Inferring the Strategy of Offensive and Defensive Play in Soccer with Inverse Reinforcement Learning

Pegah Rahimian[1] and Laszlo Toka[1,2]

[1] Budapest University of Technology and Economics, Budapest, Hungary
[2] MTA-BME Information Systems Research Group
{pegah.rahimian,toka}@tmit.bme.hu

Abstract. Analyzing and understanding strategies applied by top soccer teams has always been in the focus of coaches, scouts, players, and other sports professionals. Although the game strategies can be quite complex, we focus on the offensive or defensive approaches that need to be adopted by the coach before or throughout the match. In order to build interpretable parameterizations of soccer decision making, we propose a batch gradient inverse reinforcement learning for modeling the teams' reward function in terms of offense or defense. Our conducted experiments on soccer logs made by Wyscout company on German Bundesliga reveal two important facts: the highest-ranked teams are planning strategically for offense and defense before the match with the largest weights on pre-match features; the lowest-ranked teams apply short-term planning with larger weights on in-match features.

Keywords: Soccer Analytics · Inverse Reinforcement Learning · Deep Learning · Decision Making.

1 Introduction

Although soccer is a relatively simple sport compared to other sports in terms of rules and basic game play, deciding about strategies to be applied with the aim of wining can be quite complex. Among the vast options of the applied strategies, playing offensive or defensive is one of the most important decisions that needs to be taken by the coach before, or throughout the matches. There are several factors that might affect this decision. Several studies proposed conceivable factors and robust methods for deriving the optimal strategies to be applied for a team (e.g., [8],[7],[6],[22]).

Apart from deriving the optimal strategy, understanding the logic behind the decisions made by high-ranked clubs in the leagues, helps other clubs to learn those decisions and possibly imitate those. In this work, we propose a novel soccer strategy analysis method, taking advantage of inverse reinforcement learning (IRL). IRL is the field of learning an expert's objectives, values, or rewards by observing its behavior. The motivation behind the choice of this algorithm is that

we can observe the behavior of a team in some specific matches, and learn which states of the environment the team is trying to achieve and what the concrete goals might be. On the other hand, soccer is a sparse rewarding game. Thus, designing a manual reward assignment method for each action of the players might not be an easy task. IRL helps to infer the intention behind the smart offensive or defensive strategies throughout the matches by recovering the reward function. In summary, we seek to answer the following questions: "why does a team decide to play offensive or defensive?", "what are the most important features for each team to decide about this strategy?", "how to distinguish offensive teams from defensive ones?". The contribution of our work is multi-fold:

- We propose an end-to-end framework that receives raw actions, and infers the intention behind those via IRL by converting soccer match action logs to a possession model in an environment, which we assume to be Markovian;
- We propose a model to coaches and sports professionals to understand the policy of the top clubs, and possibly imitate those by developing robust deep recurrent neural networks for cloning the offensive and defensive behavior of the soccer clubs from their match logs;
- We design a novel reward function and corresponding features for maximizing teams' winning probabilities, and we recover those from soccer match logs;
- We make our code available online[3].

This paper is organized as follows. We present the current literature on sports analytics in Section 2. Section 3 provides some preliminaries of GIRL, an IRL method we apply. We explain our IRL framework for soccer analytics in Section 4. Section 5 describes the experimental IRL framework and extensive numerical computations for getting interesting inference results. Finally, we conclude our work in Section 6.

2 Related work

Soccer players mostly take actions according to the rewards that they expect to gain from their behavior. This reward is usually dictated by the coach to the players. Although, several works tried to suggest generic action valuation methods, focusing on passes and shots (e.g., [5], [17], [4], [9], etc.), and some others cover all types of actions (e.g., [2], [13], [12], etc.), recovering the assigned reward of offensive and defensive actions, which is specific for each soccer team, is ignored in the literature of sports analytics. The works by Gambarelli et al. [8], and Hirotsu et al. [10] derived the optimal strategy of playing offensive or defensive via game-theory. In this paper, we focus on inferring the intention behind those strategies for different teams rather than deriving the optimal strategy for them.

