. For example, Atomic-SPADL treats the initiation and receival of passes as separate actions. The advantages of Atomic-SPADL are that it can reward players' contributions more fairly (e.g., a pass-giver will not be punished for a mistake made by the intended pass receiver) and also produce more robust action values than regular SPADL [48].

A Python package to automatically convert event stream data from Opta, Wyscout, and StatsBomb to SPADL and Atomic-SPADL is available at `https://github.com/ML-KULeuven/socceraction`.

9.1.2 Learning Value from Event Stream Data

In this dissertation, we introduced two approaches to value on-the-ball actions: STARSS and VAEP.

STARSS

STARSS [51] is an approach to automatically rate the actions performed by soccer players. Viewing a soccer match as a sequence of actions performed by players, STARSS performs three steps to rate these actions. First, it splits the match into phases of related actions. Second, it assigns a rating to each phase, indicating how likely it is that the phase will end in a goal. Third, it distributes the assigned rating over the individual actions that constitute the phase.

The strength of STARSS is that it is an approach to value actions that goes beyond shots and goals. A weakness of STARSS is that it uses a predefined formula to distribute the rating of a phase over its constituent actions rather than take the specific characteristics and composition of the actions into account. Additionally, STARSS is a possession-based approach that can only consider the potential reward - and not the risk - of an action.

VAEP

VAEP [42] is a framework for valuing any type of player action based on its impact on the game outcome while accounting for the context in which the action happened. This framework enables giving a broad overview of a player's performance, including quantifying a player's total offensive and defensive contributions to their team.

Some strengths of VAEP are its elegant and simple formula and its ability to consider a more complete view of an action's context than earlier work. Additionally, it is the first framework to quantify not just the offensive value, but also the defensive value of an action by modelling the probability that the opposing team will score in the near future. A weakness of VAEP is that, in practice, the location of an actions is still by far the most dominant factor to determine an action's value. Consequently, most actions on the midfield (which comprises the vast majority of actions in a game) get assigned values close to zero and are thus still almost irrelevant when determining a players' worth. Another weakness is that VAEP's output is sensitive to the choice of learning algorithm and the used training data. Compared to other metrics such as Expected Threat [159], this makes VAEP's player ratings less robust [178].

However, preliminary research suggests that combining VAEP with Atomic-SPADL alleviates the sensitivy issue [48].

An implementation of VAEP is available at `https://github.com/ML-KULeuven/socceraction`.

9.1.3 Learning Style from Event Stream Data

In this dissertation, we introduced three approaches to learn style from event stream data: Tactics Discovery using sequential pattern mining algorithms, Player Vectors using non-negative matrix factorization, and SoccerMix using mixture models.

Tactics Discovery

We introduced an approach to mine patterns in event stream data and characterize teams by their most frequent patterns [50]. More specifically, our approach follows the following five steps: (1) divide each match into phases (2) cluster the phases using Dynamic Time Warping [123], (3) rank the clusters using user-preferences, (4) mine each cluster using sequential pattern mining algorithm [68], (5) rank the discovered patterns on user-preferences. The most difficult challenge lied in step four, which involves converting the event stream into a format compatible with sequential pattern mining algorithms.

The strength of our approach is its ability to simultaneously take many properties of event stream data into account such as actions' type, location, involved players, and relative ordering. A weakness of our approach is that the discretization of the locations of actions could not be done in a fine-grained manner without hindering the pattern mining step. As a result, this approach is not yet ready to be turned into a practical tool that generates actionable insights. Additionally, most domain experts found the intermediate spatial clusters generated by our approach more interesting than the final symbolic patterns.

Player Vectors

Player vectors [44] succinctly express where on the pitch a player tends to perform passes, dribbles, crosses, and shots. These vectors are constructed as follows. First, we overlay a grid on the pitch and construct heatmaps for each player and action type by counting how often a player performs those actions in each grid cell. Next, we compress these heatmaps to short vectors

using non-negative matrix factorization (NMF) [102] and concatenate together a player's compressed vectors for each action type to construct his player vector.

The strengths of our approach are its interpretability and its robustness. The resulting components of the factorization are interpretable as spatial groups of a specific action type (e.g., a shot close to the goal, a center dribble, a pass from the left back to the left flank) and the player vectors themselves can be seen as quantifying how often each player performs actions of a certain type and spatial group. Furthermore, our approach will usually rediscover the same principal components, even when applied to event stream data of different seasons, leagues, and vendors. The most important weakness of our NMF-based approach is its inability to handle sparsity when using a highly granular grid. To make their decomposition feasible, we had to apply a smoothing step to the heatmaps of players' passes, dribbles, crosses, and shots. For rarer actions such as tackles, clearances, and keeper saves, the decomposition was impossible even after applying this smoothing step.

SoccerMix

SoccerMix [52] is an approach based on mixture models to intelligently partition player actions into groups of similar actions. Intuitively, each group found by SoccerMix describes a prototypical action with a specific type, location, and direction.

The strength of SoccerMix is that it overcomes the problems (i.e., sparsity and abrupt boundaries) associated with grid-based approaches. Additionally, SoccerMix's interpretable action groups can be used to visually compare the playing style of teams and players. One weakness of SoccerMix is that, compared to the player vectors in Chapter 7, the configuration of the discovered action groups is more sensitive to the underlying training data. Additionally, choosing the correct number of components of each mixture model is non-trivial and currently requires some manual tuning.

An implementation of SoccerMix is available at `https://github.com/ML-KULeuven/soccermix`.

9.2 Lessons Learned

In this dissertation, we studied how the research fields of soccer analytics and artificial intelligence can be combined and what both research fields have to offer

to each other. We now summarize some general lessons learned and advice for respectively soccer analytics practitioners and artificial intelligence researchers.

9.2.1 Lessons Learned for Soccer Analytics

We discuss five lessons learned in this dissertation from the perspective of soccer analytics.

Representation. The quality of the data representation plays a large part in the success of an analysis task. A special point of attention here is using the correct levels of abstraction for various concepts in soccer. For example, Opta [126] encodes shots in four different ways depending on the end result (goal, miss, save by the keeper, ball hit the goal post). Rather than adapt a model to work with these four different event types, it is much easier to preprocess the data such that shots are always encoded as "shot" actions [42].

Visualization. Visualizing the data is often more useful than inspecting raw values. For example, it is much easier to perform a sanity check on the values of actions plotted out on a soccer pitch, than on the raw (x, y)-coordinates of those actions. Similarly, the heatmaps and mixture models in Chapters 7 and 8 only have meaning when plotted out on a pitch; they are impossible to interpret based on their numerical values alone.

Early Results. No analysis pipeline is ever perfect on the first try. Either the method does not work exactly as intended or the data contains some previously unknown quirks. Therefore it is always useful to inspect (intermediate) results as early as possible. For example, in Chapter 6 we first verified the quality of the intermediate spatial clusters before proceeding with the sequential pattern mining step for each cluster. In Chapter 8, we immediately plotted out the locations of groups of actions and realized that we needed to remove some outliers before modelling those actions with mixture models.

