



Latest updates: <https://dl.acm.org/doi/10.1145/3292500.3330758>

RESEARCH-ARTICLE

Actions Speak Louder than Goals: Valuing Player Actions in Soccer

TOM DECROOS, KU Leuven, Leuven, Vlaams-Brabant, Belgium

LOTTE BRANSEN

JAN VAN HAAREN

JESSE JON DAVIS, KU Leuven, Leuven, Vlaams-Brabant, Belgium

Open Access Support provided by:

KU Leuven



PDF Download
3292500.3330758.pdf
29 January 2026
Total Citations: 163
Total Downloads: 4033

Published: 25 July 2019

Citation in BibTeX format

KDD '19: The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining
August 4 - 8, 2019
AK, Anchorage, USA

Conference Sponsors:
SIGMOD
SIGKDD

Actions Speak Louder than Goals: Valuing Player Actions in Soccer

Tom Decroos

KU Leuven

Leuven, Belgium

tom.decroos@cs.kuleuven.be

Jan Van Haaren

SciSports

Amersfoort, Netherlands

j.vanhaaren@scisports.com

Lotte Bransen

SciSports

Amersfoort, Netherlands

l.bransen@scisports.com

Jesse Davis

KU Leuven

Leuven, Belgium

jesse.davis@cs.kuleuven.be

ABSTRACT

Assessing the impact of the individual actions performed by soccer players during games is a crucial aspect of the player recruitment process. Unfortunately, most traditional metrics fall short in addressing this task as they either focus on rare actions like shots and goals alone or fail to account for the context in which the actions occurred. This paper introduces (1) a new language for describing individual player actions on the pitch and (2) 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. By aggregating soccer players' action values, their total offensive and defensive contributions to their team can be quantified. We show how our approach considers relevant contextual information that traditional player evaluation metrics ignore and present a number of use cases related to scouting and playing style characterization in the 2016/2017 and 2017/2018 seasons in Europe's top competitions.

CCS CONCEPTS

- **Information systems** → **Data mining; Data stream mining;**
- **Computing methodologies** → **Machine learning; Artificial intelligence; Supervised learning.**

KEYWORDS

sports analytics; event stream data; soccer match data; valuing actions; probabilistic classification

ACM Reference Format:

Tom Decroos, Lotte Bransen, Jan Van Haaren, and Jesse Davis. 2019. Actions Speak Louder than Goals: Valuing Player Actions in Soccer. In *The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '19)*, August 4–8, 2019, Anchorage, AK, USA. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3292500.3330758>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

KDD '19, August 4–8, 2019, Anchorage, AK, USA

© 2019 Association for Computing Machinery.

ACM ISBN 978-1-4503-6201-6/19/08...\$15.00

<https://doi.org/10.1145/3292500.3330758>

1 INTRODUCTION

How will a soccer player's actions impact his or her team's performances in games? This question is relevant for a variety of tasks within a soccer club such as player acquisition, player evaluation, and scouting. It is also important for the media and building fan engagement, as fans like nothing better than comparing players and arguing why their favorite player is better than the others.

Nevertheless, the task of objectively quantifying the impact of the individual actions performed by soccer players during games remains largely unexplored to date. What complicates the task is the low-scoring and dynamic nature of soccer games. While most actions do not impact the scoreline directly, they often do have important longer-term effects. For example, a long pass from one flank to the other may not immediately lead to a goal but can open up space to set up a goal chance several actions down the line.

Most existing approaches for valuing actions in soccer suffer from three important limitations. First, these approaches largely ignore actions other than goals and shots as most work to date has focused on the concept of the expected value of a goal attempt [1, 5, 20, 21]. Second, existing approaches tend to assign a fixed value to each action, regardless of the circumstances under which the action was performed. For example, many pass-based metrics treat passes between defenders in the defensive third of the pitch without any pressure whatsoever and passes between attackers in the offensive third under heavy pressure from the opponents similarly. Third, most approaches only consider immediate effects and fail to account for an action's effects a bit further down the line.

