Data-Driven Action Values for Professional Sports: A Brief Review

Guiliang Liu
Simon Fraser University
Email: gla68@sfu.ca

Abstract

The availability of large scale sports data has facilitated modeling more complex sports dynamic. It provides an increasing opportunity to evaluate in-game player actions with advanced machine learning techniques. This report provides a brief review of the state-of-the-art action evaluation models and explains how the scope of studied action expands from goal-related actions to all players' actions. We analyze their limitations and discuss the potential to improve the action-value metrics by (1) incorporating game context into evaluation (2) locating influential context features and (3) modeling the impact of an individual player under different game contexts. This report also introduces the available sports datasets including box scores, play-by-play data, and game tracking data.

I. Introduction

With the advancement of high-frequency optical tracking and object detection systems, more and larger datasets for sports matches have become available. Many data providers (e.g. Sportlogiq[1] and Opta[2]) have builds numerous datasets for sports game, including box-score, play-by-play and player tracking data. The datasets provide an increasing opportunity for large-scale machine learning to model complex sports dynamics, which enable us to provide a more accurate evaluation of the player performances.

Player evaluation is a major task for sports modeling that draws attention from both fans and team managers, who want to know which players to draft, sign or trade. Figure 1 summarize the player evaluation metrics proposed in the recent years. The most common metric has been to count goals or evaluate actions that are most related to scoring (e.g. shot). For example, a common metric Plus-Minus (+/-) qualifies the influence of a player's presence on the goal-scoring opportunity by awarding a player

[1] https://sportlogiq.com/en/

[2] https://www.optasports.com/

+1 if a goal is scored by the player's own team when the player is on the pitch. Another example is the Expected Goal (XG), which quantifies the value of a shot by the probability of the shot leading to a goal. Instead of measuring a player overall performance, these metrics, however, constraint the evaluation to a player's scoring influence and neglect their impact on the defensive end. To overcome this limitation, many recent works have proposed to rate players by evaluating all their actions, including both offensive actions (e.g. shot, assist) and defensive actions (e.g.block and tackle). Many recent works [Decroos et al., 2019; Schulte et al., 2017a; Routley and Schulte, 2015] have applied the impact metrics to evaluate players actions. For example, Scoring Impacts (SI) evaluates an action's influence on the probability of scoring the next goal. In this sense, a successful offensive action will offer a positive impact on the players' team while a successful defensive action will generate a negative impact on their opponent. By summarizing all the actions' impact over an entire season, SI provides a more comprehensive evaluation of the player performance. To derive the SI values, Schulte et al. [2017a] built a Markov game model and explicitly models the game environment. Compared to the traditional metrics, SI considers the influence of game context on action impacts, which provides a more accurate evaluation of player actions.

Previous results have demonstrated the benefit of incorporating the context features into action values. In this report, we analyze the state-the-art of action evaluation models and investigate several possible updates that can potentially improve the comprehensiveness and interpretability of the action values: 1) Instead of discretizing the context features, many recent works have applied function approximation techniques and trained a deep neural network that directly learns from the raw context features as well as the recent game history. Based on the deep model, one can apply an on-policy training method Sarsa [Sutton and Barto, 2018] to learn an action-value Q-function and estimate the actions' impact. 2) Many recent works show that despite the non-linear neural models have achieved promising performance in fitting complex context features, their black-box structure often prohibits understanding the influence of these features on action values. A method to overcome this limitation is mimic learning, which applies another transparent model (e.g. tree models) to mimic the knowledge of a deep model. We investigate the relevant works and show this method can be extended to the action value models and significantly improve our understanding of the context features. 3) We also investigate the methods of incorporating the features of individual players into action values. Unlike many previous works which represent each player by a static continuous-valued vector [Ganguly and Frank, 2018], we find a players' performance can be significantly affected by the game environment, and to model this impact, we propose the task of learning embeddings of individual players conditioning on the game context. The context-specific player embeddings can replace the one-hot player representation and facilitate many downstream applications such as expected goal or game outcome prediction.

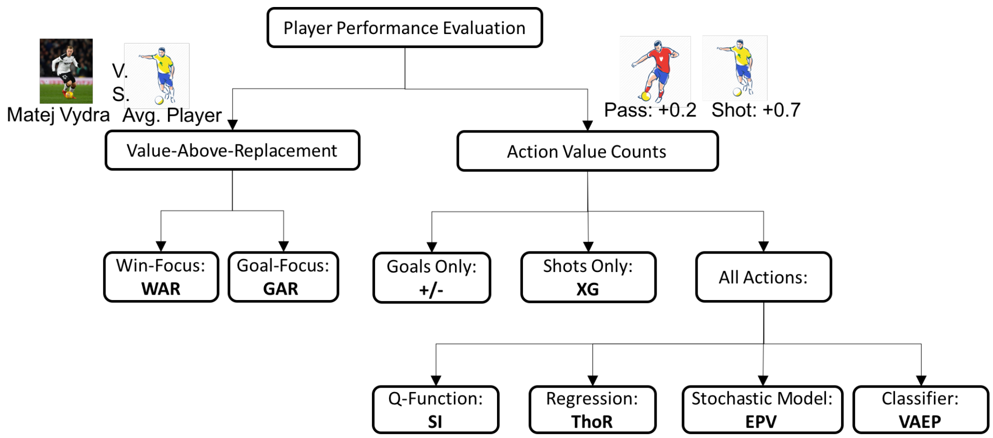

Fig. 1: A tree diagram to position our work in the research landscape. An important factor is whether a metric considers all actions or only a subset of them.