Spatial Component. One aspect that we struggled with throughout this dissertation is the spatial component of event stream data. We applied a number of techniques such as Dynamic Time Warping [123], gradient boosted decision trees [35], grid discretization [44], and mixture models [9]. Figuring out the best way to handle the spatial component is non-trivial as each approach has its own strengths and weaknesses. However, it is worth it to rigorously research the best way to handle locations as this will often be by far the most impactful modelling choice when learning value or style from event stream data.

Artifical Intelligence. The field of artificial Intelligence offers a plethora of techniques that can be used to answer important questions in soccer analytics. The following non-exhaustive list includes some illustrative examples. Binary probabilistic classification models have been used to value actions [20, 42]. Pattern mining [50, 12, 175], motif discovery [12], deep learning [101], inductive logic programming [174], and non-negative matrix factorization [44] have been used to understand tactics and playing style. Reinforcement learning [53] and Markov decision processes [159] have been used to analyze technical performance in a match. Multi-agent learning has been used to learn the value of teamwork from outcomes of pass interactions between players [11]. Predictive modeling has been used to understand and manage training load [92]. Deep learning [94] and integer linear programming [157] have been used for tracking players during a match.

9.2.2 Lessons Learned for Artificial Intelligence

We discuss five lessons learned in this dissertation from the perspective of artificial intelligence.

Representation. The first lesson learned for soccer analytics practitioners is equally relevant for artificial intelligence researchers. Finding the best representation for the data and using our understanding of the application domain to construct the correct abstractions often has a much larger impact on the success of an analysis task than fine-tuning the parameters of employed AI models.

Domain Knowledge. Incorporating domain knowledge can improve the performance of AI models. For example, the location of the goal is extremely relevant for expected-goals models as the accuracy of these models drastically improves when the distance and angle to the goal are used as features instead of the raw (x, y)-coordinates of shots [143]. Another example is using the insight that players' playing style does not drastically change over subsequent seasons to evaluate approaches that capture playing style (Chapters 7 and 8).

Interpretability. Ultimately, experts are interested in translating the findings arising from analytics into practice. These findings should always be intuitive and interpretable without fully understanding the inner workings of the model that produced them. One way to facilitate this is to report these findings in a visual way; for example, by plotting them out on a soccer pitch.

Ground Truth. Real-world data often lacks ground truth labels. These may be hard to obtain or simply not exist. For example, in soccer there are no objectively best players and a team's tactical plan is only known to the team itself. However, the richness of the soccer domain encourages researchers to devise creative solutions for this problem. Some examples are measuring correlation with proxy signals such as player transfer value [42], journalist opinions on playing style [44], and evaluating results by asking domain experts [21, 127].

Soccer Analytics. The field of soccer analytics is a fruitful source of real-world data and a testbed for many AI techniques. Additionally, a number of core areas in artificial intelligence such as interpretability, decision making, representation learning, and understanding behavior are all also relevant areas within soccer analytics.

9.3 Future Work

The contributions of this dissertation provided valuable new methods and insights for learning value and style from event stream data. Nevertheless, there are still many open questions and challenges in the field. This section presents several possible directions for future work.

Expand SPADL

The SPADL format [44] is easy to process, but sometimes lacks depth. Because it unifies the data of different event stream data vendors, it is in a way their greatest common divisor, only including information that is present in all possible sources of event stream data. Hence, it could be interesting to expand SPADL with useful information not necessarily collected by all vendors such as whether defenders are putting pressure on the player currently in possession of the ball or the locations of relevant players during key events such as shots. However, the specifics of this expansion will need to be done carefully and with attention for software engineering principles in order to not drastically increase the engineering costs when analyzing event stream data.

Learn Values with SoccerMix

SoccerMix [52] is a completely new way to represent individual soccer actions. In this dissertation, we used this probabilistic representation to capture the

playing style of teams and player, but we could also use SoccerMix to value actions. For example, we can use each action's probability distribution over SoccerMix's prototypical actions as input features for VAEP [42], or we can view each action in the data as being in a quantum state of prototypical actions and use these states in a Markov model-based approach such as Expected Threat [159].

Simulate Event Stream Data

Despite commendable efforts by Wyscout [185], StatsBomb [165], and Metrica Sports [120] in recent years, event stream data is still a scarce resource. An approach that could generate virtual event stream data indistinguishable from actual event stream data, would be immensely helpful to further democratize the field of soccer analytics. For example, SoccerMix's probabilistic representation for actions could be used to construct a generative model. Another line of research that would be interesting to further explore here is the field of Generative Adversarial Networks [77].

Model Relations Between Consecutive Actions

One aspect that is currently underexplored in soccer analytics is the relationships between consecutive actions. For example, VAEP does not care much about the actions preceding the most recent action other than to estimate the current speed of play. Similarly, Expected Threat uses no more information than the current zone in which an action occurred. Finally, the two approaches presented in this dissertation to capture the playing style of players (Player Vectors and SoccerMix) both only consider the individual actions a player performs rather than the phases of consecutive actions in which the player participates.

For now it remains an open question whether a past action has any effect on the success of future actions further down the line or if current methods are simply unable to capture these relations. We suspect that the biggest challenge in modelling these relations is the sparsity of the resulting interaction tensors. Hence, it could be interesting to look at techniques that can efficiently compress sparse matrices from research fields such as tensor factorization [158] or recommender systems [99].

Consider Player/Team-Information when Valuing Actions

All models that value actions are currently agnostic towards the player or team involved in the action. However, one could argue that a dribble made by Lionel Messi tends to have a higher chance of succeeding than a dribble made by the average soccer player; and that a model for valuing actions should consequently take this into account when determining the value of that dribble.

One way to account for this would be to construct a fingerprint of each player/team that captures the value and style of that player/team (perhaps using techniques from this dissertation) and add this fingerprint to the feature set of each individual action when training a model to value actions. However, we suspect that this approach would require gargantuan amounts of training data and would also raise a number of concerns related to how applicable the learned model would be to unseen data. Another interesting avenue to explore here is to again consider this as a problem related to sparsity (i.e. there are too few examples of the same player doing the same action in the same location) and look at techniques from the field of recommender systems such as Factorization Machines [140].

Bibliography

[1] Adewoye, G. Everton Boss Sam Allardyce Compares Idrissa Gueye to N'Golo Kante. http://www.goal.com/en/news/everton-boss-sam-allardyce-compares-idrissa-gueye-to-ngolo/gddgazktcl3b1ayeadrva1o18.

[2] Agrawal, R., and Srikant, R. Fast Algorithms for Mining Association Rules. In *Proceedings of the 20th International Conference on Very Large Data Bases, VLDB* (1994), vol. 1215, pp. 487–499.

[3] Agrawal, R., and Srikant, R. Mining Sequential Patterns. In *Proceedings of the eleventh International Conference on Data Engineering* (1995), IEEE, pp. 3–14.

