Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)

Predicting Soccer Highlights from Spatio-Temporal Match Event Streams

Tom Decroos, Vladimir Dzyuba, Jan Van Haaren, Jesse Davis
KU Leuven, Department of Computer Science, 3001 Leuven, Belgium
{tom.decroos, vladimir.dzyuba, jan.vanhaaren, jesse.davis}@cs.kuleuven.be

Abstract

Sports broadcasters are continuously seeking to make their live coverages of soccer matches more attractive. A recent innovation is the "highlight channel," which shows the most interesting events from multiple matches played at the same time. However, switching between matches at the right time is challenging in fast-paced sports like soccer, where interesting situations often evolve as quickly as they disappear again. This paper presents the POGBA algorithm for automatically predicting highlights in soccer matches, which is an important task that has not yet been addressed. POGBA leverages spatio-temporal event streams collected during matches to predict the probability that a particular game state will lead to a goal. An empirical evaluation on a real-world dataset shows that POGBA outperforms the baseline algorithms in terms of both precision and recall.

Introduction

Recent technological advances have enabled the large-scale collection of soccer data. Companies like Prozone[1] and Opta[2] rely on optical tracking systems and human annotators to gather high volumes of data during matches in major soccer competitions such as the Champions League. The tracking systems automatically record the locations of the players and the ball at a high frequency, while teams of annotators enrich the dataset by annotating the notable events that happen on the pitch. The data are collected live allowing broadcasters to show interesting statistics and visualizations.

With the increasing competition from mobile sports apps and social media, sports broadcasters are continuously seeking to improve their live coverages. A recent innovation is the "highlight channel", which shows live action from multiple matches played simultaneously. FOX Sports' *Multi-Match 90*-channel, for instance, switches from one Champions League match to another depending on which match is currently the most interesting to watch.

The challenge for broadcasters is to switch between matches at the right time. This task is especially challenging in fast-paced sports like soccer, where interesting game states often emerge and disappear extremely quickly. Typically, highlight channels only switch to a match after a goal has been scored, but ideally they would switch several seconds earlier. For that to happen, sports broadcasters need a predictive model telling them when an interesting game situation is about to arise.

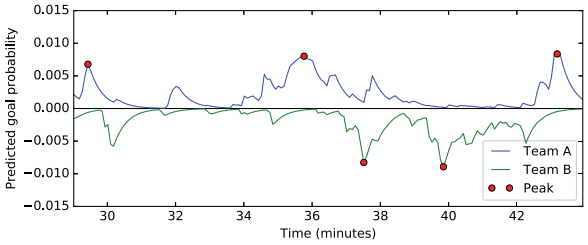

Figure 1: Our proposed POGBA algorithm predicts highlights in soccer matches from spatio-temporal match data.

This paper addresses the novel but important task of predicting the probability that a game situation will lead to a goal in the near future. Although this task could most naturally be modeled as a conditional probability estimation problem, several factors make this a challenging problem. First, very few goals are scored in a soccer match. Second, no two game states are identical as players can freely move around the pitch. Third, a game can evolve into many different ways depending on the actions of the players.

To address this task, we propose the POGBA (**P**rediction **o**f **G**oals **b**y **A**ssessing Phases) algorithm for automatically predicting highlights in soccer matches from spatio-temporal match data. More specifically, our algorithm predicts the probability that a given game state will lead to a goal. Instead of directly modeling the conditional probability, POGBA exploits the insight that goals are preceded by goal attempts, which are much more frequent than goals. Viewing the available spatio-temporal data as one long stream of events, POGBA first estimates the probability that a game state will lead to an attempt, and then estimates the probability that the attempt will lead to a goal. An empirical evaluation for predicting highlights on a real-world dataset comprising 69 soccer matches shows that POGBA outperforms several baseline approaches in terms of F_1 score.

[1] http://prozonesports.stats.com

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

Preliminaries

We introduce the notation used in this paper and provide background on soccer, the data, and dynamic time warping.

Notation

We use upper case to denote sets of objects, lower case to denote individual objects, e.g., $x \in X$, and boldface to denote random variables and sets thereof, e.g., $\mathbf{x} \in \mathbf{X}$.