To help fill the gap in objectively quantifying player performances, this paper proposes a novel data-driven framework for valuing actions in a soccer game. Unlike most existing work, it considers all types of actions (e.g., passes, crosses, dribbles, take-ons, and shots) 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. Our approach fits within the growing line of data science research for analyzing sports data (e.g., [9, 19, 24, 27]).

In summary, this paper makes the following five contributions:

- (1) a language for representing player actions;

- (2) a framework for valuing player actions and rating players based on their impact on the game;
- (3) a model for predicting short-term scoring and conceding probabilities at any moment in a game;
- (4) a number of use cases showcasing our most interesting results and insights;
- (5) a Python package¹ 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.

2 SPADL: A LANGUAGE FOR DESCRIBING PLAYER ACTIONS

Two primary data sources about soccer games exist that can be used to value actions: (1) event stream data and (2) optical tracking data. Event stream data annotates the times and locations of specific events (e.g., passes, shots, and cards) that occur in a game. Optical tracking data records the locations of the players and the ball at a high frequency using optical tracking systems during games. Multiple different companies (e.g., Opta, Wyscout, STATS, Second Spectrum, SciSports, and StatsBomb) generate one or both of these types of data. Due to the high cost of optical tracking systems, tracking data is only available in wealthy leagues or clubs, while event stream data is more widely and cheaply available. Moreover, optical tracking data is usually not shared across leagues. Hence, this paper focuses exclusively on event stream data. However, the contributions in this paper could also be applied to full tracking data with some minor extensions. A key challenge from a data science perspective is that the nature of the event stream data complicates analysis. We first describe the data science challenges posed by currently available event stream data and then show how our proposed SPADL language addresses these challenges.

2.1 Five data science challenges posed by current event stream data

The first challenge is that the event stream data serves multiple different objectives (e.g., reporting information to broadcasters, newspapers, or soccer clubs), which means that the data is not necessarily designed to facilitate data analysis. Some important information can be missing (e.g., Wyscout does not record exact end locations for shots). Some of the recorded information might be irrelevant for data analysis and may actually hinder it by increasing the complexity of the preprocessing steps (e.g., Wyscout records duels between two players as two separate events).

The second challenge is that each vendor of event stream data uses their own unique terminology and definitions to describe the events that occur during a game. Hence, software written to analyze data has to be tailored to a specific vendor and cannot be used without modifications to analyze data from another vendor.

The third challenge is that vendors' current event stream formats typically remain backward compatible with their previous formats. Some vendors have been providing data for over a decade, and are unable to alter initial suboptimal design choices. Moreover, what the vendors annotate has evolved and now includes additional

events and more detailed information. For example, Opta has four different event types for a shot, depending on its result, which makes it extremely cumbersome to query shot characteristics.

The fourth challenge is that most vendors offer optional information snippets per type of event. For example, for fouls, Opta often specifies more details on the exact type of foul that was committed. While sometimes useful, this dynamic information makes it extremely hard to apply automatic analysis tools.

The final challenge is that most machine learning algorithms require fixed-length feature vectors and cannot handle variable-sized vectors arising from, e.g., the sporadic presence of optional information snippets. Therefore, analysts usually have to write a complicated event preprocessor that extracts the features relevant to their analysis. Developing these preprocessors requires an excessive amount of programming effort and intimate knowledge of the event stream format, but the end result is a one-off script tailored to one specific vendor's current event stream format.

2.2 Language description

Based on domain knowledge and feedback from soccer experts, we propose SPADL (Soccer Player Action Description Language) as an attempt to unify the existing event stream formats into a common vocabulary that enables subsequent data analysis. It is designed to be *human-interpretable*, *simple* and *complete* to accurately define and describe actions on the pitch. The human-interpretability allows reasoning about what happens on the pitch and verifying whether the action values correspond to soccer experts' intuitions. The simplicity reduces the chance of making mistakes when automatically processing the language. 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 formats and to benefit the data science community, we release a Python package that automatically converts event streams to SPADL. Our package currently supports event streams provided by 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 $[a_1, a_2, \dots, a_m]$, where m is the total number of actions that happened in the game. Each action is a tuple of nine attributes:

- StartTime:** the action's start time,
- EndTime:** the action's end time,
- StartLoc:** the (x, y) location where the action started,
- EndLoc:** 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 formats, we always store the same nine attributes for each action. Excluding optional information snippets enables us to more easily apply automatic analysis tools.