[4] Alamar, B., and Mehrotra, V. Beyond 'Moneyball': The Rapidly Evolving World of Sports Analytics, Part I. *Analytics Magazine* (2011).

[5] Alpaydin, E. *Introduction to Machine Learning*. MIT press, 2020.

[6] Altman, D. Beyond Shots: A New Approach to Quantifying Scoring Opportunities. OptaPro Analytics Forum, 2015.

[7] Anderson, C., and Sally, D. *The Numbers Game: Why Everything You Know About Football is Wrong*. Penguin UK, 2013.

[8] Ayres, J., Flannick, J., Gehrke, J., and Yiu, T. Sequential Pattern Mining using a Bitmap Representation. In *Proceedings of the eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2002), pp. 429–435.

[9] Bailey, T. L., Elkan, C., et al. Fitting a Mixture Model by Expectation Maximization to Discover Motifs in Bipolymers. *Proceedings of the International Conference on Intelligent Systems for Molecular Biology 2* (1994), 28–36.

[10] BallJames. http://www.balljames.com. Accessed: 2020-06-27.

[11] Beal, R., Changder, N., Norman, T. J., and Ramchurn, S. D. Learning the Value of Teamwork to Form Efficient Teams. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence* (2020).

[12] Bekkers, J., and Dabadghao, S. Flow Motifs in Soccer: What can Passing Behavior Tell Us? *Journal of Sports Analytics*, Preprint (2017), 1–13.

[13] Best, D., and Fisher, N. I. Efficient Simulation of the von Mises Distribution. *Journal of the Royal Statistical Society: Series C (Applied Statistics) 28*, 2 (1979), 152–157.

[14] Bialkowski, A., Lucey, P., Carr, P., Yue, Y., and Matthews, I. "Win at Home and Draw Away": Automatic Formation Analysis Highlighting the Differences in Home and Away Team Behaviors. *Procedings of MIT Sloan Sports Analytics* (2014).

[15] Bialkowski, A., Lucey, P., Carr, P., Yue, Y., Sridharan, S., and Matthews, I. Identifying Team Style in Soccer Using Formations Learned from Spatiotemporal Tracking Data. In *Data Mining Workshop (ICDMW), 2014 IEEE International Conference on* (2014), IEEE, pp. 9–14.

[16] Bialkowski, A., Lucey, P., Carr, P., Yue, Y., Sridharan, S., and Matthews, I. Large-Scale Analysis of Soccer Matches using Spatiotemporal Tracking Data. In *2014 IEEE International Conference on Data Mining* (2014), IEEE, pp. 725–730.

[17] Bishop, C. M. *Pattern Recognition and Machine Learning*. springer, 2006.

[18] Bojinov, I., and Bornn, L. The Pressing Game: Optimal Defensive Disruption in Soccer. *Proceedings of the MIT Sloan Sports Analytics Conference* (2016).

[19] Boshnakov, G., Kharrat, T., and McHale, I. G. A Bivariate Weibull Count Model for Forecasting Association Football Scores. *International Journal of Forecasting 33*, 2 (2017), 458–466.

[20] Bransen, L. Valuing Passes in Football Using Ball Event Data. Master's thesis, Erasmus University Rotterdam, 2017.

[21] Bransen, L., Robberechts, P., Van Haaren, J., and Davis, J. Choke or Shine? Quantifying Soccer Players' Abilities to Perform Under

Mental Pressure. In *Proceedings of the MIT Sloan Sports Analytics Conference* (2019).

[22] Bransen, L., and Van Haaren, J. Measuring Football Players' On-the-Ball Contributions from Passes During Games. In *ECML/PKDD 2018 Workshop on Machine Learning and Data Mining for Sports Analytics* (2018).

[23] Bransen, L., and Van Haaren, J. Player Chemistry: Striving for a Perfectly Balanced Soccer Team. In *MIT Sloan Sports Analytics Conference* (2020).

[24] Bransen, L., Van Haaren, J., and van de Velden, M. Measuring Soccer Players' Contributions to Chance Creation by Valuing Their Passes. *Journal of Quantitative Analysis in Sports* (2019).

[25] Breunig, M. M., Kriegel, H.-P., Ng, R. T., and Sander, J. LOF: Identifying Density-based Local Outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data* (2000), pp. 93–104.

[26] Brooks, J., Kerr, M., and Guttag, J. Using Machine Learning to Draw Inferences from Pass Location Data in Soccer. *Statistical Analysis and Data Mining: The ASA Data Science Journal 9*, 5 (2016), 338–349.

[27] Caley, M. Premier League Projections and New Expected Goals. `http://cartilagefreecaptain.sbnation.com/2015/10/19/9295905/premier-league-projections-and-new-expected-goals`, 2015.

[28] Callaghan, S. Everton Boss was Spot-On with Idrissa Gueye - N'Golo Kante Comparison. `http://www.hitc.com/en-gb/2018/04/12/everton-boss-was-spot-on-with-idrissa-gueye-ngolo-kante-comparis/`, 2018.

[29] Cao, H., Mamoulis, N., and Cheung, D. Mining Frequent Spatio-temporal Sequential Patterns. In *Proceedings of the 5th International Conference on Data Mining* (11 2005), pp. 82–89.

[30] Castellano, J., Álvarez, D., Figueira, B., Coutinho, D., and Sampaio, J. Identifying the Effects from the Quality of Opposition in a Football Team Positioning Strategy. *International Journal of Performance Analysis in Sport 13*, 3 (2013), 822–832.

[31] Catapult. `http://www.catapultsports.com`, 2020. Accessed: 2020-06-27.

[32] Cervone, D., D'Amour, A., Bornn, L., and Goldsberry, K. POINTWISE: Predicting Points and Valuing Decisions in Real Time with NBA Optical Tracking Data. In *Proceedings of the 8th MIT Sloan Sports Analytics Conference, Boston, MA, USA* (2014), vol. 28.

[33] Charnes, A., Cooper, W. W., and Rhodes, E. Measuring the Efficiency of Decision Making Units. *European Journal of Operational Research 2*, 6 (1978), 429–444.

[34] Chawla, S., Estephan, J., Gudmundsson, J., and Horton, M. Classification of Passes in Football Matches using Spatiotemporal Data. *ACM Transactions on Spatial Algorithms and Systems (TSAS) 3*, 2 (2017), 1–30.

[35] Chen, T., and Guestrin, C. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016), ACM, pp. 785–794.

[36] ChyronHego. `http://www.chyronhego.com`, 2020. Accessed: 2020-06-27.

[37] Cintia, P., Rinzivillo, S., and Pappalardo, L. A Network-based Approach to Evaluate the Performance of Football Teams. In *Machine Learning and Data Mining for Sports Analytics Workshop, Porto, Portugal* (2015).

[38] Collins, T. 4 Possible Replacements Should Real Madrid Sell Sergio Ramos, 2015. `http://bleacherreport.com/articles/2509541-4-possible-replacements-should-real-madrid-sell-sergio-ramos`.