Recently, deep learning models proved to be promising in soccer analytics. In this domain, Fernandez and Bornn [4] present a convolutional neural network

[3] `https://github.com/Peggy4444/soccer_IRL`

architecture that is capable of estimating full probability surfaces of potential passes in soccer. Liu et al. took advantage of Q-function in reinforcement learning for action valuation in ice-hockey [13] and soccer [12]. With regards to the application of IRL in team sports, Luo et al. [14] combined Q-function with IRL to provide a player ranking system, and Muelling et al. [16] used IRL for extracting strategic elements from table tennis data. However, a particular application of gradient IRL to recover the previously assigned reward of the performed offensive and defensive actions is missing in the literature. In this work, we use a truly batch Gradient IRL method, which eliminates the necessity of environment dynamics and online interaction of the players with the environment. Our method extracts intention behind strategies solely from soccer logs, thus, conforms to real soccer matches and is applicable for coaches and sports professionals.

3 Preliminaries of Gradient Inverse Reinforcement Learning

In this section, we provide basic notations and formulations of IRL, used throughout this paper.

3.1 Markov Decision Processes without Reward

A Markov Decision Process without Reward ($MDP\backslash R$) [19] is denoted by the tuple (S, A, P), where S is the state space, A is the action space, and $P : S \rightarrow A \rightarrow S$ is the transition function. In this environment, the expert's behavior is described by a stochastic policy $\pi : S \rightarrow A$. In this work, we consider that policies are differentiable and belong to a parametric space $\prod_{\Theta} = \{\pi_\theta : \theta \in \Theta\}$, where θ is the policy parameter.

3.2 Inverse Reinforcement Learning

The goal of IRL is to infer a reward function R given an optimal policy $\pi^* : S \rightarrow A$ for the $MDP\backslash R$. Typically, we observe samples (s, a) of states and actions recorded from full history of expert's trajectories $\tau = (s_0, a_0, ..., s_{T-1}, a_{T-1}, s_T)$, which are following policy π^*. In order to recover the rewards gained by each action of the expert, we define a parametric linear reward function as the weighted combination of features in (1).

$$R_\omega(s, a) = \sum_{i=1}^{q} \omega_i f_i(s, a) = \omega^T \mathbf{f}(s, a), \tag{1}$$

where $\mathbf{f}$ is the vector of reward features, ω is the vector of weights, and q is the number of our selected reward features. Moreover, ω^E can be defined as the weight vector of the rewards, which we assume to be optimized by expert E.

Furthermore, the feature expectation of policy π can be described as:

$$\psi^{\pi} = E\left[\sum_{t=0}^{\infty} \gamma^t \mathbf{f}(S_t, A_t)\right], \tag{2}$$

where γ is discount factor (set to 0.99 in this work), $A_t \sim \pi(.|S_t)$, and $\pi_\theta \in \Pi_\Theta$, that $(\psi(\theta) = \psi^{\pi_\theta})$. Finally, the expected value under policy π_θ of our reward feature vector $\mathbf{f}$ can be formulated as:

$$J(\theta, \omega) = E\left[\sum_{t=0}^{\infty} \gamma^t R_\omega(S_t, A_t)\right] = \omega^T \psi(\theta). \tag{3}$$

3.3 Gradient Inverse Reinforcement Learning

Solving an IRL problem through Policy Gradient (PG) is a straightforward solution when the policy is parameterized and can be estimated and consequently represented through its parameter θ. Several algorithms using this method are proposed by different studies, e.g., [15], [18], [20].

In general, policy gradient under linear reward function can be defined as the gradient of the expected value of the expert's policy as in (4).

$$\nabla_\theta J(\theta, \omega) = E_{\tau \sim \pi_\theta}\left[\left(\sum_{l=0}^{t} \nabla_\theta \log \pi_\theta(A_l|S_l)\right)\left(\sum_{t=0}^{\infty} R_\omega(S_t, A_t)\right)\right], \tag{4}$$