II. A Review of Players Evaluation Metrics

The most common application of the action values is evaluating players' performance. In this section, we provide a review of the action-value based player evaluation metrics by introducing the approach to (1) computing the actions values (2) applying the value to evaluate player performance.

Albert et al. [2017] provides several up-to-date survey articles on player evaluation. The player evaluation metrics are generally divided into two categories: Action Value Counts and Value-Above-Replacement:

A. Single-Action Value Counts

Probably one of the simplest approaches to qualifying a player's contribution is counting the number of goals that he or she manages to score during a sports game. Following this intuition, to evaluate a player's performance, traditional player evaluation metrics focus only on the actions that have an immediate impact on the scoring probabilities, for example shooting and passing. Among all the single-action evaluation metrics proposed in recent years, the most popular and well-studied metrics are Plus-Minus and Expecting Goal:

- *+/- (Plus-Minus)* is a commonly applied player evaluation metric that is almost applicable to every professional sport. It qualifies the influence of a player's presence on the goal-scoring opportunity for his team. The basic version awards a player +1 if a goal is scored by the player's own team when the player is on the pitch, and -1 if the other team scores. Some recent works modify the basic plus-minus metric. Instead of assigning equal importance to each goal, they weight the goals based on expected win probability, game time and game frequency [Schultze and Wellbrock, 2018], or with machine learning and survival models to estimate both expected goals and expected points to assess a player's defensive and offensive influence [Kharrat et al., 2017]. Macdonald [2011] also proposed an adjusted plus-minus statistic for NHL players by applying a weighted least squares regression to estimate an NHL player's effect on his team's success in scoring and preventing goals at even strength. The weighting approach incorporates the influence of team performance and game environment into the basic +/-, and thus provides a more comprehensive evaluation to a player's overall performance.

- *Expected Goals* (XG) quantifies the value of a shot by the probability of the shot leading to a goal. A common method to compute the XG values is training a goal prediction model with shot features (e.g. x,y coordinates or angle to goal) and assigning the soft outputs (expected goals) to players' shots. Players can be ranked by the summation of their total expected goals [Ali, 2011]. To study the effect of other actions, several recent works have extended the application of traditional XG and evaluated players by measuring the quality of their passes. Passing is one of the most strategic and frequent actions in many team sports, such as soccer and basketball. For each passing, Brooks et al. [2016] measured its value as the estimated probability of resulting in a successful shot. Bransen and Van Haaren [2018] measured its value as the difference between the goal-scoring probability before and after the pass.

B. *Multi-Actions Value Counts*

The above metrics evaluate the players by measuring only one type of action before a scoring event. However, to evaluate a player's overall performance, we should not only measure players' offensive influence (e.g. increasing their team's scoring chance) but also consider their defensive contribution (e.g. preventing their opponent scoring). To achieve it, several recent works have proposed to rate players by evaluating *all* their actions, including both offensive actions (e.g. shot, assist) and defensive actions (e.g. block and tackle):

- The *Expected Possession Value (EPV)* [Cervone et al., 2016] evaluates all the basketball players' actions within a continuous possession by estimating the expected number of points from the possession. They modeled the evolution of a basketball possession with a stochastic process model, which was implemented at multiple levels of resolution, differentiating between continuous, infinitesimal movements of players, and discrete events such as shot attempts and turnovers. Based on this framework, Fernández et al. [2019] built a deep model from the full resolution spatio-temporal data of 22 soccer players on the pitch and computed the EPVs for all actions during a game. They study the action impacts of individual soccer players under different game situations. Their models require tracking data that has the complete observability of all players. Many other play-by-play datasets, however, provide only partial observability of game context: they record only actions of the players who possess the ball at a given time.