[39] Constantinou, A. C., and Fenton, N. E. Determining the Level of Ability of Football Teams by Dynamic Ratings based on the Relative Discrepancies in Scores between Adversaries. *Journal of Quantitative Analysis in Sports 9*, 1 (2013), 37–50.

[40] Danneels, G., Van Haaren, J., Op De Beéck, T., and Davis, J. Identifying Playing Styles in Professional Football. *KU Leuven* (2014).

[41] Davis, J., Bransen, L., Decroos, T., Robberechts, P., and Van Haaren, J. Assessing the Performances of Soccer Players. In *Proceedings of the 12th International Symposium on Computer Science in Sport (IACSS 2019)* (Cham, 2019), M. Lames, A. Danilov, E. Timme, and Y. Vassilevski, Eds., Springer International Publishing, pp. 3–10.

[42] Decroos, T., Bransen, L., Van Haaren, J., and Davis, J. Actions Speak Louder than Goals: Valuing Player Actions in Soccer. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2019), KDD '19, ACM, pp. 1851–1861.

[43] Decroos, T., Bransen, L., Van Haaren, J., and Davis, J. VAEP: An Objective Approach to Valuing On-the-Ball Actions in Soccer (Extended Abstract). In *Proceedings of the 29th International Joint Conference on Artificial Intelligence* (2020), AAAI Press.

[44] Decroos, T., and Davis, J. Player Vectors: Characterizing Soccer Players' Playing Style from Match Event Streams. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (2019), Springer.

[45] Decroos, T., and Davis, J. Interpretable Prediction of Goals in Soccer. In *AAAI 2020 Workshop on Artificial Intelligence in Team Sports* (2020).

[46] Decroos, T., De Causmaecker, P., and Demoen, B. Solving Euclidean Steiner Tree Problems with Multi Swarm Optimization. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation* (2015), ACM; New York, NY, USA, pp. 1379–1380.

[47] Decroos, T., Dzyuba, V., Van Haaren, J., and Davis, J. Predicting Soccer Highlights from Spatio-Temporal Match Event Streams. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence* (2017), pp. 1302–1308.

[48] Decroos, T., Robberechts, P., and Davis, J. Introducing Atomic-SPADL: A New Way to Represent Event Stream Data. DTAI Sports Analyics Lab Blog, 2020. `https://dtai.cs.kuleuven.be/sports/blog/introducing-atomic-spadl-a-new-way-to-represent-event-stream-data`.

[49] Decroos, T., Schütte, K., De Beéck, T. O., Vanwanseele, B., and Davis, J. AMIE: Automatic Monitoring of Indoor Exercises. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (2018), Springer, pp. 424–439.

[50] Decroos, T., Van Haaren, J., and Davis, J. Automatic Discovery of Tactics in Spatio-Temporal Soccer Match Data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2018), pp. 223–232.

[51] Decroos, T., Van Haaren, J., Dzyuba, V., and Davis, J. STARSS: A Spatio-Temporal Action Rating System for Soccer. In *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2017 workshop* (2017).

[52] Decroos, T., Van Roy, M., and Davis, J. SoccerMix: Representing Soccer Actions with Mixture Models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (2020), Springer.

[53] Dick, U., and Brefeld, U. Learning to Rate Player Positioning in Soccer. *Big data 7*, 1 (2019), 71–82.

[54] Dixon, M. J., and Coles, S. G. Modelling Association Football Scores and Inefficiencies in the Football Betting Market. *Journal of the Royal Statistical Society: Series C (Applied Statistics) 46*, 2 (1997), 265–280.

[55] Duch, J., Waitzman, J. S., and Amaral, L. A. N. Quantifying the Performance of Individual Players in a Team Activity. *PloS one 5*, 6 (2010), e10937.

[56] Elo, A. E. *The Rating of Chessplayers, Past and Present.* Arco Publications, 1978.

[57] Everitt, B. S., Landau, S., Leese, M., and Stahl, D. *Cluster Analysis.* John Wiley and Sons, Ltd, 2011.

[58] Everitt, B. S., Landau, S., Leese, M., and Stahl, D. Hierarchical Clustering. *Cluster Analysis, 5th Edition* (2011), 71–110.

[59] Fernández, J., and Bornn, L. Wide Open Spaces: A Statistical Technique for Measuring Space Creation in Professional Soccer. In *Sloan Sports Analytics Conference* (2018), vol. 2018.

[60] Fernández, J., Bornn, L., and Cervone, D. Decomposing the Immeasurable Sport: A Deep Learning Expected Possession Value Framework for Soccer. In *MIT Sloan Sports Analytics Conference* (2019).

[61] Fernandez-Navarro, J., Fradua, L., Zubillaga, A., Ford, P. R., and McRobert, A. P. Attacking and Defensive Styles of Play in Soccer: Analysis of Spanish and English Elite Teams. *Journal of Sports Sciences 34*, 24 (2016), 2195–2204.

[62] Fernando, T., Wei, X., Fookes, C., Sridharan, S., and Lucey, P. Discovering Methods of Scoring in Soccer Using Tracking Data. In *Proceedings of the Workshop on Large-Scale Sports Analytics* (2015).

[63] Ferri, C., Hernández-Orallo, J., and Modroiu, R. An Experimental Comparison of Performance Measures for Classification. *Pattern Recognition Letters 30*, 1 (2009), 27–38.

[64] Flynn, M. STATS Playing Styles – An Introduction, 2016.

[65] Fonseca, S., Milho, J., Travassos, B., and Araújo, D. Spatial Dynamics of Team Sports Exposed by Voronoi Diagrams. *Human Movement Science 31*, 6 (2012), 1652–1659.

[66] Fournier-Viger, P., Gomariz, A., Campos, M., and Thomas, R. Fast vertical mining of sequential patterns using co-occurrence information. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (2014), Springer, pp. 40–52.

[67] Fournier-Viger, P., Gomariz, A., Gueniche, T., Soltani, A., Wu, C.-W., and Tseng, V. S. SPMF: A Java Open-Source Pattern Mining Library. *Journal of Machine Learning Research 15* (2014), 3389–3393.

[68] Fournier-Viger, P., Lin, J. C.-W., Kiran, R. U., Koh, Y. S., and Thomas, R. A Survey of Sequential Pattern Mining. *Data Science and Pattern Recognition 1*, 1 (2017), 54–77.

[69] Franks, A., Miller, A., Bornn, L., and Goldsberry, K. Characterizing the Spatial Structure of Defensive Skill in Professional Basketball. *Annals of Applied Statistics 2015, Vol. 9, No. 1* (2015), 94–121. arXiv:1405.0231.

[70] Fujimura, A., and Sugihara, K. Geometric Analysis and Quantitative Evaluation of Sport Teamwork. *Systems and Computers in Japan 36*, 6 (2005), 49–58.

[71] Garmin. `http://www.garmin.com`, 2020. Accessed: 2020-06-27.