and according to (3),

$$\nabla_\theta J(\theta, \omega) = \nabla_\theta \psi(\theta)\omega, \tag{5}$$

where $\nabla_\theta \psi(\theta)$ is a Jacobian matrix. Assuming that the expert optimized the policy under some unknown R^E, its policy gradient should be zero in R^E. In other words, if the expert's policy π_{θ^E} optimizes its reward function R_{ω^E}, then the policy parameter θ_E will be a stationary point of the expected value $J(\theta, \omega^E)$. Thus, one way to recover the weight ω^E is to get it from the null space of the Jacobian $\nabla_\theta \psi(\theta)$. A good approach is discussed in the method called Gradient Inverse Reinforcement Learning (GIRL) [18]. GIRL is a method of recovering reward function that minimizes the gradient of a parameterized representation of the expert's policy. At the first step, we assume that the expert's policy π^E is known, and GIRL tries to recover the weight ω^E, associated with its reward function R^E. Pirotta et al. [18] discuss that estimating the Jacobian matrix $\nabla_\theta \psi(\theta)$ from expert trajectories might result in a full rank matrix. Thus, it might prevent finding the corresponding null space. As a solution to this problem, GIRL proposes recovering ω by searching for the direction of minimum growth by minimizing the $L^2 - norm$ of gradient, as in (6):

$$\min_{\omega} \left\| \nabla_\theta \psi(\theta^E)\omega \right\|_2^2 \tag{6}$$

4 IRL framework for reward recovery in soccer

In this section, we propose an end-to-end framework, which gets the raw expert team's trajectories from the soccer logs, estimates expert team's policy by training robust convolutional recurrent neural networks and understands the intuition behind playing offensive or defensive actions by recovering the expert team's reward function through GIRL.

4.1 Behavioral cloning

The estimation of the Jacobian matrix $\nabla_\theta \psi(\theta)$, i.e., policy gradient, can be performed with several methods, such as REINFORCE algorithm [21], or G(PO)MDP [1]. Furthermore, an approximation of the expert's policy parameter θ^E can be estimated through Behavioral Cloning (BC) in the expert's trajectories. One reasonable approach to estimate this parameter is to use Maximum-Likelihood estimation. In this work, we proceed with developing robust deep neural networks, which are able to estimate the expert's policy from its trajectories. Consequently, feeding any kind of state with the corresponding features from expert's trajectory, this network should accurately estimate the occurrence probability of action space: $\pi(a|s)$. Thus, the expert's policy will be accurately learned by the networks.

IRL experts, actions, and trajectories in soccer We aim to construct a Markovian possession environment from soccer logs. To this end, data preparation is a core task to achieve a reliable IRL model.

In our soccer analysis task, we assume that the coach and players of each team are always trying to maximize their winning probabilities. Thus, the performed policy of playing offensive or defensive, which is dictated by the coach and obeyed by the players, is optimal in their own opinion. This assumption totally matches the optimality assumption of IRL, in which the expert policy from its demonstrations must be optimal in their own opinion. In this setting, we consider each team competing in the leagues as an IRL expert. Therefore, we have a set of expert teams $\mathbf{E} = (E_1, E_2, ..., E_m)$, and set of unknown reward functions $\mathbf{R} = (R_{\omega_1}, R_{\omega_2}, ..., R_{\omega_m})$ for each of them.

Moreover, each expert team demonstrates a set of trajectories. In a game, in which two teams are competing with each other, we always set one team as the IRL expert, and the other team as the opponent. Both teams demonstrate a set of ball possessions, i.e., action sequences. In this work, we assume that the possession is transferred if and only if the team is not in the possession of the ball over two consecutive events. Thus, the unsuccessful touches of the opponent in fewer than 3 consecutive actions are not considered as a possession loss.

In the offensive vs defensive analysis, we define each trajectory τ_t as one possession of the match. Then we separate the possessions according to their possessor of being the expert team or the opponent. Now the expert team has four options as an action: perform offensive action by terminating an own possession with a "shot", or perform defensive actions for terminating the opponent's

possession with "tackle", "clearance", or "interception". Consequently, each expert team demonstrates the set of trajectories as $D_i = (\tau_1, \tau_2, ..., \tau_t)$, which terminate with the four above-mentioned offensive or defensive actions.