Let $E = [\ldots e_{i-1}, e_i, e_{i+1} \ldots]$ denote a discrete stream of events and e_i denote an individual event. Each event is a tuple (l, t, a), where l is the event's type or label, t a timestamp, and a a tuple of event attributes. We use $l(e)$ and $t(e)$ to refer to an event's label and timestamp, respectively. The set of possible attributes varies for different event types. Furthermore, let L denote the finite set of possible event types and $l^* \in L$ a special event type called *critical*. Events of this type are of particular interest in a given application. We will use e^* to refer to an event e where $l(e) = l^*$.

Given a timestamp t and a user-defined time window length ω (e.g., 10 seconds), we use $E_{-\omega}$ to denote the subsequence of events in the previous window, i.e., $[e_{t-\omega}, \ldots, e_t]$. Similarly, $E_{+\omega}$ denotes the subsequence of events in the following window, i.e., $[e_{t+1}, \ldots, e_{t+\omega}]$. Note that the *number* of events in a window varies depending on the properties of the stream. The problem addressed in this paper requires making inferences about $E_{+\omega}$ given $E_{-\omega}$.

Soccer

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

Dataset

The dataset consists of play-by-play data for 69 soccer matches from a Belgian professional soccer club. The data for each match consist of a *match sheet* with details on the players and managers, an *event stream*, and *tracking information* for the players as well as the ball. However, the tracking information is only available for thirteen matches.

For each match, the event stream contains around 2,600 events of over 40 different types. Besides passes between players, the most frequent events include players running with the ball, receiving the ball, shooting towards goal, fouling other players, crossing the ball, and clearing the ball.

For each event, the type, the players involved, a timestamp, and the start and end location are known. Depending on the type, additional information can be available such as the body part and the type of play (i.e., open or set play).

Dynamic Time Warping

Dynamic time warping (DTW) is a state-of-the-art distance measure for time-dependent sequences (Müller 2007). Unlike basic Euclidean distance, DTW does not require that sequences have the same length and is insensitive to minor mismatches between sequences, such as delays or shifts. Intuitively, the sequences are "warped" in a nonlinear fashion to match each other. Given two univariate numeric sequences $E = [e_1, \ldots, e_M]$ and $F = [f_1, \ldots, f_N]$, DTW typically employs a dynamic programming approach to evaluate the cost of all possible alignments. The DTW distance is then the cost associated with the best (i.e., lowest cost) alignment. Often, constraints are incorporated into the calculation to limit how much warping can occur.

POGBA Algorithm

We address the following problem:

Given: A subsequence of an event stream $E_{-\omega}$, where ω is a user-defined window length.

Predict: The probability that a critical event e^* occurs in the next ω seconds.

We focus on predicting goals. This problem presents several challenges. First, this problem can naturally be viewed as estimating the conditional probability $P(\exists e^* \in E_{+\omega} \mid E_{-\omega})$. However, goals occur very infrequently in soccer and a huge number of game situations can lead to a goal. Hence, there may be insufficient data to learn an accurate conditional distribution. Consequently, we will model this conditional distribution indirectly by using a generative model. Second, the problem requires projecting how an event stream will evolve over multiple time steps. We model the evolution of a sequence by using a nearest-neighbor-based scheme.

The remainder of this section is organized as follows. First, we explain POGBA's four key elements: (i) defining the generative model, (ii) preprocessing the event stream, (iii) estimating the probabilities, and (iv) making predictions. Next, we discuss its efficiency. Finally, we explain how to predict highlights in soccer matches using POGBA.

Generative Probabilistic Model

We first identify a set of (application-specific) event types $L^{\mathcal{C}} \subset L$ called *preconditions*, which satisfy two criteria. One, these events should occur much more frequently than a critical event. Two, the probability that a critical event is preceded by a precondition is arbitrarily close to 1.

$$P(\exists e^{\mathcal{C}} \in E \mid e^* \in E,\ t(e^*) > t(e^{\mathcal{C}})) \approx 1$$

where $e^{\mathcal{C}}$ denotes an event such that $l(e^{\mathcal{C}}) \in L^{\mathcal{C}}$. Essentially, if we observe a critical event, it was almost certainly preceded by a preconditional event. In soccer, goal attempts (e.g., headers, shots, penalties, etc.) are preconditional events. Attempts occur much more frequently than goals and a goal is almost always preceded by an attempt.