[72] Geerts, A., Decroos, T., and Davis, J. Characterizing Soccer Players' Playing Style from Match Event Streams. In *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2018 workshop* (2018), vol. 2284, Springer, pp. 115–126.

[73] Giannotti, F., Nanni, M., Pinelli, F., and Pedreschi, D. Trajectory Pattern Mining. In *Proceedings of the 13th International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2007), ACM, pp. 330–339.

[74] Goal.com. Messi Admits Difficulties in Dybala Partnership: He Plays Like Me at Juve. `http://www.goal.com/en/news/messi-`

admits-difficulties-in-dybala-partnership-he-plays-like-me-/1uq96ju5zageb1s1vez93omsi3. Accessed: 2019-05-05.

[75] Goes, F. R., Kempe, M., Meerhoff, L. A., and Lemmink, K. A. Not Every Pass can be an Assist: A Data-Driven Model to Measure Pass Effectiveness in Professional Soccer Matches. *Big data 7*, 1 (2019), 57–70.

[76] Goldner, K. A Markov Model of Football: Using Stochastic Processes to Model a Football Drive. *Journal of Quantitative Analysis in Sports 8*, 1 (2012).

[77] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems* (2014), pp. 2672–2680.

[78] Green, S. Assessing the Performance of Premier League Goalscorers. Opta Sports Blog, 2012. https://www.optasportspro.com/news-analysis/assessing-the-performance-of-premier-league-goalscorers.

[79] Gregory, S. How We Assign Credit in Football. Opta Sports Blog, 2017. http://www.optasportspro.com/about/optapro-blog/posts/2017/blog-how-we-assign-credit-in-football/.

[80] Gudmundsson, J., and Wolle, T. Football Analysis using Spatio-Temporal Tools. *Computers, Environment and Urban Systems 47* (2014), 16–27.

[81] Gyarmati, L., and Anguera, X. Automatic Extraction of the Passing Strategies of Soccer Teams. *arXiv preprint arXiv:1508.02171* (2015).

[82] Gyarmati, L., and Hefeeda, M. Analyzing In-Game Movements of Soccer Players at Scale. *arXiv preprint arXiv:1603.05583* (2016).

[83] Gyarmati, L., Kwak, H., and Rodriguez, P. Searching for a Unique Style in Soccer. *arXiv preprint arXiv:1409.0308* (2014).

[84] Gyarmati, L., and Stanojevic, R. QPass: A Merit-based Evaluation of Soccer Passes. In *KDD 2016 Workshop on Large-Scale Sports Analytics* (2016).

[85] Hastie, T. J. Generalized Additive Models. In *Statistical models in S*. Routledge, 2017, pp. 249–307.

[86] Hirotsu, N., Wright, M., et al. Using a Markov Process Model of an Association Football Match to Determine the Optimal Timing of Substitution and Tactical Decisions. *Journal of the Operational Research Society 53*, 1 (2002).

[87] Hvattum, L. M., and Arntzen, H. Using ELO Ratings for Match Result Prediction in Association Football. *International Journal of Forecasting 26*, 3 (2010), 460–470.

[88] IJtsma, S. A Close Look at My New Expected Goals Model. 11tegen11, 2015. http://11tegen11.net/2015/08/14/a-close-look-at-my-new-expected-goals-model/.

[89] IJtsma, S. The Best Predictor for Future Performance Is Expected Goals. 11tegen11, 2015. http://11tegen11.net/2015/01/05/the-best-predictor-for-future-performance-is-expected-goals.

[90] Impey, S. STATS and Perform Confirm Merger. *SportsPro* (Jul 2019). https://www.sportspromedia.com/news/stats-perform-merger-vista-dazn-artificial-intelligence-data.

[91] James, R. Scouting Enters Brave New World as Clubs Step Up Search for Talent. *The Guardian* (Dec 2012). https://www.theguardian.com/football/2012/dec/13/scouting-transfer-window-new-stars.

[92] Jaspers, A., De Beéck, T. O., Brink, M. S., Frencken, W. G., Staes, F., Davis, J. J., and Helsen, W. F. Relationships Between the External and Internal Training Load in Professional Soccer: What can we Learn from Machine Learning? *International Journal of Sports Physiology and Performance 13*, 5 (2018), 625–630.

[93] Kharrat, T., McHale, I. G., and Peña, J. L. Plus–Minus Player Ratings for Soccer. *European Journal of Operational Research 283*, 2 (2020), 726–736.

[94] Kim, W., Moon, S.-W., Lee, J., Nam, D.-W., and Jung, C. Multiple Player Tracking in Soccer Videos: An Adaptive Multiscale Sampling Approach. *Multimedia Systems 24*, 6 (2018), 611–623.

[95] Kleebauer, A. Everton's Idrissa Gueye is the New N'Golo Kante - and Here are the Stats to Prove it. Liverpool Echo, 2017. https://www.liverpoolecho.co.uk/sport/football/football-news/evertons-idrissa-gueye-new-ngolo-12965076.

[96] Knauf, K., and Brefeld, U. Spatio-Temporal Convolution Kernels for Clustering Trajectories. In *Proceedings of the Workshop on Large-Scale Sports Analytics* (2014).

[97] Knauf, K., Memmert, D., and Brefeld, U. Spatio-Temporal Convolution Kernels. *Machine Learning 102*, 2 (2016), 247–273.

[98] Knutson, T. Introducing xGChain. `http://www.statsbombservices.com/introducing-xgchain`, 2017.

[99] Koren, Y., Bell, R., and Volinsky, C. Matrix Factorization Techniques for Recommender Systems. *Computer 42*, 8 (2009), 30–37.

[100] Kullowatz, M. Goals Added: Deep Dive Methodology. American Soccer Analysis, 2020. `https://www.americansocceranalysis.com/home/2020/5/4/goals-added-deep-dive-methodology`.

[101] Le, H. M., Carr, P., Yue, Y., and Lucey, P. Data-Driven Ghosting using Deep Imitation Learning. In *MIT Sloan Sports Analytics Conference* (2017).

[102] Lee, D. D., and Seung, H. S. Learning the Parts of Objects by Non-Negative Matrix Factorization. *Nature 401*, 6755 (1999), 788–791.

[103] Lewis, M. *Moneyball: The Art of Winning an Unfair Game*. W. W. Norton & Company, 2004.

[104] Link, D., Lang, S., and Seidenschwarz, P. Real Time Quantification of Dangerousity in Football using Spatiotemporal Tracking Data. *PloS one 11*, 12 (2016).

[105] Littman, M. Markov Games as a Framework for Multi-Agent Reinforcement Learning. In *Proceedings of the International Conference on Machine Learning* (1994).

[106] Liu, G., Luo, Y., Schulte, O., and Kharrat, T. Deep soccer analytics: learning an action-value function for evaluating soccer players. *Data Mining and Knowledge Discovery* (2020), 1–29.

[107] Liu, G., and Schulte, O. Deep Reinforcement Learning in Ice Hockey for Context-Aware Player Evaluation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence* (2018), pp. 3442–3448.