¹<https://github.com/ML-KULeuven/socceraction>

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 is in Appendix A.1.

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 VAEП: A FRAMEWORK FOR VALUING PLAYER ACTIONS

This section introduces the VAEP (Valuing Actions by Estimating Probabilities) 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.

3.1 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.

Suppose that for each game state $S_i = [a_1, \dots, a_i]$, 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:

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

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 can be computed as:

$$\Delta P_{concedes}(a_i, x) = P_{concedes}(S_i, x) - P_{concedes}(S_{i-1}, x). \quad (2)$$

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 1 and 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.*

$$V(a_i, x) = \Delta P_{scores}(a_i, x) + (-\Delta P_{concedes}(a_i, x)) \quad (3)$$

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.

3.2 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 rating as

$$rating(p) = \frac{90}{m} \sum_{a_i \in A_p^T} V(a_i), \quad (4)$$

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.

4 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 $S_i = [a_1, \dots, a_i]$;

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 $F_i^k = [a_{i+1}, \dots, a_{i+k}]$ 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) $P_{concedes}(S_i, x_i)$.

For both binary classification problems we train a probabilistic classifier to estimate the probabilities. In principle, any machine learning model (e.g., Logistic Regression, Random Forest, or Neural Network) that predicts a probability could be used to address these tasks. However, an important criteria is that the probability estimates should be well-calibrated [22]. We use CatBoost [26] and justify this selection empirically in Section 5.6.3.

Applying a standard machine learning algorithm requires converting the sequence of actions $[a_1, a_2, \dots, a_m]$ describing an entire game into examples in the feature-vector format. Thus, one training example is constructed for each game state S_i . We now describe how we compute the labels and features for each game state.

4.1 Constructing 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 paper, we chose $k = 10$ based on domain knowledge and preliminary experiments.

4.2 Constructing features

For each example, instead of defining features based on the entire current game state $S_i = [a_1, \dots, a_i]$, we only consider the previous 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. Based on the SPADL representation, we consider three categories of features.

1. 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.

2. 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.

3. 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 EXPERIMENTS

Evaluating our 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.

We focus our analysis on Wyscout data 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.

We train two classification models using the CatBoost algorithm and the feature set detailed in Section 4 to produce scoring and conceding probabilities, action values, and player ratings. We train the first model on the 2012/2013 through 2015/2016 seasons to produce the outcomes for the 2016/2017 season. Similarly, we train the second model on the 2012/2013 through 2016/2017 seasons to produce the outcomes for the 2017/2018 season.

5.1 Intuition behind the action values

Figure 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.

	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

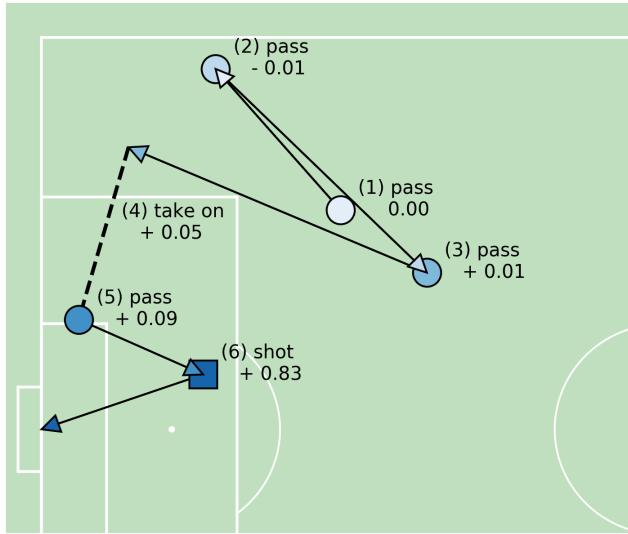


Figure 1: The attack leading up to Barcelona’s final goal in their 3-0 win against Real Madrid on December 23, 2017.