Based on the above insight, we approximate the desired conditional distribution with the following joint distribution:

$$P(e^*, e^{\mathcal{C}}, E_{-\omega}) = P(e^* \mid e^{\mathcal{C}}) \times P(e^{\mathcal{C}} \in E_{+\omega} \mid E_{-\omega}) \times P(E_{-\omega}) \quad (1)$$

where $P(e^* \mid e^{\mathcal{C}})$ captures the probability that an attempt results in a goal, $P(e^{\mathcal{C}} \in E_{+\omega} \mid E_{-\omega})$ is the probability

of observing a goal attempt in the next ω seconds given the observed event stream $E_{-\omega}$, and $P(E_{-\omega})$ is the probability of observing the input sequence.

Stream Preprocessing

The training data are subsequences of events, which we obtain as follows. For each match, we divide the event stream into subsequences of length ω seconds. We allow an overlap of τ seconds and thus let a new window start every $\omega - \tau$ seconds. While this is a straightforward procedure, we need to account for the following two domain-specific issues.

The first issue is that the time between two consecutive events greatly differs from one match to another due to a difference in intensity and the unreliability of human annotators. We account for this by generating a virtual event every half a second. We set the features of these virtual events based on the features of the preceding and succeeding events. Real-valued features (e.g., spatial location) are set via linear interpolation. Categorical features take the value of the preceding event. Intuitively, the previous event (e.g., a pass, shot or dribble) can be viewed as being "in progress" if no new event has been recorded.

The second issue is that the attacking team is not always playing in the same direction. More specifically, the attacking team plays from left to right in some of the subsequences and from right to left in others. We account for this by normalizing the subsequences such that the dominant team, which is the team possessing the ball in the majority of the events, is always playing in the same direction.

Estimating Parameters

The prediction by Equation 6 is agnostic to which team may score. Since we want to measure how good or bad the current game situation is for each team, we have to compute the probability that each team will score at any given time point.

Estimating $P(e^* \mid e^{\mathcal{C}})$. For our soccer task, estimating this probability requires predicting $P(\text{goal} \mid \text{attempt})$, which we do by training a probabilistic classifier. We evaluate a variety of different classifiers in the empirical evaluation.

We construct one example for each occurrence of a preconditional event in the training set. There are three possible labels for each example: the dominating team has an attempt and scores, the non-dominating team has an attempt and scores, and either team has an attempt but there is no goal. Each example is described by features such as the distance to goal, shot angle, number of passes in the last ω seconds, average ball speed in the last ω seconds, and average ball angle in the last ω seconds. These features are commonly used for this task (Ijtsma 2015; Eastwood 2015; Caley 2015; Lucey et al. 2014).

Estimating $P(e^{\mathcal{C}} \in E_{+\omega} \mid E_{-\omega})$. Estimating this probability requires projecting how the current event sequence will evolve over the next ω seconds, as this will influence whether a goal attempt is likely or not. Given the current subsequence $E_{-\omega}$, we employ a weighted k-nearest-neighbor approach and search the training set to find similar sequences to $E_{-\omega}$. Then, for each retrieved neighbor E', we use $E'_{+\omega}$ (i.e., the events that occur in the next ω seconds after E') as a projection of how $E_{-\omega}$ will evolve.

To measure similarity between $E_{-\omega}$ and a training event subsequence E', we use a multivariate variant of DTW:

$$d(E_{-\omega}, E') = \sqrt{\sum_{i=1}^{n} DTW_i(E_{-\omega}, E')^2} \tag{2}$$

where DTW_i is the DTW-based similarity of $E_{-\omega}$ and E' in the i^{th} dimension. We consider $n = 2$ dimensions: the x and y coordinates of the event's location.

After identifying the k nearest neighbors, the probability is computed as

$$P(e^{\mathcal{C}} \mid E_{-\omega}) = \frac{\sum_{E' \in NN(E_{-\omega})} \frac{1}{d(E_{-\omega}, E')^2} \mathbb{1}_{dom}(E')}{Z} \tag{3}$$

where $\mathbb{1}_{dom}(E')$ is an indicator function that is 1 if a preconditional event for the dominating team occurred in the ω seconds following E' and 0 otherwise, and Z is a normalization constant defined as:

$$Z = \sum_{E' \in NN(E_{-\omega})} \frac{1}{d(E_{-\omega}, E')^2}. \tag{4}$$

The probability of a preconditional event for the non-dominating team can be computed in an analogous manner by changing the indicator function in the appropriate way.