[108] Lucey, P., Bialkowski, A., Monfort, M., Carr, P., and Matthews, I. Quality vs. Quantity: Improved Shot Prediction in Soccer Using Strategic Features from Spatiotemporal Data. In *MIT Sloan Sports Analytics Conference* (2014).

[109] Lucey, P., Oliver, D., Carr, P., Roth, J., and Matthews, I. Assessing Team strategy using Spatiotemporal Data. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining* (2013), ACM, pp. 1366–1374.

[110] Mackay, N. Predicting Goal Probabilities for Possessions in Football. Master's thesis, Vrije Universiteit Amsterdam, 2017.

[111] Maher, M. J. Modelling Association Football Scores. *Statistica Neerlandica 36*, 3 (1982), 109–118.

[112] Mardia, K. V., and Jupp, P. E. *Directional Statistics*, vol. 494. John Wiley & Sons, 2009.

[113] McGuiness, R. Spectators who are on the Ball: A Look Inside the World of Sports Stats. *Metro* (Jan 2012). https://metro.co.uk/2012/01/03/spectators-who-are-on-the-ball-a-look-inside-the-world-of-sports-statistics-272423/.

[114] McHale, I., and Scarf, P. Modelling the Dependence of Goals Scored by Opposing Teams in International Soccer Matches. *Statistical Modelling 11*, 3 (2011), 219–236.

[115] McHale, I. G., Scarf, P. A., and Folker, D. E. On the Development of a Soccer Player Performance Rating system for the English Premier League. *Interfaces 42*, 4 (2012), 339–351.

[116] McHale, I. G., and Szczepański, Ł. A Mixed Effects Model for Identifying Goal Scoring Ability of Footballers. *Journal of the Royal Statistical Society: Series A (Statistics in Society) 177*, 2 (2014), 397–417.

[117] McLachlan, G. J., and Basford, K. E. *Mixture Models: Inference and Applications to Clustering*, vol. 38. M. Dekker New York, 1988.

[118] Meert, W., Hendrickx, K., Van Craenendonck, T., et al. wannesm/dtaidistance v1.2.2. *Github* (Jul 2019). https://github.com/wannesm/dtaidistance.

[119] Memmert, D., Lemmink, K. A., and Sampaio, J. Current Approaches to Tactical Performance Analyses in Soccer using Position Data. *Sports Medicine 47*, 1 (2017), 1–10.

[120] Metrica Sports. https://metrica-sports.com. Accessed: 2020-05-03.

[121] Michalczyk, K. An Attempt to Extend xG Gain. https://kubamichalczyk.github.io//2018/11/18/An-attempt-to-extend-xG-gain.html, 2018. Accessed: 2019-06-19.

[122] Morris, B. Lionel Messi Is Impossible. FiveThirtyEight, 2014. http://fivethirtyeight.com/features/lionel-messi-is-impossible.

[123] Müller, M. *Dynamic Time Warping.* Information Retrieval for Music and Motion. Springer, 2007, ch. 4, pp. 69–84.

[124] Nori, H., Jenkins, S., Koch, P., and Caruana, R. InterpretML: A Unified Framework for Machine Learning Interpretability. *arXiv preprint arXiv:1909.09223* (2019).

[125] Nørstebø, O., Bjertnes, V. R., and Vabo, E. Valuing Individual Player Involvements in Norwegian Association Football. Master's thesis, Norwegian University of Science and Technology, 2016.

[126] Opta Sports. http://www.optasports.com. Accessed: 2017-02-13.

[127] Pappalardo, L., Cintia, P., Ferragina, P., Massucco, E., Pedreschi, D., and Giannotti, F. PlayeRank: Data-Driven Performance Evaluation and Player Ranking in Soccer via a Machine Learning Approach. *ACM Trans. Intell. Syst. Technol. 10*, 5 (Sept. 2019).

[128] Pappalardo, L., Cintia, P., Rossi, A., Massucco, E., Ferragina, P., Pedreschi, D., and Giannotti, F. A Public Data Set of Spatio-Temporal Match Events in Soccer Competitions. *Scientific data 6*, 1 (2019), 1–15.

[129] Pedregosa, F., Varoquaux, G., et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research 12*, Oct (2011), 2825–2830.

[130] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[131] Pei, J., Han, J., Mortazavi-Asl, B., Wang, J., Pinto, H., Chen, Q., Dayal, U., and Hsu, M.-C. Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach. *IEEE Transactions on knowledge and data engineering 16*, 11 (2004), 1424–1440.

[132] Pena, J. L. A Markovian Model for Association Football Possession and its Outcomes. *arXiv preprint arXiv:1403.7993* (2014).

[133] Pettigrew, S. Assessing the Offensive Productivity of NHL Players using In-Game Win Probabilities. In *MIT Sloan Sports Analytics Conference* (2015).

[134] Pierce, J. Henderson: "I'm Learning Fast in the New Midfield Role Klopp's Given Me". Liverpool Echo, 2016. https://www.liverpoolecho.co.uk/sport/football/football-news/henderson-im-learning-fast-new-11862193.

[135] Polar. http://www.polar.com. Accessed: 2020-06-27.

[136] Pollard, R., Benjamin, B., and Reep, C. Sport and the Negative Binomial Distribution. *Optimal Strategies in Sports* (1977), 188–195.

[137] Power, P., Ruiz, H., Wei, X., and Lucey, P. Not All Passes Are Created Equal: Objectively Measuring the Risk and Reward of Passes in Soccer from Tracking Data. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2017), pp. 1605–1613.

[138] Prenderville, L. Sergio Ramos Identifies Aymeric Laporte and Matthijs de Ligt as His Long-Term Replacements at Real Madrid. The Mirror, 2017. https://www.mirror.co.uk/sport/football/transfer-news/sergio-ramos-identifies-aymeric-laporte-11710624.

[139] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A., and Gulin, A. CatBoost: Unbiased Boosting with Categorical Features. In *Advances in Neural Information Processing Systems* (2018), pp. 6639–6649.

[140] Rendle, S. Factorization Machines. In *2010 IEEE International Conference on Data Mining* (2010), IEEE, pp. 995–1000.

[141] Reynolds, D. A. Gaussian Mixture Models. *Encyclopedia of Biometrics 741* (2009).

[142] Robberechts, P., and Davis, J. How Data Availability Affects the Ability to Learn Good xG Models. In *ECML/PKDD 2020 Workshop on Machine Learning and Data Mining for Sports Analytics* (2020).

[143] Robberechts, P., and Davis, J. Illustrating the Interplay between Features and Models in xG. DTAI Sports Analytics Lab, 2020. https://dtai.cs.kuleuven.be/sports/blog/illustrating-the-interplay-between-features-and-models-in-xg.

[144] Robberechts, P., Van Haaren, J., and Davis, J. Who Will Win It? An In-Game Win Probability Model for Football. *arXiv preprint arXiv:1906.05029* (2019).