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.

5.2 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 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. Furthermore, the ranking in terms of goals + assists per 90 minutes aims 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 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).

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

5.3 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 2017/2018 season in the English and Spanish leagues (Table 2a), and the French, Dutch and Belgian leagues (Table 2b).

Marcus Rashford, who was linked with a € 110 million move to Real Madrid in January 2019,³ and Ousmane Dembélé, who moved to Barcelona in August 2017 for a fee of € 120 million, are the most notable players in Table 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 current estimated market value of € 5 million.

²<https://www.squawka.com/en/news/every-player-with-10-goals-and-10-assists-in-europe-s-top-five-leagues-this-season-ranked-by-contribution-per-90/1031863>

³<https://www.thesun.co.uk/sport/football/8318008/real-madrid-marcus-rashford-transfer-man-utd/>

Table 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 1, 2019 according to Transfermarkt.de.

(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 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

David Neres tops Table 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 is now a transfer target for top clubs Liverpool, Chelsea, and Arsenal, who all wish 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 2a and 2b demonstrate our framework’s ability to serve as a useful tool for talent scouts. Our framework can generate rankings for each league in the world (e.g., second divisions or leagues in North America, South America, and Asia) given that the required event stream data is available.

5.4 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 2a compares Dembélé, Coutinho 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

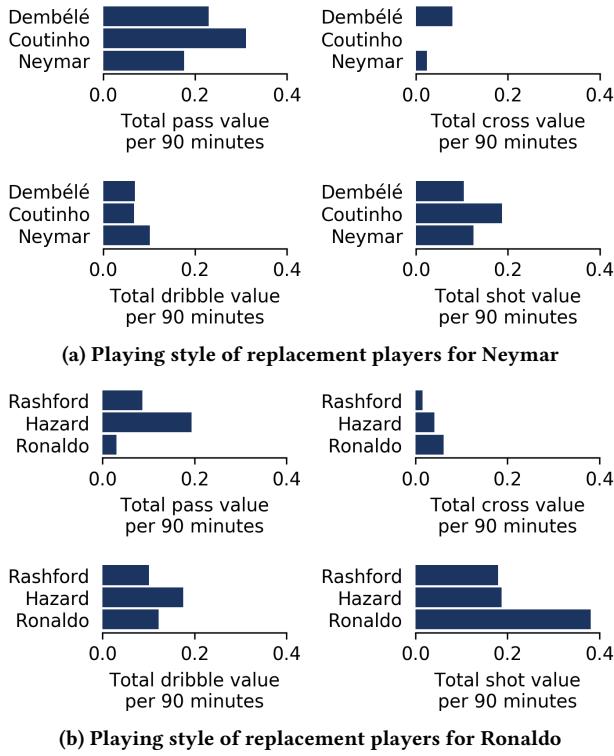


Figure 2: Overview of the total contribution per 90 minutes for different types of actions for (a) Neymar, Ousmane Dembélé, and Philippe Coutinho during the 2016/2017 season, and (b) Cristiano Ronaldo, Marcus Rashford, and Eden Hazard during the 2017/2018 season.

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. The struggling club appears in desperate need of a suitable replacement. Manchester United's Marcus Rashford and Chelsea's Eden Hazard have both been linked with moves to Madrid. However, Figure 2b shows that neither comes 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. If Real Madrid want to stick to their current playing style, our analysis suggests that the 21-year-old Rashford would be the better choice. However, if their aim is to immediately strengthen their team, then the 28-year-old Hazard would be the preferred choice as he is a better player regardless of his specific playing style.

5.5 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 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 2017/2018 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,⁴ Messi is clearly in a class of his own.

Zooming in on Figure 3a, Figure 3b shows the top-10 players in the 2017/2018 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 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. Lionel Messi is an outlier in the sense that he rates high on action quality and quantity at the same time.

5.6 Evaluating design choices

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 and conceding probabilities for which ground truth labels are available.