- The Valuing Actions by Estimating Probabilities (VAEP) framework [Decroos et al., 2019] evaluates all on-the-ball actions of soccer players based on their influence on the game outcome. They also proposed a new language SPADL for describing individual player actions on the pitch. Applying SPADL, their model considers a set of hand-crafted action features from the recent game history and evaluates an action by the probability that it will lead to a goal within a constant number of future steps. To quantify a player's overall performance, their model computes the change in scoring and conceding probability after a player's action and summarizes the probabilities overall the entire season. Total Hockey Rating (THoR) [Schuckers and Curro, 2013] applied a similar method and they determined the impact of each play by the probability that it leads to a goal for a player's team (or their opponent) in the subsequent 20 seconds. Their rating that accounts for all on-the-ice players as well as the impact of where a shift starts and of every non-shooting events such as turnovers and hits that occur when a player is on the ice.
- Scoring Impact (SI) measures a player's performance by summarizing their impact over a game. The impact is the difference of scoring probabilities before and after the player controls the ball (or puck in ice hockey). To compute the scoring probabilities, Schulte et al. [2017a] formulated an ice hockey play-by-play dataset into a Markov model, where actions record the player movements and states capture the game context. They modeled event data of the form $s_0, a_0, r_1, s_1, a_1, \ldots, s_t, a_t$, $r_{t+1}, s_{t+1}, a_{t+1}$: environment state s_t occurs, an action a_t is chosen, resulting in a reward r_{t+1} and state s_{t+1}. At the next time step, another action a_{t+1} is chosen. The data are often separated into local *transitions* of the form $T\{s, a, r', s', a'\}$. Under this setting, Reinforcement Learning (RL) has been applied to evaluate the actions of players. The RL approach computes the expected scoring probabilities of player actions under different game context by a Q-function using dynamic programming [Puterman and Patrick, 2017] based on the Bellman equation:

$$Q(s,a) = \mathbb{E}_{s',a'}[r' + Q(s',a')|s,a] = \sum_{r'} \Pr(r'|s,a)r' + \sum_{s',a'} \Pr(s',a'|s,a)Q(s',a') \qquad (1)$$

This recurrence allows us to estimate the Q value at a current context s, a given an estimate for the next Q values and transition probabilities. To implement such a method, Schulte et al. [2017a]; Routley and Schulte [2015] , however, discretized location and time coordinates, and used maximum likelihood estimates for the resulting discrete transition probabilities. The discretization leads to loss of information and undesirable spatial-temporal discontinuities in the Q-function. Those drawbacks prohibit the model from generalizing to the unobserved part of state space.

C. Value-Above-Replacement

Value-Above-Replacement (VAR) evaluates players by measure how much they outperform the average player in their league. The most common VARs include *Goals/Wins Above Replacement (GAR/WAR)* which measure the player's contribution to his or her team by estimating the difference of team's scoring/winning chances when the target player plays, vs. replacing him or her with an average player. Although GAR and WAR are common player evaluation metrics for team sports like baseball, basketball, and American football, there has not been an agreed-upon definition for soccer [3] which has a more complex game context including a large football pitch over 7k square meters and 22 on-the-pitch players.

III. A Review of Sport Analytic Datasets

In this section, we investigate the available sports datasets, show some data examples, and discuss their applications for Sports Analytics.

a) Box Score Data: Box Score Data is commonly applied in team sports like baseball, basketball, football, and hockey. A box score is a structured summary of the results from professional sport games. The box score summarizes the game score as well as individual and team achievements in the game. Figure 2 and Figure 3 show the examples of box scores for players and teams respectively. Box score directly uses counting numbers to demonstrate teams' and player's performance in each game. Those numbers can be aggregated along an entire game season to analyze a team or a player's general performance. The main drawbacks of box scores are 1) instead of concluding a player's overall performance with one metric, box score often applies different numbers to represent a player's ability in different aspects (e.g., defensive and offensive performance). 2) Based on the score or action counting, the box score assigns the same value to action and neglects the influence of game contexts.

b) Play-by-Play Dataset: A Play-by-play dataset consists of the logs of discrete action events specifying various properties of the action (e.g. action type, acting player, time and location). The dataset tracks action events around the ball from the beginning until the end of a game. An action event often records the actions of on-the-ball players as well as the spatial and the temporal context features. Table I shows an example of the play-by-play soccer dataset, where each event (a line in Table I) records context

[3] https://www.americansocceranalysis.com/home/2019/1/11/points-above-replacement

PLAYER	TEAM	DATE	MATCHUP	W/L	MIN	PTS	FGM	FGA	FG%	3PM	3PA	3P%
Trae Young	ATL	03/01/2019	ATL vs. CHI	L	56	49	17	33	51.5	6	13	46.2
Damian Lillard	POR	03/18/2019	POR vs. IND	W	40	30	9	16	56.3	4	9	44.4
Bradley Beal	WAS	01/13/2019	WAS vs. TOR	L	55	43	17	36	47.2	6	12	50.0
James Harden	HOU	01/03/2019	HOU @ GSW	W	44	44	13	32	40.6	10	23	43.5
Bradley Beal	WAS	12/22/2018	WAS vs. PHX	W	54	40	17	33	51.5	4	12	33.3
De'Aaron Fox	SAC	11/01/2018	SAC @ ATL	W	34	31	9	13	69.2	3	4	75.0

Fig. 2: An example of box scores for NBA players. It summarizes the player performance of a game and reports the basketball statistics including playing Minutes (Min), points (PTS), field goals made/attempted/percentage (FGM/FGA/FG%) and three-point field goals made/attempted/percentage (3PM/3PA/3P%).