Making Predictions

Finding potential game situations of interest requires performing marginal inference in the generative model defined by Equation 1 to estimate the probability of the critical event occurring in the near future given the current game situation. Given the previous window, we marginalize over all possible preconditions as follows:

$$P(e^* \in E_{+\omega} \mid E_{-\omega}) = \sum_{e^{\mathcal{C}}} \Big(P(e^{\mathcal{C}} \in E_{+\omega} \mid E_{-\omega}) \times P(e^* \mid e^{\mathcal{C}}) \Big) \tag{5}$$

The first term $P(e^{\mathcal{C}} \in E_{+\omega} \mid E_{-\omega})$ represents the probability of observing a preconditional event in the next ω seconds. Computing the second term requires constructing features based on the next ω seconds of the event stream. Clearly, we do not know how the current event stream will evolve. Hence, the nearest-neighbors mechanism for computing $P(e^{\mathcal{C}} \in E_{+\omega})$ occurs during inference to project possible evolutions of the current stream. Thus, the marginalization can be written as:

$$P(e^* \in E_{+\omega} \mid E_{-\omega}) = \sum_{E' \in NN(E_{-\omega})} \frac{\frac{1}{d(E_{-\omega}, E')^2} \mathbb{1}_{dom}(E')}{Z} \times P(e^* \mid e^{\mathcal{C}}) \tag{6}$$

where $\mathbb{1}_{dom}(E')$ and Z are defined in the Estimating Parameters subsection. The features used to make the prediction $P(e^* \mid e^{\mathcal{C}})$ are constructed based on $E'_{+\omega}$ (i.e., the observed ω second window following E' in the training data). Algorithm 1 provides pseudocode for the full prediction pipeline.

Algorithm 1 Making predictions

Parameters: Set of event sequences $\{E_i\}$, window length ω, window overlap τ, number of neighbors k

Subsequences $S = \bigcup \text{SPLIT}(E_i, \omega, \tau)$
$P^* = \text{CRITICALEVENTMODEL}(\{s \in S \mid e^{\mathcal{C}} \in s\})$
function PREDICT($E_{-\omega}$)
  $NN = \text{NEARESTSEQUENCES}(E_{-\omega}, k, S)$
  $Z = \sum_{E' \in NN} d(E_{-\omega}, E')^{-2}$
  $p = 0$
  for $E' \in NN$ **do**
    if $e^{\mathcal{C}} \in E'$ **then**
      $p \leftarrow p + P^*(e^{\mathcal{C}}) \times d(E_{-\omega}, s)^{-2}/Z$
  return p

Efficiency

The prediction time for a subsequence is dominated by identifying its nearest neighbors (i.e., the NEARESTSEQUENCES method in Algorithm 1). As DTW has a relatively high computational cost, performing an exhaustive search to identify the nearest neighbors is undesirable. Unfortunately, traditional index trees are unsuitable for multivariate streaming data. Consequently, we use a vantage-point (VP) tree (Yianilos 1993), which achieves efficient ordering using an application-specific distance metric. Using a VP tree, identifying nearest neighbors scales logarithmically in the number of training examples. VP trees do not require a metric based on a coordinate form, but do require a distance metric, and DTW does not satisfy the triangle inequality (Müller 2007). However, empirical evidence strongly suggests that DTW almost always satisfies the triangle inequality on real-world data (Vidal et al. 1988). Thus, we feel that the run-time improvements of using a VP tree outweigh the small risk of potentially missing a nearest neighbor.

Predicting Highlights Using POGBA

For a given match, we identify highlights in three steps. First, we construct a time series by building one ω-second window starting at each second of the match. For each window, we compute Equation 6, which yields a time series $P = [p_1, ..., p_n]$, where p_i is the predicted probability at time step i and n is the number of windows in the match.