State representation: In order to address the sequential nature of actions in the soccer logs, we define the state as one possession. We describe a game state by generating the most relevant state features and labels to them. In order to demonstrate and prepare states for machine learning, we built a set of hand-crafted spatial state features. For each time-step, we demonstrate the state as a 7-dimensional feature vector X (see Table 1), and one-hot representation of the action A for all the actions within each possession, excluding the ending action. Thus, the varying possession length is the number of actions inside a possession, excluding the ending one. Then, the state is a 2-dimensional array, with the first dimension of possession length (varying for each possession), and the second dimension of features (of the fixed length of 7). Therefore, a m^{th} state, i.e., m^{th} possession, with length of n actions is represented as:

$$S_m = [[X_0, A_0], [X_1, A_1], ..., [X_{n-1}, A_{n-1}]]$$

.

Table 1: State Features List

State feature name	**Description**
Angle to goal	the angle between the goal posts seen from the shot location
Distance to goal	Euclidean distance from shot location to center of the goal line
Time remaining	time remained from action occurrence to the end of match half
Home/Away	is the action performed by home or away team?
Action result	successful or unsuccessful
Body ID	is the action performed by head or body or foot?
Goal difference	actual difference between the expert team and opponent goals

Network architecture As mentioned before, each soccer team is considered as an expert denoted by E_i. As the first step, we are interested in recovering the behavioral policy of each expert team from the set of trajectories demonstrated by them. To this end, we formulate the problem as follows: each team plays 34 matches in the league in 34 rounds, competing with 17 other opponent teams. Thus, we need to run IRL 18 times for 18 independent expert teams. In order to recover the behavioral policy for each of them, we collected 2 different datasets. The first dataset consists of all possessions of the expert team through one season of the league. The second dataset is the concatenation of all possessions of the other 17 opponent teams competing with the expert team in that league. Then, we trained a CNN-LSTM network [3], using CNN for spatial feature extraction of input possessions, and an LSTM layer with 100 memory units to

support sequence prediction and to interpret features across time steps. Since the input possessions have a three-dimensional spatial structure, CNN is capable of picking invariant features for different actions inside a possession. Then, these learned consolidated spatial features serve as input to the LSTM layer. Finally, the dense output layers are used with sigmoid and softmax activation functions to perform our binary and multi-classification task. The sigmoid activation function classifies the offensive possessions to the two ending actions of "Shot" and "Not Shot", and softmax activation function classifies the defensive possessions (i.e., the offensive possessions from opponent which are terminated by expert team through a defensive action) to the four ending actions of "Interception", "Tackle", "Clearance", and "Others".

Note that the feature vector X is of fixed length for each action, but it varies for all actions in the state (because possession length or the number of actions varies). This is one of the main challenges in our work as most machine learning methods require fixed-length feature vectors. In order to address this challenge, we use truncating and padding. We mapped each action in possession to a 9 length real-valued vector. Also, we limit the total number of actions in a possession to 10, truncating long possessions and we pad the short possessions with zero values. The architecture for cloning behavioral policy of the expert team is depicted in Figure 1.

- The offensive network is trained with all expert team possessions, and estimates the probability of the expert team's possession resulting in a shot. Intuitively, this network recovers the offensive part of the expert team's policy by estimating the probability of expert's possessions ending up in a shot.
- The defensive network is trained with all opponent's team possessions in the same matches with the expert team, and estimates the probability of the opponent team's possession ending in one of the defensive actions, (i.e., clearance, tackle, intercept) made by the expert team. The aim of this network is to recover the defensive part of the expert team policy.

4.2 Rewards features and weights recovery

We aim to employ GIRL algorithm to infer the intention behind soccer teams for playing offensive or defensive tactics through the matches with different opponents. More specifically, we seek to find valid answers for the following questions: "Why does a team decide to play offensive or defensive? How do they reward the offensive vs defensive actions throughout the match?". To achieve this goal, first, we need to define reward functions according to (1). In our soccer analysis, the features representing the reward are the combination of pre-game information (e.g., own ranking, opponent ranking, home advantage) and in-game information at each moment of the match (e.g., goal difference, time remaining, and player's intention to play offensive or defensive - correlating with the player's market value according to the money-ball analysis in [11]). Table 2 lists the corresponding reward features and their correlations with the number of offensive actions