TEAM	DATE	MATCHUP	W/L	MIN	PTS	FGM	FGA	FG%	3PM	3PA	3P%
DEN	10/17/2019	DEN vs. POR	W	241	110	41	89	46.1	8	28	28.6
MEM	03/20/2019	MEM vs. HOU	W	265	126	41	89	46.1	13	37	35.1
HOU	03/08/2019	HOU vs. PHI	W	239	107	42	85	49.4	13	41	31.7
IND	02/23/2019	IND @ WAS	W	240	119	47	83	56.6	11	25	44.0
TOR	02/05/2019	TOR @ PHI	W	239	119	41	89	46.1	11	31	35.5
DEN	01/13/2019	DEN vs. POR	W	239	116	45	81	55.6	9	22	40.9

Fig. 3: An example of box scores for NBA teams. It summarizes the team performance of a game and reports the basketball statistics including playing Minutes (Min), points (PTS), field goals made/attempted/percentage (FGM/FGA/FG%) and three-point field goals made/attempted/percentage (3PM/3PA/3P%).

features including X-Y coordinate, manpower, score differential and game time remains. Compared to box score data, play-by-play data provides more detailed in-game information. The fine-grained data allows a more advanced machine learning algorithm to model the spatial and temporal features. For example, [Schulte et al., 2017b] built a Markov model where they used the context features (including manpower, score differential and game time) to define the states and computed the transition probabilities between those states with the frequency numbers in the play-by-play dataset. However, a major drawback of the play-by-play data is its partial observability. Instead of recording the actions and the locations of all the player on the court, the data often focus on the on-the-ball player (player who controls the ball). This issue becomes more serious for the team sport with a complex game context, for example, soccer. To alleviate the problem, [Liu and Schulte, 2018] included the play history into the states of the Markov model and applied an LSTM to capture the temporal features.

MP=Manpower, GD=Goal Difference, OC = Outcome, S=Succeed, F=Fail, H=Home, A=Away, T=Team who performs action, GTR = Game Time Remain, ED = Event Duration

GTR	X	Y	MP	GD	Action	OC	Velocity	ED	Angle	T	Reward
35m44s	87	26	Even	1	simple pass	S	(2.2, 1.7)	11.0	0.19	H	[0,0,0]
35m42s	90	17	Even	1	standard shot	F	(1.5, -4.5)	2.0	0.11	H	[0,0,0]
35m42s	99	44	Even	1	save	S	(0, 0)	0.0	0.06	A	[0,0,0]
35m9s	100	1	Even	1	cross	S	(0.0, -1.3)	33.0	0.0	H	[0,0,0]
35m7s	85	56	Even	1	simple pass	S	(-7.3, 27.6)	2.0	0.39	H	[0,0,0]
35m5s	92	67	Even	1	simple pass	S	(3.6, 5.4)	2.0	0.28	H	[0,0,0]
35m4s	97	50	Even	1	corner shot	S	(5.1, -16.2)	1.0	1.74	H	[0,0,0]
35m4s	100	50	Even	1	goal	S	(0, 0)	0.0	0.0	H	[1,0,0]
....	...	...	....	...	....	...	....	...	.....	.	......
3m41s	62	96	Even	2	long ball	F	(4.5, 9.3)	9.0	0.08	A	[0,0,0]
3m39s	19	89	Even	2	clearance	S	(-21.5, -3.2)	2.0	0.07	H	[0,0,0]
3m35s	24	100	Even	2	throw in	S	(1.3, 2.7)	4.0	0.09	A	[0,0,0]
3m33s	27	96	Even	2	simple pass	S	(1.1, -2.2)	2.0	0.1	A	[0,0,0]
3m31s	12	95	Even	2	cross	S	(-7.5, -0.5)	2.0	0.07	A	[0,0,0]
3m28s	6	46	Even	2	simple pass	S	(-1.7, -16.3)	3.0	0.79	A	[0,0,0]
3m26s	14	48	Even	2	standard shot	S	(3.8, 1.3)	2.0	0.44	A	[0,0,0]
3m26s	0	50	Even	2	goal	S	(0, 0)	0.0	0.0	A	[0,1,0]

TABLE I: An example of a play-by-play data sample featuring team scoring: a sequence of events where home team scores and then away team scores. The rewards [1,0,0] and [0,1,0] indicate the scoring event of the home team and away team respectively (see Section 4.1). We skip some events in the middle due to space issues.