5.6.1 Evaluation methodology. To generate scoring and conceding probabilities, we train classification models using the features described in Section 4 with the CatBoost algorithm. We evaluate these design choices by comparing the performance of our approach to alternative approaches that use either a different feature set or a different algorithm.

Similar to our main approach, we train two classification models for each alternative: we train a first model on the 2012/2013 through 2015/2016 seasons to produce the outcomes for the 2016/2017 season, and a second model on the 2012/2013 through 2016/2017 seasons to produce the outcomes for the 2017/2018 season.

⁴<https://fivethirtyeight.com/features/lionel-messi-is-impossible/>

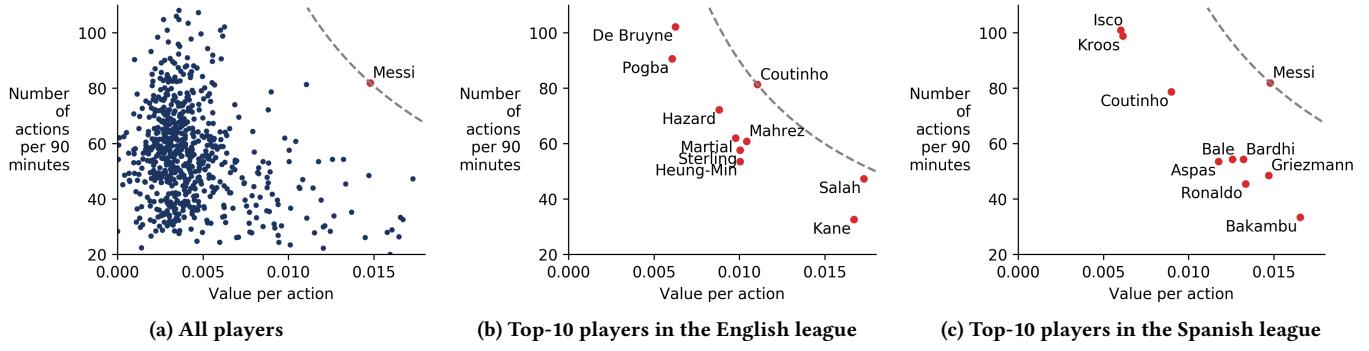


Figure 3: Scatter plots of players in the 2017/2018 season who played at least 900 minutes in the Spanish or English league. The plots contrast the average number of actions performed per 90 minutes with the average value of the actions of the player. As shown by the grey-dotted isoline in (a) and (c), Lionel Messi is clearly in a class of his own.

We evaluate the performance of each approach using two metrics often used for evaluating probabilistic predictions: the Brier score and ROC AUC [12]. The Brier score measures the accuracy and calibration of the predictions and is minimized when the true underlying probability distribution of the data is reported. This property is important because we sum and subtract the predicted probabilities to generate action values. The area under the receiver operator curve (ROC AUC) evaluates how well the approaches can discern positive examples from negative examples. An important advantage of ROC AUC is that the metric is unaffected by unbalanced data sets, as in our data only 1.5% (0.5%) of all game states lead to a scored (conceded) goal.

5.6.2 Choice of feature set. Most existing models that analyze soccer event data only use location and action type data [1, 8]. To evaluate the usefulness of our comprehensive feature set (detailed in Section 4), we compare it to four baseline feature sets: no features,⁵ location, action type, and location + action type (Table 3). We evaluate each feature set using the CatBoost algorithm. For estimating both P_{scores} and $P_{concedes}$, our feature set outperforms the baseline feature sets on both evaluation metrics. This result suggests that our features capture important game state context that is missing from the baseline feature sets.

5.6.3 Choice of learning algorithm. The most popular choices of learning algorithm in data science projects are Logistic Regression [25], Random Forest [25], and more recently XGBoost [7] and CatBoost [26]. Gradient boosting methods have a successful track record in a variety of learning problems with heterogeneous features, noisy data, and complex dependencies. Table 3 compares the performances of the four learning algorithms using the features from Section 4. CatBoost performs best in all cases, with XGBoost a close second. This narrow victory can be attributed to CatBoost’s intelligent handling of categorical features compared to XGBoost’s more naive one-hot encoding.