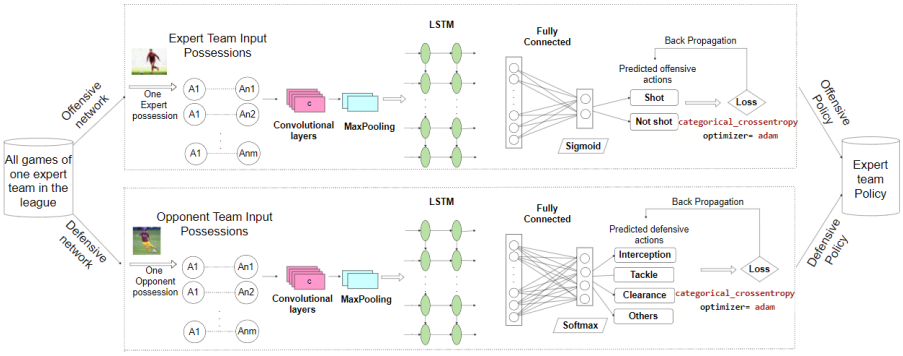


Fig. 1: Offensive and Defensive networks for the expert team behavioral cloning

in the Bundesliga dataset. Thus, we formulate the reward function according to (7).

$$R_\omega = \underbrace{\omega_1 f_1}_{\text{Expert rank}} + \underbrace{\omega_2 f_2}_{\text{Opponent rank}} + \underbrace{\omega_3 f_3}_{\text{H/A}} + \underbrace{\omega_4 f_4}_{\text{GD}} + \underbrace{\omega_5 f_5}_{\text{TR}} + \underbrace{\omega_6 f_6}_{\text{PI}} \tag{7}$$

Now we can employ GIRL algorithm to recover weights ($\omega_i s$), according to (4),(5),(6), to understand what each team cares about the most, i.e., which set of features. After recovering the weights, we will be able to compute the assigned reward to each action through the matches by the personal opinion of the coach and players.

Table 2: Reward Features List

Notation	**Reward feature name**	**Type**	**Description**	**Correlation to offensive actions**
f_1: Expert rank	Expert team ranking	pre-game	Inverse of the expert team place in the league table at the date of the match; fixed in a match; varying through a season	0.63
f_2: Opponent rank	Opponent team ranking	pre-game	Inverse of the opponent team place in the league table at the date of the match; fixed in a match; varying through a season	-0.21
f_3: H/A	Home/Away	pre-game	Expert team is home team or away team?; fixed in a match; varying through a season	0.33
f_4: GD	Goal Difference	in-game	Goal difference of expert team and opponent at each time-step; varying in a match; varying through a season	-0.12
f_5: TR	Time Remaining	in-game	Time remaining to the end of the game at each time-step; varying in a match; varying through a season	0.39
f_6: PI	Player Intention $\sim$ market value[11]	in-game	Offensive or defensive intention of the player performing the action at each timestep; varying in a match; varying through a season	0.41

5 Experiments and results

Dataset: In order to conduct the experiments of our proposed approach, we use a match event dataset[4] provided by Wyscout. The Wyscout dataset covers 1,941 matches, 3,251,294 events, and 4,299 players from an entire season of seven competitions (La Liga, Serie A, Bundesliga, Premier League, Ligue 1, FIFA World Cup 2018, UEFA Euro Cup 2016). Our results prove the sufficiency of this dataset size for our experiments. Moreover, players' market value and dynamic teams ranking in the leagues are collected from `transfermarkt.com`, and `worldfootball.net`. In order to facilitate the reproducibility of the analysis, we converted the format of event stream data to SPADL representation[5]. In this section we show the result of our experiment on the 2017-2018 season of German Bundesliga competition, which consists of 306 matches, 142 teams, and 519407 events. The teams in the collection of matches are the following: Bayern München, Bayer Leverkusen, Augsburg, Eintracht Frankfurt, Borussia M'gladbach, Köln, Werder Bremen, Hoffenheim, Hannover 96, Stuttgart, Mainz 05, Schalke 04, Wolfsburg, Hertha BSC, Freiburg, Hamburger SV, RB Leipzig, and Borussia Dortmund.

Fig. 2: Range of reward weights recovered by GIRL. The algorithm finds large pre-game weights for high-ranked teams, whereas large in-game weights for low-ranked teams in 2017-2018 season of German Bundesliga.