c) Tracking Dataset: Compared to play-by-play data, tracking data records the location of each player at dense time intervals. Figure 4 shows an example of tracking the location of on-the-ball player from the broadcast video frames. The broadcast video, however, contains only a part of players on pitch, and thus provides only partial observability of the game. To overcome this limitation, SPORTLOGiQ [4] provides a multi-camera hardware solution. By installing multiple cameras around stadiums and extracting players' action data from the collected video with computer vision techniques (e.g. object detection [Ibrahim et al., 2016]), they achieve full observability of all 22 soccer player on the pitch. The

[4]https://sportlogiq.com/en/

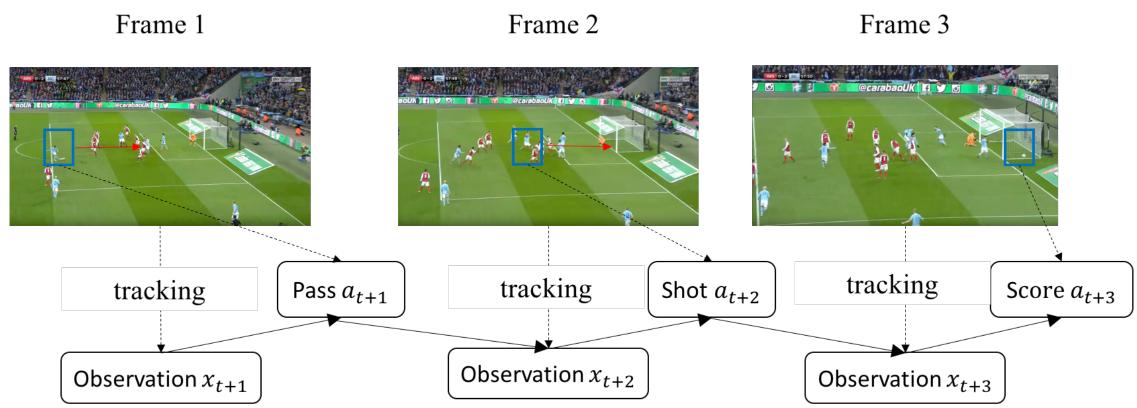

Fig. 4: An example of video tracking: at each video frame, the tracking system dynamically locates the position of on-the-ball player (in blue frame) with the object detection algorithm and records the player action. The system might require manually labelling the position of the ball (or the puck) for the sports with a large pitch (or rink).

tracking data fuels many models that requiring the full observability of a game at every time step (e.g. EPV).

IV. Challenge and Opportunities

The large-scale sport analytics datatsets provide an increasing opportunity for advanced machine learning algorithms to model sports dynamics and evaluate the players' performance. Although numerous player evaluation metrics have been proposed in recent years, those metrics, however, have some limitations in modeling the complex game context, explaining the influences of players' actions and incorporating playing style into the evaluation. In this section, we specify those limitations and introduce potential solutions to the problems.

A. *Modeling Game Context*

We introduce the motivation of including sports game context into the player evaluation and our approach to modeling them with a Markov model.

In a professional sports game, game context often describes the environment around a player. Table II shows a list of game context features for ice hockey. It is a common fact that game context can substantially influence the player performance, for example, the scoring opportunities will become larger as a player approaches his opponent's goal. Correspondingly, the evaluation model should assign different values to player actions under different game context.

Many previous player evaluation model [Routley and Schulte, 2015; Schulte et al., 2017a; Cervone et al., 2016] have considered the game context. Among them, [Schulte et al., 2017a] explicitly modeled

Type	Name	Range
Spatial Features	X Coordinate of Puck	[-100, 100]
	Y Coordinate of Puck	[-42.5, 42.5]
	Velocity of Puck	(-inf, +inf)
	Angle between the puck and the goal	$[-3.14, 3.14]$
Temporal Features	Game Time Remain	(-inf, 3600]
	Event Duration	(0, +inf)
Sports Game Features	Score Differential	(-inf, +inf)
	Manpower Situation	$\{$EV, SH, PP$\}$
	Home or Away Team	$\{$Home, Away$\}$
	Action Outcome	$\{$successful, failure$\}$

TABLE II: An example of the game context features for ice hockey.

the game context with the state of a Markov model. To build the Markov model, they discretized location and time coordinates and used maximum likelihood estimates to compute the resulting discrete transition probabilities. The discretization leads to the loss of information and undesirable spatial-temporal discontinuities in the action-value Q-function. The overcome the undesirable discontinuities, instead of explicitly modeling transitions in a discrete MDP, we propose to employ a *model-free approach* which learns Q values without explicitly estimating transition and reward probabilities [Sutton and Barto, 2018]. Many previous model-free Reinforcement Learning (RL) works [Mnih et al., 2015] have applied model-free learning with deep neural networks to capture continuous action and state features. These works mainly focused on *controlling* in continuous-flow games (e.g., Atari games). However, the real agents—players—in professional sports games are subject to evaluation, but not subject to control by an RL method. Instead of controlling the players, we propose to build a RL model for the play data and dynamically evaluate their actions under different contexts with the Sarsa [Sutton and Barto, 2018] method.