⁵ The *no features* baseline always predicts the mean class probability, i.e., if our data set contains 1.5% positive examples, we always predict 0.015.

Table 3: Different design choices evaluated on both scoring and conceding probabilities using the Brier score and ROC AUC. For the Brier score lower values are better, whereas for ROC AUC higher values are better.

Design	Choice	P_{scores}		$P_{concedes}$	
		Brier	AUC	Brier	AUC
Feature set	All features	0.01376	0.7693	0.00547	0.7313
	No features	0.01632	-	0.00564	-
	Location	0.01562	0.7330	0.00560	0.6770
	Action type	0.01590	0.6405	0.00562	0.6348
	Loc + Action type	0.01549	0.7417	0.00550	0.6912
Algorithm	CatBoost	0.01376	0.7693	0.00547	0.7313
	Logistic Regression	0.01601	0.7231	0.00562	0.6578
	Random Forest	0.01409	0.7050	0.00552	0.6457
	XGBoost	0.01390	0.7556	0.00550	0.7255

5.7 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 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.

6 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 Nørstebø et al. [23], Bransen et al. [2] and Fernández et al. [11] for soccer, Routley and Schulte [27] and Liu and Schulte [19] for ice hockey, and Cervone et al. [6] 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 [18]. In contrast to Nørstebø et al. [23] and Routley and Schulte [27], which divide the pitch into a fixed number of zones, our approach models the exact locations of each action. Unlike Cervone et al. [6], 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 [16, 28], American football [13], and baseball [29].

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. Decroos et al. [10], Knutson [17], and Gregory [14] address the task of valuing the actions leading up to a goal attempt, whereas Bransen et al. [4], Bransen and Van Haaren [3], and Gyarmati and Stanojevic [15] 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 [1, 5, 8, 20, 21]. 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.

7 CONCLUSION

This paper introduced SPADL, a language for representing event stream data that is designed with the goal of facilitating data analysis, and 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, and shots), (2) bases its valuation on the game context, and (3) 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.

8 ACKNOWLEDGMENTS

Tom Decroos is supported by the Research Foundation-Flanders (FWO-Vlaanderen). Jesse Davis is partially supported by the EU Interreg VA project Nano4Sports and the KU Leuven Research Fund (C14/17/07, C32/17/036). The authors thank Wyscout for supplying the event stream data used in this paper.