Behavioral cloning results: For each iteration of the GIRL algorithm, one team in this collection is assigned as the IRL expert team, and the rest of the teams are the opponents. Thus, the prepared dataset for each result consists of 34 matches of an expert team from 34 rounds of 2017-2018 season of German Bundesliga, in which the expert team participated. As the first step, we estimated the Jacobian $\nabla_\theta \psi(\theta)$ by training the offensive and defensive networks separately for each expert team. Learning all weights from the expert's team trajectories in the Bundesliga dataset took about 90 seconds (5 seconds for each expert team on average) on a server with a Tesla K80 GPU. The input data size to each network was approx. 3300 possessions for each of the expert team and opponent

[4] `https://figshare.com/collections/Soccer_match_event_dataset/4415000/5`
[5] `https://github.com/ML-KULeuven/socceraction`

teams. Consequently, each network constructed 56,890 trainable parameters on average. Validation split of 30% of consecutive possessions is used to evaluate BC models during training, and cross-entropy loss on train and validation datasets is used to evaluate the model, achieving the accuracy of 79% and loss of 0.4 (cross entropy) on average through all expert teams for predicting the actions.

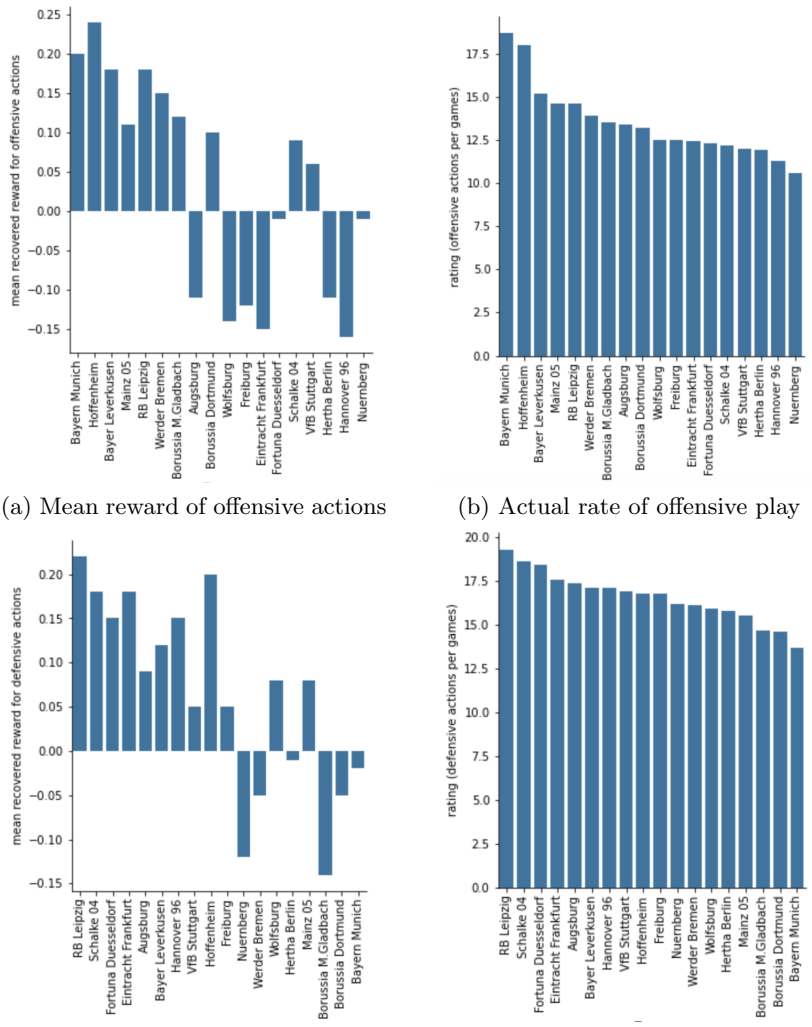
Reward weights recovery results: Table 3 presents the results of reward weight estimation for each expert team in Bundesliga. The value of each recovered weight is an indicator of the importance of that feature for that expert team. For instance, the "ω_1: own ranking" feature is the most important feature for FC Schalke team. Hoffenheim mostly cares about "ω_2: opponent ranking" feature in its reward assignment. After recovering the weight of the rewards by GIRL, one can estimate the assigned reward using (7) for each team. We classified the teams into 3 categories according to their final ranking in the league: high-ranked, middle-ranked, and low-ranked teams. By analyzing the range of weights recovered by GIRL, we found it surprising that high-ranked teams mostly pay attention to the pre-game features (own ranking, opponent ranking, home advantage) with large weights: $(\omega_1, \omega_2, \omega_3)$. Thus, it seems that the coaches in these teams are planning strategically to play offensive or defensive. On the other hand, low-ranked teams apply short-term planning with large weights $(\omega_4, \omega_5, \omega_6)$ on in-match features: (goal difference, time remaining, and player intention). Moreover, middle-ranked teams show small variance except for ω_5: time remaining. Figure 2 is the evidence of these claims.