B. Understanding the Influence of Game Context on Action Values

A fundamental goal of sports statistics is to quantify how much physical player actions contribute to winning in what situation. Several very recent works [Le et al., 2017a; Wang et al., 2018] have built deep neural networks to model players' actions and value them under different situations. deep models support a more comprehensive evaluation because 1) Deep neural networks generalize well to different actions and complex game contexts and 2) Various network structures (e.g. LSTM) can be applied to model the current game context and its sequential game history. However, a neural network is an opaque black-box model. It prohibits understanding when or why the player's action is valuable, and which context features are the most influential for this assessment. A promising approach to overcome this limitation is *Mimic Learning*. Recent works [Ba and Caruana, 2014] have demonstrated that simple models can mimic the function of a deep neural network. Soft outputs of neural networks were collected by passing inputs to a large, complex and accurate deep neural network. Then they trained a mimic model with the same input and soft output as a supervisor. The results indicated that training a mimic model with soft output achieves substantial improvement in accuracy and efficiency over training the same model type directly with hard targets from the dataset. Applying the mimic learning, the mimic models distill the knowledge form the opaque model to an interpretable data structure.

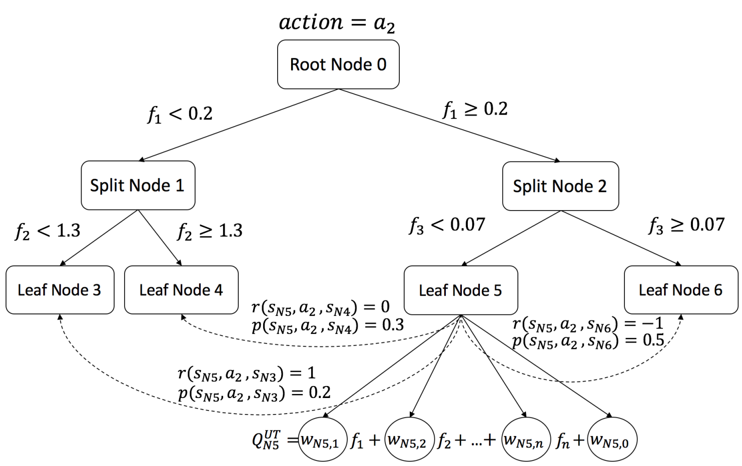

Fig. 5: An example of Linear Model U-Tree (LMUT).

TABLE III: Partition Cell

Node Name	Partition Cell
Leaf Node 3	$f_1 < 0.2$, $f_2 < 1.3$
Leaf Node 4	$f_1 < 0.2$, $f_2 \geq 1.3$
Leaf Node 5	$f_1 \geq 0.2$, $f_2 < 0.07$
Leaf Node 6	$f_1 \geq 0.2$, $f_2 \geq 0.07$

We also investigate the available implementations for the mimic model. The most common implementation is shallow feed-forward neural network or decision trees [Che et al., 2016; Dancey et al., 2007]. Those models, however, have two major drawbacks: 1) they have only limited generalization ability to the complex sports game context. 2) We model the sports game context with a Markov Decision Process and train a deep RL model to compute the Q functions. The learning process is different from the classic supervised learning setting. To tackle the problem, we refer to U-tree [McCallum et al., 1996; Uther and Veloso, 1998], which a classic online reinforcement learning method that represents a Q function using a tree structure. To strengthen its generalization ability of tree structure, we propose to add a linear model to each leaf node, which defines a novel Linear Model U-Tree (LMUT). As shown in Figure 5 and Table III, each leaf node of a LMUT defines a partition cell of the input space, which can be interpreted as a discrete state s for the decision process. Within each partition cell, LMUT also records the reward r and the transition probabilities p of performing action a on the current state s, as shown in the Leaf Node 5 of Figure 5. So LMUT learning builds a Markov Decision Process (MDP) from the interaction data between environment and deep model. Compared to a linear Q-function approximator [Sutton and Barto, 2018], an LMUT defines an ensemble of linear Q-functions, one for each partition cell. Since each Q-value prediction Q_N^{UT} comes from a single linear model, the prediction can be explained by the feature weights of the model.

C. Modeling Players Impact on Actions Values

Many recent works Liu and Schulte [2018]; Decroos et al. [2019]; Fernández et al. [2019] have proposed to estimate the expected team success following a player's actions. These expected values support many downstream applications, such as predicting game outcomes or evaluating player performance. However, when estimating the expected values, previous works often overlook the player-specific features (e.g. scoring ability) and assign the same values to actions performed by different players. Neglecting differences among individual players compromise the model performance.