Second, we smooth the predicted probabilities using *exponential smoothing*. For each probability p_i, the smoothed probability $s_i = \alpha \cdot p_i + (1 - \alpha) \cdot s_{i-1}$, where α is a user-specified parameter and $s_0 = p_0$. Hence, we obtain a smoothed time series $S = [s_1, ..., s_n]$. Figure 1 shows the smoothed probabilities for 15 minutes of gameplay.

Third, we detect peaks in the time series S. We consider a smoothed probability s_i to be a peak if the following three conditions hold:

1. s_i is larger than its immediate neighbors s_{i-1} and s_{i+1};
2. s_i is larger than the mean of all smoothed probabilities;
3. s_i is the largest peak within the interval $[i - w, i + w]$ where w is a user-defined window size.

We predict each peak to be a highlight.

Experiments

First, we evaluate POGBA's performance on the core highlight prediction task and then present several auxiliary experiments that provide additional information about its components. All experiments use the dataset from the Preliminaries section, which is described in more detail below.

Highlight Prediction (Main Experiment)

When evaluating POGBA's performance for highlight prediction in soccer matches, we aim to show the benefits of its two core components.

1. Considering full spatio-temporal data (as opposed to static spatial-only snapshots).
2. *Indirectly* estimating the conditional probability of critical events (Equation 6).

To our knowledge, the specific task that we tackle has not been addressed before. Hence, we devised strong baseline algorithms based on the core components of POGBA, related tasks, and domain knowledge. More specifically, we compare the following approaches:

POGBA: Our proposed approach, which considers the full spatio-temporal data and performs the indirect probability estimation.

SpatDir: A baseline that only considers the *spatial* data, i.e., it ignores an event sequence's temporal evolution and performs *direct* conditional probability estimation.

SpatIndir: A baseline that only considers *spatial* data, but uses the same *indirect* conditional probability estimation technique as POGBA. This baseline is based on an expected-goals model, which is the most advanced metric for quantifying goal attempts in the soccer analytics community (Ijtsma 2015; Lucey et al. 2014).

SpatTempDir: A baseline that considers full spatio-temporal data and performs *direct* probability estimation. This baseline is a state-of-the-art technique for time series classification (Wang et al. 2013; Bagnall and Lines 2014).

Final 4th: A deterministic baseline that predicts a highlight for the dominating team if, in a given minute, it spent at least ω seconds in the final quarter of the pitch, i.e., close to the opponent's goal (see Figure 2).

Random: A baseline that predicts timestamps and teams for highlights uniformly at random.

To construct the spatial-only baselines, we divide the soccer pitch into 20 zones, as shown in Figure 2. In the prediction phase, each static approach (i.e., Spatdir and SpatIndir) discards all events in the window $E_{-\omega}$, except for the last one e_{t-1}; determines the pitch zone in which it occurred; and returns the probability estimate for this zone. The baselines differ in the way they estimate the goal probability for the given zone:

$$P_z{}^* = P(e^* \in E_{+\omega} \mid \mathbf{zone}(\mathbf{e_{t-1}}) = z)$$

1. *SpatDir: Direct goal probability estimation*

$$P_z{}^* = \frac{1 + \#\text{Goals within } \omega\text{s from an event in } z}{1000 + \#\text{Events in } z}$$

Time, s	Event		**X**	**Y**	Team	Player	Windows
225.1	Running with ball	+3 virtual events↓	3560	2100	a	q	
227.0	Pass	+2 virtual events↓	3560	2100	a	q	
228.8	**Shot on target** ($e^{\mathcal{C}}$)		4960	-810	a	r	
229.1	Diving save		5240	-450	b	s	
229.6	Out for corner	+56 virtual events↓	5250	3200	$-$	$-$	
258.1	Cross	+3 virtual events↓	5225	3375	a	t	
259.9	Clearance	+1 virtual event↓	4400	-50	b	u	
261.2	Pass		4140	950	a	v	
261.3	Reception		4190	1060	b	w	

Table 1: An anonymized excerpt of raw sequence data, showing several available event attributes. Overlapping windows contain a varying number of non-virtual events. In this paper, we only use the x- and y-coordinates for inference.