[145] Romero, A. Cristiano Ronaldo: The Change to a 'Number 9', 2016. https://en.as.com/en/2016/12/19/opinion/1482164003_264275.html.

[146] Routley, K., and Schulte, O. A Markov Game Model for Valuing Player Actions in Ice Hockey. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence* (2015), pp. 782–791.

[147] Rudd, S. A Framework for Tactical Analysis and Individual Offensive Production Assessment in Soccer Using Markov Chains. In *New England Symposium on Statistics in Sports* (2011).

[148] RunEASI. http://www.runeasi.ai. Accessed: 2020-06-27.

[149] Sæbø, O. D., and Hvattum, L. M. Evaluating the Efficiency of the Association Football Transfer Market using Regression Based Player Ratings. In *Norsk IKT-konferanse for forskning og utdanning* (2015).

[150] Sammut, C., and Webb, G. I. *Encyclopedia of Machine Learning*. Springer Science & Business Media, 2011.

[151] Schuckers, M., and Curro, J. Total Hockey Rating (THoR): A Comprehensive Statistical Rating of National Hockey League Forwards and Defensemen Based Upon All On-Ice Events. In *7th Annual MIT Sloan Sports Analytics Conference* (2013).

[152] SciSports. http://www.scisports.com. Accessed: 2020-06-27.

[153] SciSports. SciSkill Index: Why and How. https://www.scisports.com/sciskill-index-why-and-how/, 2016. Accessed: 2020-06-30.

[154] Second Spectrum. http://www.secondspectrum.com. Accessed: 2020-06-27.

[155] Shapiro, L., and Stockman, G. C. Computer Vision. *Prentence Hall* (2001).

[156] Sharma, R. How Cristiano Ronaldo Transformed from a Winger into a Deadly No 9... and Why He Could Really Play for Real Madrid into his 40s. Daily Mail, 2017. http://www.dailymail.co.uk/sport/football/article-4469198/How-Ronaldo-transformed-winger-deadly-No9.html.

[157] Shitrit, H. B., Berclaz, J., Fleuret, F., and Fua, P. Tracking Multiple People under Global Appearance Constraints. In *2011 International Conference on Computer Vision* (2011), IEEE, pp. 137–144.

[158] Sidiropoulos, N. D., De Lathauwer, L., Fu, X., Huang, K., Papalexakis, E. E., and Faloutsos, C. Tensor Decomposition for Signal Processing and Machine Learning. *IEEE Transactions on Signal Processing 65*, 13 (2017), 3551–3582.

[159] Singh, K. Introducing Expected Threat (xT), 2019. Accessed: 2019-10-12.

[160] Smith, R. Is Paulo Dybala the Next Lionel Messi? "He Can Go as High as He Likes". The New York Times, 2017. https://www.nytimes.com/2017/04/10/sports/soccer/paulo-dybala-juventus-lionel-messi-barcelona.html.

[161] Spearman, W. Beyond Expected Goals. In *Proceedings of the 12th MIT Sloan Sports Analytics Conference* (2018), pp. 1–17.

[162] Spearman, W., Basye, A., Dick, G., Hotovy, R., and Pop, P. Physics-Based Modeling of Pass Probabilities in Soccer. In *Proceeding of the 11th MIT Sloan Sports Analytics Conference* (2017).

[163] Srikant, R., and Agrawal, R. Mining Sequential Patterns: Generalizations and Performance Improvements. In *International Conference on Extending Database Technology* (1996), Springer, pp. 1–17.

[164] STATS Perform. http://www.statsperform.com, 2020. Accessed: 2020-06-27.

[165] StatsBomb. http://www.statsbomb.com. Accessed: 2020-05-03.

[166] STATSports. https://statsports.com/. Accessed: 2020-06-08.

[167] Stracuzzi, D. J., Fern, A., Ali, K., Hess, R., Pinto, J., Li, N., Konik, T., and Shapiro, D. G. An Application of Transfer to American Football: From Observation of Raw Video to Control in a Simulated Environment. *AI Magazine 32*, 2 (2011), 107–125.

[168] Szczepański, Ł., and McHale, I. Beyond Completion Rate: Evaluating the Passing Ability of Footballers. *Journal of the Royal Statistical Society: Series A (Statistics in Society) 179*, 2 (2016), 513–533.

[169] Taki, T., and Hasegawa, J.-i. Visualization of Dominant Region in Team Games and its Application to Teamwork Analysis. In *Proceedings computer graphics international 2000* (2000), IEEE, pp. 227–235.

[170] Tango, T., Lichtman, M., and Dolphin, A. *The Book: Playing the Percentages in Baseball*. Potomac Books, Inc., 2007.

[171] Tiedemann, T., Francksen, T., and Latacz-Lohmann, U. Assessing the Performance of German Bundesliga Football Players: A Non-Parametric Metafrontier Approach. *Central European Journal of Operations Research 19*, 4 (2011), 571–587.

[172] Van Gool, J., Van Haaren, J., and Davis, J. The Automatic Analysis of the Playing Style of Soccer Teams. *KU Leuven* (2015).

[173] Van Haaren, J., Ben Shitrit, H., Davis, J., and Fua, P. Analyzing Volleyball Match Data from the 2014 World Championships using Machine Learning Techniques. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016), ACM, pp. 627–634.

[174] Van Haaren, J., Dzyuba, V., Hannosset, S., and Davis, J. Automatically Discovering Offensive Patterns in Soccer Match Data. In *International Symposium on Intelligent Data Analysis* (Oct. 2015), E. Fromont, T. De Bie, and M. van Leeuwen, Eds., vol. 9385 of *Lecture Notes in Computer Science*, Springer, pp. 286–297.

[175] Van Haaren, J., Hannosset, S., and Davis, J. Strategy Discovery in Professional Soccer Match Data. In *Proceedings of the KDD-16 Workshop on Large-Scale Sports Analytics* (2016), pp. 1–4.

[176] Van Haaren, J., Robberechts, P., Decroos, T., Bransen, L., and Davis, J. Analysing Performance and Playing Style using Ball Event Data. *Football Analytics: Now and Beyond. A Deep Dive into the Current State of Advanced Data Analytics.* (2019), 36–47.

[177] Van Leeuwen, M. Interactive Data Exploration using Pattern Mining. In *Interactive Knowledge Discovery and Data Mining in Biomedical Informatics*. Springer, 2014, pp. 169–182.

[178] Van Roy, M., Robberechts, P., Decroos, T., and Davis, J. Valuing On-the-Ball Actions in Soccer: A Critical Comparison of xT and VAEP. In *AAAI 2020 Workshop on Artificial Intelligence in Team Sports* (2020).

[179] Verstraete, K., Decroos, T., Coussement, B., Vannieuwenhoven, N., and Davis, J. Analyzing Soccer Players' Skill Ratings Over Time Using Tensor-Based Methods. In *Proceedings of the 6th International Workshop on Machine Learning and Data Mining for Sports Analytics at ECML/PKDD 2019* (2019), Springer.