Some previous works have explored the approach to incorporating player information into modeling. Probably the most straightforward approach is to apply a one-hot vector recording the player identity (pid) and train the neural model to dynamically learn the correlations between pids and game context Le et al. [2017b]. Despite its simplicity, the one-hot representation is not informative enough for a neural network to adequately model the correlation between individuals and success. As evidence, our experiment shows very limited improvement when we directly complement the input space with pids. A recent work Ganguly and Frank [2018] proposed to learn player embeddings by training a neural network encoder to perform a secondary prediction task: given the current game context, predict the pids of all on-court players. They extracted the middle layer from the trained encoder and used it as a player embedding to facilitate the training of other primary tasks. However, the predictive accuracy on the secondary task was low. A problem with training a neural net as a deterministic regression model is that the player presence has a multi-modal distribution with several almost equally likely outcomes.

To represent the multi-modality of player distribution, we propose to apply variational auto-encoders which can produce a distribution over outcomes that accommodates multiple modes. Variational Auto-Encoder (VAE) has achieved promising performance in recovering multimodal distributions and generating many kinds of complicated data, including handwritten faces [Kingma and Welling, 2013], images [Gregor et al., 2015] and players actions [Mehrasa et al., 2019]. VAE applies a set of latent variables $\mathbf{z}$ to capture the variations of observed variables $\boldsymbol{o}$. During the generative process, the prior of $\mathbf{z}$ is generally chosen to be a simple Gaussian distribution. VAE models the likelihood function $p(\boldsymbol{o}|\mathbf{z})$ with a decoder (usually implemented as a Gaussian or Bernoulli Multi-Layer Perception (MLP) [Kingma and Welling, 2013], which applies a highly non-linear mapping from $\mathbf{z}$ to $\boldsymbol{o}$.

The non-linearity in complicated likelihood function $p(\boldsymbol{o}|\mathbf{z})$ leads to the intractable inference of the posterior $p(\mathbf{z}|\boldsymbol{o})$. Instead, VAE approximates the true posterior with a recognition model (decoder) $q(\mathbf{z}|\boldsymbol{o})$, which is usually defined as a Gaussian as $\mathbf{z} \sim \mathcal{N}[\boldsymbol{\mu}, diag(\boldsymbol{\sigma}^2)]$ ($\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are computed with observed variables $\boldsymbol{o}$). Parameters of both decoder and encoder are optimized by maximizing a lower bound of the marginal likelihood of observation $p(x)$:

$$\mathcal{L}(p(\boldsymbol{o})) = -KL(q(\mathbf{z}|\boldsymbol{o})||p(\mathbf{z})) + \mathbb{E}_{q(\mathbf{z}|\boldsymbol{o})}\Big[\log p(\boldsymbol{o}|\mathbf{z})\Big] \tag{2}$$

Kingma and Welling [2013] introduced an alternative method for generating samples from $q(\mathbf{z}|pl)$ and described a reparameterizing trick for VAE. By rewriting:

$$\mathbb{E}\Big[\log(p(\boldsymbol{o}|\mathbf{z}))\Big] = \mathbb{E}\Big[\log p(\boldsymbol{o}|\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon})\Big] \tag{3}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, 1)$, reparameterizing makes the estimations of the expectation w.r.t. $q(\mathbf{z}|\boldsymbol{o})$ differentiable.

To handle sequential data, Chung et al. [2015] combined the latent variables with a recurrent model. The proposed Variational Recurrent Neural Network (VRNN) includes a VAE at every time step t. The object function is a timestep-wise variational lower bound:

$$\sum_{t=1}^{T}\Big[-KL(q(\mathbf{z}_t|\boldsymbol{o}_{\leq t}, \mathbf{z}_{<t})||p(\mathbf{z}_t|\boldsymbol{o}_{<t}, \mathbf{z}_{<t})) + \log p(\boldsymbol{o}_t|\mathbf{z}_{\leq t}, \boldsymbol{o}_{<t})\Big] \tag{4}$$

V. Conclusion

This report provides a brief review over the previous player evaluation metrics and explains how these metrics evolve from single-action evaluation to multiple-actions evaluation. We also introduce three kinds of commonly applied sports datasets including Box score, play-by-play dataset and game tracking dataset, analyze their connections, and review the approaching of developing a metric with these datasets. Based on the previous results, in this report, we also present the potential to improve the metric performance by (1) incorporating environment information into evaluation (2) finding influential environment features and (3) modeling the impact of an individual player under different game contexts. For future work, we will continuously explore these potentials and propose promising models to achieve a more reliable evaluation metric.

REFERENCES

Albert, J., Glickman, M. E., Swartz, T. B., and Koning, R. H. (2017). *Handbook of Statistical Methods and Analyses in Sports*. CRC Press.

Ali, A. (2011). Measuring soccer skill performance: a review. *Scandinavian journal of medicine & science in sports*, 21(2):170–183.

Ba, J. and Caruana, R. (2014). Do deep nets really need to be deep? In *Advances in neural information processing systems*, pages 2654–2662.