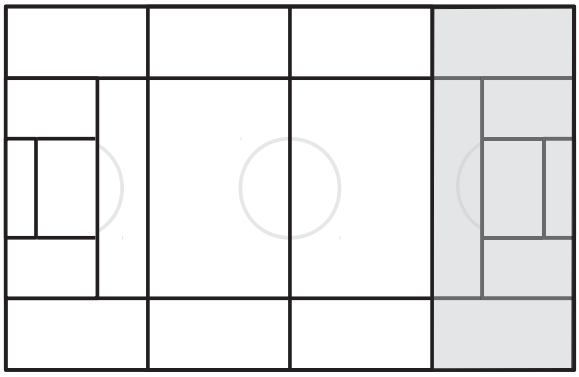

Figure 2: Division of the pitch into 20 zones for the baseline approaches. The direction of play is from left to right.

2. *SpatIndir: Indirect goal probability estimation*

$$P_z^* = \frac{1 + \sum_{\text{Attempts within } \omega\text{s from an event in } z} P(\text{goal} \mid \text{attempt})}{1000 + \#\text{ Events in } z}$$

where $1/1000$ is the prior goal probability. Each zone's conditional probability estimate is computed from the data.

Ground Truth

Soccer highlights are a subjective concept. A goalless 0:0 draw can have more highlights than a run-of-the-mill 2:1 match. Therefore, instead of limiting ourselves to goals as highlights, we construct the ground truth from judgments of professional journalists. For 25 matches, we manually select highlights from live text commentary downloaded from the leading Belgian sports website. The following is an example message corresponding to the data excerpt shown in Table 1:

> 49': Excellent attack from [Team a] via [Player q] who launches [Player r]. [Player q] temporizes and then finds [Player r], whose shot is stopped by [Player s].

In particular, we consider goals, goal attempts, and penalties ("attempt highlights") as well as penalty claims, crosses, dribbles, opponent errors, and offsides ("non-attempt highlights"). The highlights from the latter group only include the events that merited being mentioned in the live commentary, e.g., not all offsides were automatically considered highlights. This results in 644 highlights in total. The ground truth consists of a set of highlight timestamps annotated with the team to which a particular highlight is attributed. These highlights are never used during the training phase.

Data and Methodology

Our dataset, which covers 69 matches, contains 179,000 events, 2,027 preconditions, and 185 critical events. In addition, our stream-preprocessing procedure generates 423,000 virtual events. We split each match into subsequences of ten seconds allowing an overlap of five seconds between consecutive subsequences. Hence, each subsequence covers 20 events, which can be actual or virtual events, and the overlap between consecutive subsequences amounts to ten events. After preprocessing our dataset, we obtain 62,000 subsequences, where each subsequence corresponds to an example. Table 1 shows an anonymized excerpt of the data.

We predict highlights as explained earlier. As the timestamps in the commentary feeds are coarse-grained and approximate, we consider a prediction correct, if there is a true highlight in the commentary feed within 90 seconds of the peak. If peaks for both teams fall in the same window, we attribute the highlight to the team with the highest probability. We compute performance measures using the "leave-one-match-out" procedure. For each match, we train the model on the 68 other matches and compute the precision and recall for the remaining match aggregated over both teams. We report average precision, recall, and F_1 score over the 25 matches that we manually selected highlights from.

Based on our preliminary experiments and domain knowledge, we set the number of nearest neighbors k to 100, the exponential smoothing parameter α to 0.2, and the peak detection window w to 18. The choice of w is based on the analysis of the training data. It showed that, on average, a noteworthy event (e.g., a goal attempt) happens approximately every three minutes. Hence, we set the peak detection window to 18, which corresponds to 90 seconds, as this yields roughly one detected peak per three minutes.

Our unoptimized Python implementation classifies a window in approximately three seconds. Hence, we can predict

	Temporal aspect	Indirect estimation	Precision	Recall	F_1
POGBA	✓	✓	**0.44**	0.61	**0.51**
SpatTempDir	✓	–	0.43	0.59	0.50
SpatIndir	–	✓	0.36	0.46	0.40
SpatDir	–	–	0.34	0.39	0.36
Final 4th	–	–	0.22	**0.72**	0.33
Random	–	–	0.05	0.05	0.05

Table 2: Precision and recall for highlight prediction.

highlights in real-time since we use a five-second overlap between consecutive windows.