Table 3: Reward weights recovered for expert teams

		pre-game features			**in-game features**		
Rank	**Expert team**	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6
1	Bayern München	0.20	-0.22	0.28	0.12	0.18	0.00
2	FC Schalke 04	0.32	-0.27	0.20	0.10	-0.01	0.09
3	1899 Hoffenheim	0.10	-0.35	0.22	0.17	0.04	0.12
4	Borussia Dortmund	0.29	-0.31	0.11	0.12	-0.17	0.00
5	Bayer Leverkusen	0.15	-0.20	0.15	0.14	0.18	0.22
6	RB Leipzig	0.22	-0.20	0.18	0.15	-0.15	0.1
7	VfB Stuttgart	0.32	-0.15	0.12	0.15	0.04	0.22
8	Eintracht Frankfurt	0.10	-0.22	0.14	0.12	0.33	0.09
9	Mönchengladbach	0.11	-0.20	0.11	0.11	0.42	0.05
10	Hertha BSC	0.11	-0.08	0.11	0.21	-0.11	0.31
11	Werder Bremen	0.11	-0.15	0.20	0.11	-0.32	0.11
12	FC Augsburg	0.09	-0.11	0.12	0.11	0.46	0.11
13	Hannover 96	0.05	-0.08	0.19	0.41	0.10	0.11
14	1. FSV Mainz 05	0.11	-0.20	0.22	0.25	-0.15	0.01
15	SC Freiburg	0.03	-0.15	0.14	0.32	-0.32	0.22
16	VfL Wolfsburg	0.09	-0.08	0.18	0.21	0.33	0.11
17	Hamburger SV	0.12	-0.11	0.12	0.31	-0.31	0.25
18	1. FC Köln	0.11	-0.09	0.15	0.22	-0.22	0.31

Evaluation: Using the 2018-2019 season of Bundesliga dataset, Figure 3 shows a comparison of the mean recovered reward for offensive and defensive actions by our proposed approach, versus the actual offensive/defensive rating collected from `whoscored.com`. It is the evidence of the robustness of our reward recovery approach as it recovered larger positive reward of offensive actions for

the offensive teams (e.g., Bayern, Hoffenheim, etc.) in Figures 3a and 3b, and larger positive reward of defensive actions for the defensive teams (e.g., Leipzig, Schalke, etc.) in Figures 3c and 3d.



(a) Mean reward of offensive actions

(b) Actual rate of offensive play

(c) Mean reward of defensive actions

(d) Actual rate of defensive play

Fig. 3: Comparison of the mean recovered reward with the actual rating of offensive and defensive play

6 Conclusion

In the current literature of sports analytics, the following two domains are intensively studied by researchers: 1) action valuation methods, 2) deriving optimal strategies. However, the study of inferring the reward assigned to offensive and defensive actions by the coaches and players, which are team-specific, is missing in the literature. In this work we proposed a model for inferring the intention behind playing offensive or defensive tactics according to some intuitive features. Our experimental results showed that the high-ranked teams mostly plan based on pre-match information, and play strategically. The low-ranked teams plan on short-term based on information available throughout the game. Finally, we showed how the recovered reward of *offensive* and *defensive* actions in German Bundesliga teams are conforming with their final ranking in the league with respect to their *offensive* and *defensive* plays. Our framework will help coaches and sports professionals to infer the intention of soccer actions from highest-rank clubs, and possibly imitate those. Moreover, the recovered rewards for each action can be used for any action and player evaluation tasks. To the best of our knowledge, our work constitutes the first usage of truly batch gradient inverse reinforcement learning to infer the intention behind offensive or defensive plays in soccer. As future work, we aim to use the recovered rewards for deriving the optimal strategy to be applied by each team via reinforcement learning.