Bransen, L. and Van Haaren, J. (2018). Measuring football players' on-the-ball contributions from passes during games. In *MLSA-ECML Workshop*, pages 3–15. Springer.

Brooks, J., Kerr, M., and Guttag, J. (2016). Developing a data-driven player ranking in soccer using predictive model weights. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 49–55. ACM.

Cervone, D., D'Amour, A., Bornn, L., and Goldsberry, K. (2016). A multiresolution stochastic process model for predicting basketball possession outcomes. *Journal of the American Statistical Association*, 111(514):585–599.

Che, Z. et al. (2016). Interpretable deep models for ICU outcome prediction. In *AMIA Annual Symposium Proceedings*, volume 2016, page 371. AMIA.

Chung, J., Kastner, K., Dinh, L., Goel, K., Courville, A. C., and Bengio, Y. (2015). A recurrent latent variable model for sequential data. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 2980–2988.

Dancey, D., Bandar, Z. A., and McLean, D. (2007). Logistic model tree extraction from artificial neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(4):794–802.

Decroos, T., Bransen, L., Haaren, J. V., and Davis, J. (2019). Actions speak louder than goals: Valuing player actions in soccer. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019.*, pages 1851–1861.

Fernández, J., Barcelona, F., Bornn, L., and Cervone, D. (2019). Decomposing the immeasurable sport: A deep learning expected possession value framework for soccer.

Ganguly, S. and Frank, N. (2018). The problem with win probability. In *Proceedings of the 12th MIT Sloan Sports Analytics Conference. Boston.*

Gregor, K., Danihelka, I., Graves, A., Rezende, D., and Wierstra, D. (2015). Draw: A recurrent neural network for image generation. In *International Conference on Machine Learning*, pages 1462–1471.

Ibrahim, M., Muralidharan, S., Deng, Z., Vahdat, A., and Mori, G. (2016). A hierarchical deep temporal model for group activity recognition. In *Computer Vision and Pattern Recognition (CVPR)*.

Kharrat, T., Pena, J. L., and McHale, I. (2017). Plus-minus player ratings for soccer. *arXiv preprint arXiv:1706.04943*.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Le, H. M., Carr, P., Yue, Y., and Lucey, P. (2017a). Data-driven ghosting using deep imitation learning. In *MIT sloan sports analytics conference*.

Le, H. M., Carr, P., Yue, Y., and Lucey, P. (2017b). Data-driven ghosting using deep imitation learning. In *MIT Sloan Sports Analytics Conference*.

Liu, G. and Schulte, O. (2018). 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*, pages 3442–3448. International Joint Conferences on Artificial Intelligence Organization.

Macdonald, B. (2011). A regression-based adjusted plus-minus statistic for nhl players. *Journal of Quantitative Analysis in Sports*, 7(3):29.

McCallum, A. K. et al. (1996). Learning to use selective attention and short-term memory in sequential tasks. In *Proceedings SAB*, volume 4, pages 315–325.

Mehrasa, N., Jyothi, A. A., Durand, T., He, J., Sigal, L., and Mori, G. (2019). A variational auto-encoder model for stochastic point processes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3165–3174.

Mnih, V., Kavukcuoglu, K., Silver, D., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533.

Puterman, M. L. and Patrick, J. (2017). Dynamic programming. In *Encyclopedia of Machine Learning and Data Mining*, pages 377–388.

Routley, K. and Schulte, O. (2015). A markov game model for valuing player actions in ice hockey. In *Uncertainty in Artificial Intelligence (UAI)*, pages 782–791.

Schuckers, M. and Curro, J. (2013). Total hockey rating (thor): A comprehensive statistical rating of national hockey league forwards and defensemen based upon all on-ice events. In *MIT sloan Sports Analytics Conference*.

Schulte, O., Khademi, M., Gholami, S., Zhao, Z., Javan, M., and Desaulniers, P. (2017a). A markov game model for valuing actions, locations, and team performance in ice hockey. *Data Mining and Knowledge Discovery*, pages 1–23.

Schulte, O., Zhao, Z., Javan, M., and Desaulniers, P. (2017b). Apples-to-apples: Clustering and ranking nhl players using location information and scoring impact. In *Proceedings MIT Sloan Sports Analytics Conference*.

Schultze, S. R. and Wellbrock, C.-M. (2018). A weighted plus/minus metric for individual soccer player performance. *Journal of Sports Analytics*, 4(2):121–131.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Uther, W. T. and Veloso, M. M. (1998). Tree based discretization for continuous state space reinforcement learning. In *AAAI/IAAI*, pages 769–774.

Wang, J., Fox, I., Skaza, J., Linck, N., Singh, S., and Wiens, J. (2018). The advantage of doubling: A deep reinforcement learning approach to studying the double team in the NBA. In *MIT sloan sports analytics conference*.