REFERENCES

- [1] Daniel Altman. 2015. Beyond Shots: A New Approach to Quantifying Scoring Opportunities. (2015). <http://northyardanalytics.com/Dan-Altman-NYA-OptaPro-Forum-2015.pdf> OptaPro Analytics Forum.
- [2] Lotte Bransen, Pieter Robberechts, Jan Van Haaren, and Jesse Davis. 2019. Choke or Shine? Quantifying Soccer Players’ Abilities to Perform Under Mental Pressure. In *MIT Sloan Sports Analytics Conference*.
- [3] Lotte Bransen and Jan Van Haaren. 2018. 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*.
- [4] Lotte Bransen, Jan Van Haaren, and Michel van de Velden. 2019. Measuring Soccer Players’ Contributions to Chance Creation by Valuing Their Passes. *Journal of Quantitative Analysis in Sports* (2019).
- [5] Michael Caley. 2015. Premier League Projections and New Expected Goals. (2015). <https://cartilagefreecaptain.sbnation.com/2015/10/19/9295905/premier-league-projections-and-new-expected-goals> Cartilage Free Captain.
- [6] Dan Cervone, Alexander D’Amour, Luke Bornn, and Kirk Goldsberry. 2014. POINTWISE: Predicting Points and Valuing Decisions in Real Time with NBA Optical Tracking Data. In *MIT Sloan Sports Analytics Conference*.
- [7] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 785–794.
- [8] Tom Decroos, Vladimir Dzyuba, Jan Van Haaren, and Jesse Davis. 2017. Predicting Soccer Highlights from Spatio-Temporal Match Event Streams. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*. 1302–1308.
- [9] Tom Decroos, Jan Van Haaren, and Jesse Davis. 2018. Automatic Discovery of Tactics in Spatio-Temporal Soccer Match Data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM.
- [10] Tom Decroos, Jan Van Haaren, Vladimir Dzyuba, and Jesse Davis. 2017. STARSS: A Spatio-temporal Action Rating System for Soccer. In *ECML/PKDD 2017 Workshop on Machine Learning and Data Mining for Sports Analytics*.
- [11] Javier Fernández, Luke Bornn, and Dan Cervone. 2019. Decomposing the Immeasurable Sport: A Deep Learning Expected Possession Value Framework for Soccer. In *MIT Sloan Sports Analytics Conference*.
- [12] César Ferri, José Hernández-Orallo, and R. Modroiu. 2009. An Experimental Comparison of Performance Measures for Classification. *Pattern Recognition Letters* 30, 1 (2009), 27–38.
- [13] Keith Goldner. 2012. A Markov Model of Football: Using Stochastic Processes to Model a Football Drive. *Journal of Quantitative Analysis in Sports* 8, 1 (2012).
- [14] Sam Gregory. 2017. How We Assign Credit in Football. (2017). <http://www.optasportspro.com/about/optapro-blog/posts/2017/blog-how-we-assign-credit-in-football/> OptaPro Blog.
- [15] László Gyarmati and Rade Stanojevic. 2016. QPass: A Merit-based Evaluation of Soccer Passes. In *KDD 2016 Workshop on Large-Scale Sports Analytics*.
- [16] Nobuyoshi Hirotsu, Michael Wright, et al. 2002. 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).
- [17] Ted Knutson. 2017. Introducing xGChain. (2017). <http://www.statsbomb.com/introducing-xgchain-StatsBomb-IQ-Services>.
- [18] Michael Littman. 1994. Markov Games as a Framework for Multi-Agent Reinforcement Learning. In *Proceedings of the International Conference on Machine Learning*.
- [19] Guiliang Liu and Oliver Schulte. 2018. Deep Reinforcement Learning in Ice Hockey for Context-Aware Player Evaluation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*. 3442–3448.
- [20] Patrick Lucey, Alina Bialkowski, Mathew Monfort, Peter Carr, and Iain Matthews. 2014. Quality vs. Quantity: Improved Shot Prediction in Soccer Using Strategic Features from Spatiotemporal Data. In *MIT Sloan Sports Analytics Conference*.
- [21] Nils Mackay. [n.d.]. *Predicting Goal Probabilities for Possessions in Football*. Master’s thesis. Vrije Universiteit Amsterdam.
- [22] Alexandru Niculescu-Mizil and Rich Caruana. 2005. Predicting Good Probabilities with Supervised Learning. In *Proceedings of the Twenty-Second International Conference on Machine Learning*. 625–632.
- [23] Olav Nørstebø, Vegard Rødseth Bjertnes, and Eirik Vabo. 2016. *Valuing Individual Player Involvements in Norwegian Association Football*. Master’s thesis. Norwegian University of Science and Technology.
- [24] Luca Pappalardo, Paolo Cintia, et al. 2018. PlayeRank: data-driven performance evaluation and player ranking in soccer via a machine learning approach. *arXiv preprint arXiv:1802.04987* (2018).
- [25] Fabian Pedregosa, Gaël Varoquaux, et al. 2011. scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12, Oct (2011), 2825–2830.
- [26] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. 2018. CatBoost: Unbiased Boosting with Categorical Features. In *Advances in Neural Information Processing Systems*. 6639–6649.
- [27] Kurt Routley and Oliver Schulte. 2015. A Markov Game Model for Valuing Player Actions in Ice Hockey. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*. 782–791.
- [28] Sarah Rudd. 2011. A Framework for Tactical Analysis and Individual Offensive Production Assessment in Soccer Using Markov Chains. In *New England Symposium on Statistics in Sports*. <http://nessis.org/nessis11/rudd.pdf>
- [29] Tom Tango, Mitchel Lichtman, and Andrew Dolphin. 2007. *The Book: Playing the Percentages in Baseball*. Potomac Books, Inc.