Acknowledgment

Project no. 128233 has been implemented with the support provided by the Ministry of Innovation and Technology of Hungary from the National Research, Development and Innovation Fund, financed under the FK_18 funding scheme.

References

1. Baxter, J., Bartlett, P.L.: Infinite-horizon policy-gradient estimation. Journal of Artificial Intelligence Research **15**, 319–350 (2001)
2. Decroos, T., Bransen, L., Van Haaren, J., Davis, J.: Actions speak louder than goals: Valuing player actions in soccer. In: In The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '19) (2019)
3. Donahue, J., Hendricks, L.A., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. CoRR **abs/1411.4389** (2014)
4. Fernandez, J., Born, L.: Soccermap: A deep learning architecture for visually-interpretable analysis in soccer. In: ECML PKDD (2020)
5. Fernandez, J., Bornn, L., Cervone, D.: Decomposing the immeasurable sport: A deep learning expected possession value framework for soccer. In: In Proceedings of the 13th MIT Sloan Sports Analytics Conference (2019)
6. Fernandez-Navarro, J.: Analysis of styles of play in soccer and their effectiveness. Ph.D. thesis, Universidad de Granada (2018)

7. Fernandez-Navarro, J., Fraduab, L., Zubillagac, A., R. Forda, P., P. McRobert, A.: Attacking and defensive styles of play in soccer: analysis of spanish and english elite teams. Journal of Sports Sciences **34**, 1–10 (2016)
8. Gambarelli, D., Gambarelli, G., Goossens, D.: Offensive or defensive play in soccer : a game-theoretical approach. Journal of Quantitative Analysis in Sports **15**(4), 261–269 (2019)
9. Gyarmati, L., Stanojevic, R.: Qpass: a merit-based evaluation of soccer passes (2016)
10. Hirotsu, N., Wright, M.: Modeling tactical changes of formation in association football as a zero-sum game. Journal of Quantitative Analysis in Sports **2** (2006)
11. Inna, Z., Daniil, S.: Moneyball in offensive vs defensive actions in soccer. European Economics: Labor & Social Conditions eJournal (2020)
12. Liu, G., Luo, Y., Schulte, O., Kharra, T.: Deep soccer analytics: learning an action-value function for evaluating soccer players. Data Mining and Knowledge Discovery **34**(2) (2020)
13. Liu, G., 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 (IJCAI-18) (2018)
14. Luo, Y., Schulte, O., Poupart, P.: Inverse reinforcement learning for team sports: Valuing actions and players. In: Bessiere, C. (ed.) Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20. pp. 3356–3363 (2020)
15. Metelli, A.M., Pirotta, M., Restelli, M.: Compatible reward inverse reinforcement learning. In: The Thirty-first Annual Conference on Neural Information Processing Systems (2017)
16. Muelling, K., Boularias, A., Mohler, B., Schoelkopf, B., Peters, J.: Inverse reinforcement learning for strategy extraction. In: In ECML PKDD 2013 Workshop on Machine Learning and Data Mining for Sports Analytics (MLSA 2013) (2013)
17. Peralta Alguacil, F., Fernandez, J., Piñones Arce, P., Sumpter, D.: Seeing in to the future: using self-propelled particle models to aid player decision-making in soccer. In: In Proceedings of the 14th MIT Sloan Sports Analytics Conference (2020)
18. Pirotta, M., Restelli, M.: Inverse reinforcement learning through policy gradient minimization. In: AAAI (2016)
19. Puterman, M.L.: Markov decision processes: Discrete stochastic dynamic programming. In: John Wiley & Sons, Inc. (1994)
20. Tateo, D., Pirotta, M., Restelli, M., Bonarini, A.: Gradient-based minimization for multi-expert inverse reinforcement learning. In: 2017 IEEE Symposium Series on Computational Intelligence (SSCI). pp. 1–8 (2017)
21. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning **8**, 229–256 (1992)
22. Zaytseva, I., Shaposhnikov, D.: Moneyball in offensive vs defensive actions in soccer. European Economics: Labor and Social Conditions eJournal (2020)