[180] Vidal, E., Casacuberta, F., Benedi, J. M., Lloret, M. J., and Rulot, H. On the Verification of Triangle Inequality by Dynamic Time Warping Dissimilarity Measures. *Speech Communication 7*, 1 (1988), 67–79.

[181] Vroonen, R., Decroos, T., Van Haaren, J., and Davis, J. Predicting the Potential of Professional Soccer Players. In *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2017 workshop* (2017), vol. 1971, pp. 1–10.

[182] Wang, Q., Zhu, H., Hu, W., Shen, Z., and Yao, Y. Discerning Tactical Patterns for Professional Soccer Teams: An enhanced Topic Model with Applications. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2015), pp. 2197–2206.

[183] Wei, X., Sha, L., Lucey, P., Morgan, S., and Sridharan, S. Large-Scale Analysis of Formations in Soccer. In *Digital Image Computing: Techniques and Applications (DICTA), 2013 International Conference on* (2013), IEEE, pp. 1–8.

[184] Williams, G. Jordan Henderson is Relishing His New role in the Liverpool Midfield. Liverpool Echo, 2016. `https://www.liverpoolecho.co.uk/sport/football/football-news/liverpool-jordan-henderson-jurgen-klopp-12123785`.

[185] Wyscout. `http://www.wyscout.com`. Accessed: 2020-05-03.

[186] Yam, D. Attacking Contributions: Markov Models for Football. StatsBomb, 2019. `https://statsbomb.com/2019/02/attacking-contributions-markov-models-for-football/`.

[187] Yianilos, P. N. Data Structures and Algorithms for Nearest Neighbor Search in General Metric Spaces. In *SODA '93: Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms* (1993), pp. 311–321.

[188] Zaki, M. J. SPADE: An Efficient Algorithm for Mining Frequent Sequences. *Machine Learning 42*, 1-2 (2001), 31–60.

Curriculum Vitae

Tom Decroos was born on 20 February 1993. He obtained his Bachelor of Science in Informatics at KU Leuven Kulak in 2014 and his Master of Engineering in Computer Science (option Artificial Intelligence) at KU Leuven in 2016.

Tom started his doctoral studies at KU Leuven in September 2016 in the DTAI research group under the supervision of Prof. Dr. Jesse Davis. He received a personal 4-year PhD fellowship from the Research Foundation Flanders (FWO). In 2018, he did a 3-month internship at Facebook where he worked as a software engineer in machine learning. Tom has published papers at top-tier data science conferences such as AAAI, KDD and ECML/PKDD, and received the best applied data science paper award at the 2019 SIGKDD Conference on Knowledge Discovery and Data Mining. Additionally, his research has been popularized in various Belgian and International Media (Sporza, de Standaard, Karrewiet, the Mirror, the Daily Star, ESPN, Diario AS, etc.)

List of publications

Peer-reviewed Conference Papers

Decroos, T., De Causmaecker, P., and Demoen, B. Solving Euclidean Steiner Tree Problems with Multi Swarm Optimization. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation* (2015), ACM; New York, NY, USA, pp. 1379–1380

Decroos, T., Dzyuba, V., Van Haaren, J., and Davis, J. Predicting Soccer Highlights from Spatio-Temporal Match Event Streams. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence* (2017), pp. 1302–1308

Decroos, T., Van Haaren, J., and Davis, J. Automatic Discovery of Tactics in Spatio-Temporal Soccer Match Data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2018), pp. 223–232

Decroos, T., Schütte, K., De Beéck, T. O., Vanwanseele, B., and Davis, J. AMIE: Automatic Monitoring of Indoor Exercises. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (2018), Springer, pp. 424–439

Decroos, T., Bransen, L., Van Haaren, J., and Davis, J. Actions Speak Louder than Goals: Valuing Player Actions in Soccer. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2019), KDD '19, ACM, pp. 1851–1861
★ *Best paper in the applied data science track.*

Decroos, T., and Davis, J. Player Vectors: Characterizing Soccer Players' Playing Style from Match Event Streams. In *Joint European Conference on*

Machine Learning and Knowledge Discovery in Databases (2019), Springer

Davis, J., Bransen, L., Decroos, T., Robberechts, P., and Van Haaren, J. Assessing the Performances of Soccer Players. In *Proceedings of the 12th International Symposium on Computer Science in Sport (IACSS 2019)* (Cham, 2019), M. Lames, A. Danilov, E. Timme, and Y. Vassilevski, Eds., Springer International Publishing, pp. 3–10

Decroos, T., Van Roy, M., and Davis, J. SoccerMix: Representing Soccer Actions with Mixture Models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (2020), Springer

Decroos, T., Bransen, L., Van Haaren, J., and Davis, J. VAEP: An Objective Approach to Valuing On-the-Ball Actions in Soccer (Extended Abstract). In *Proceedings of the 29th International Joint Conference on Artificial Intelligence* (2020), AAAI Press

Book Chapters

Van Haaren, J., Robberechts, P., Decroos, T., Bransen, L., and Davis, J. Analysing Performance and Playing Style using Ball Event Data. *Football Analytics: Now and Beyond. A Deep Dive into the Current State of Advanced Data Analytics.* (2019), 36–47

Peer-reviewed Workshop Papers

Decroos, T., Van Haaren, J., Dzyuba, V., and Davis, J. STARSS: A Spatio-Temporal Action Rating System for Soccer. In *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2017 workshop* (2017)

Vroonen, R., Decroos, T., Van Haaren, J., and Davis, J. Predicting the Potential of Professional Soccer Players. In *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2017 workshop* (2017), vol. 1971, pp. 1–10

Geerts, A., Decroos, T., and Davis, J. Characterizing Soccer Players' Playing Style from Match Event Streams. In *Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2018 workshop* (2018), vol. 2284, Springer, pp. 115–126

Verstraete, K., Decroos, T., Coussement, B., Vannieuwenhoven,

N., AND DAVIS, J. Analyzing Soccer Players' Skill Ratings Over Time Using Tensor-Based Methods. In *Proceedings of the 6th International Workshop on Machine Learning and Data Mining for Sports Analytics at ECML/PKDD 2019* (2019), Springer

DECROOS, T., AND DAVIS, J. Interpretable Prediction of Goals in Soccer. In *AAAI 2020 Workshop on Artificial Intelligence in Team Sports* (2020)

VAN ROY, M., ROBBERECHTS, P., DECROOS, T., AND DAVIS, J. Valuing On-the-Ball Actions in Soccer: A Critical Comparison of xT and VAEP. In *AAAI 2020 Workshop on Artificial Intelligence in Team Sports* (2020)

FACULTY OF ENGINEERING SCIENCE
DEPARTMENT OF COMPUTER SCIENCE
DECLARATIVE LANGUAGES AND ARTIFICIAL INTELLIGENCE (DTAI)
Celestijnenlaan 200A box 2402
3001 Heverlee
tom.decroos@{cs.kuleuven.be, gmail.com}
https://tomdecroos.github.io