A APPENDIX ON REPRODUCIBILITY

A.1 SPADL action types

Table 4 provides an overview of the 21 action types in the SPADL representation alongside their descriptions.

A.2 Data description

We ran our experiments on Wyscout data for the English, Spanish, German, Italian, French, Dutch, and Belgian top divisions. We considered 11,565 games played in the 2012/2013 through 2017/2018 seasons. After transforming the Wyscout data to our SPADL representation, each game contains ± 1250 actions on average. As shown in Figure 4, the most frequent action types in our data set are passes (64.63%), dribbles (8.69%), and interceptions (5.01%).

A.3 Experimental setup and implementation

We performed all experiments in this paper in Python. We evaluated the performance of four popular learning algorithms:

Logistic Regression We used the implementation from the `scikit-learn`⁶ Python package. We used an L2 regularization penalty and L-BFGS as the solver for the optimization problem.

Random forest We used the implementation from the `scikit-learn`⁷ Python package. We trained a forest of 100 trees using 40 parallel threads.

XGBoost We used the official Python implementation.⁸ We trained 100 trees of maximum depth 3 with a learning rate of 0.1 using 40 parallel threads.

CatBoost We used the official Python implementation from Yandex.⁹ We set all parameters to their default values, except for the number of parallel threads, which we set to 40.

We trained and evaluated all models on a computing server running Ubuntu 16.04 with 128GB of RAM and two CPUs of type Xeon(R) CPU E5-2630 v4 @ 2.20GHz, providing a maximum of 20 cores and 40 threads. The runtime for each learning algorithm per task is available in Table 5.

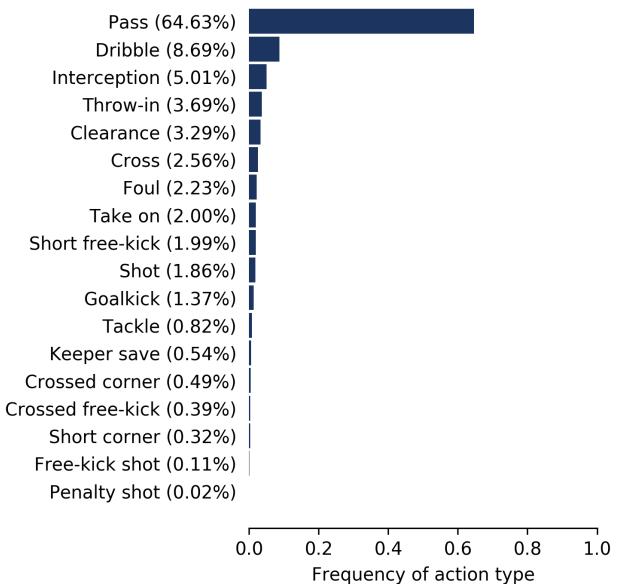


Figure 4: Frequency of each action type in the Wyscout data converted to the SPADL representation.

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

⁷<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

⁸<https://xgboost.ai>

⁹<https://tech.yandex.com/catboost/>

Table 4: 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 result values.

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	-

Table 5: Runtimes for each learning algorithm per task. The two training sets, seasons 2012/2013 through 2015/2016 and seasons 2012/13 through 2016/2017, contain respectively 8,518,378 and 11,438,956 actions each. The two evaluation sets, season 2016/2017 and season 2017/2018, contain respectively 2,920,578 and 2,988,847 actions each.

Task	Logistic Regression	Random Forest	XGBoost	CatBooost
Training on seasons 2012/2013 - 2015/2016	4 minutes	16 minutes	16 minutes	100 minutes
Training on seasons 2012/2013 - 2016/2017	6 minutes	25 minutes	22 minutes	140 minutes
Predicting on season 2016/2017	30 seconds	1 minute	40 seconds	3 minutes
Predicting on season 2017/2018	30 seconds	1 minute	50 seconds	3 minutes