Results

Table 2 summarizes the results. POGBA outperforms all the baselines in terms of F_1, indicating that its core components of considering the full spatio-temporal data and performing indirect probability estimation are essential to the highlight prediction task. Random highlight prediction is practically infeasible. The simple deterministic baseline *Final 4th* has the highest recall at the expense of low precision. The baseline *SpatDir*, which neither considers the temporal aspect nor performs indirect critical event probability estimation, achieves a recall of 39% and a precision of 44%. Introducing indirect probability estimation into the baseline *SpatIndir* slightly increases the recall at the expense of precision. Introducing full spatio-temporal data into the baseline *SpatTempDir* allows increasing both measures.

Figure 3 shows POGBA's output for the second half of a representative match, including probability estimates, predicted highlights, and ground-truth highlights. We discuss the match in detail and highlight some of POGBA's errors. Even though the match ended in a goalless draw, the ground truth contains over 20 highlights. The precision is 39% and recall is 44% for highlight prediction on this match, which is below the average performance over the whole dataset. False positives often cluster together, along the stretches where one team controls the game, e.g., the stretch after the 60th minute for Team A. Furthermore, false positives tend to correspond to peaks with lower predicted probabilities than those of true positives, which suggests that more sophisticated estimation of the average level in the peak detection algorithm might improve performance. Finally, a false positive for Team A slightly before the 80th minute indicates the ground truth's imperfections: in that moment, a player of Team B committed a foul leading to a yellow card, implying a highly favourable situation for Team A.

Auxiliary Experiments

We briefly overview two auxiliary experiments concerning individual building blocks of POGBA. More extensive and detailed results are available in the online supplement.[3]

[3]https://dtai.cs.kuleuven.be/sports/pogba

Goal attempt prediction. We consider a simplified version of this task, where we only predict whether an attempt will occur in the next window, instead of predicting its attributes, as required by the full highlight-prediction pipeline. Only 5% of the windows are followed by an attempt. The DTW approach used in the previous experiments attains an area under the ROC curve (AUROC) of 0.8. The nearest-neighbor classifier based on the Euclidean distance between the last events in each window performs considerably worse, only attaining an AUROC of 0.6. This further illustrates the utility of considering full spatio-temporal data.

Goal probability estimation. This task is equivalent to probabilistic classification. In our data, only 9% of the attempts results in a goal. We compare four state-of-the-art probabilistic classifiers: *Naive Bayes*, *Logistic Regression*, *Random Forests*, and *Extremely Randomized Trees* (ERT). Empirically, the overall performance is not sensitive to the choice of the classifier. The AUROC values are high, ranging from 0.73 to 0.79. In the main experiment, we used ERT.

Related Work

Our approach is related to learning from time series and sports analytics.

Learning from time series. Nearest-neighbor-based methods with dynamic time warping (Berndt and Clifford 1994) (NN-DTW) are the state-of-the-art techniques for time series classification (Xi et al. 2006; Shokoohi-Yekta, Wang, and Keogh 2015). Alternative approaches based on symbolic time series abstractions (Lin et al. 2007; Rakthanmanon and Keogh 2013; Baydogan, Runger, and Tuv 2013), are neither faster, nor more accurate than NN-DTW (Wang et al. 2013; Bagnall and Lines 2014). Hence, we use a NN-DTW variant for estimating the event probabilities.

Alternatively, highlight prediction can be seen as an instance of outlier detection in time series (Gupta et al. 2014). However, outlier detection is usually an unsupervised task, whereas we evaluate on labeled examples. Goals might be considered outliers, but situations potentially leading to goals are more frequent and interesting on their own.

Sports analytics. The task of estimating $P(\text{goal} \mid \text{attempt})$ is a popular research topic in the soccer analytics community, where it is known as the expected-goals value of an attempt. The approaches proposed to date are mostly logistic regression models using different features of the attempts (Ijtsma 2015; Caley 2015; Lucey et al. 2014).

Furthermore, predicting how a game state will evolve has been studied for other sports. Cervone et al. (2014) propose the Expected Possession Value (EPV) model for basketball, which gives the number of points a team is expected to score during a possession. Routley and Schulte (2015) introduce a conceptually similar model for ice hockey.

Conclusions

This paper presents the POGBA algorithm for automatically predicting highlights in soccer matches from spatio-temporal data. The algorithm views this task as predicting the probability that a game state will lead to a goal. It first

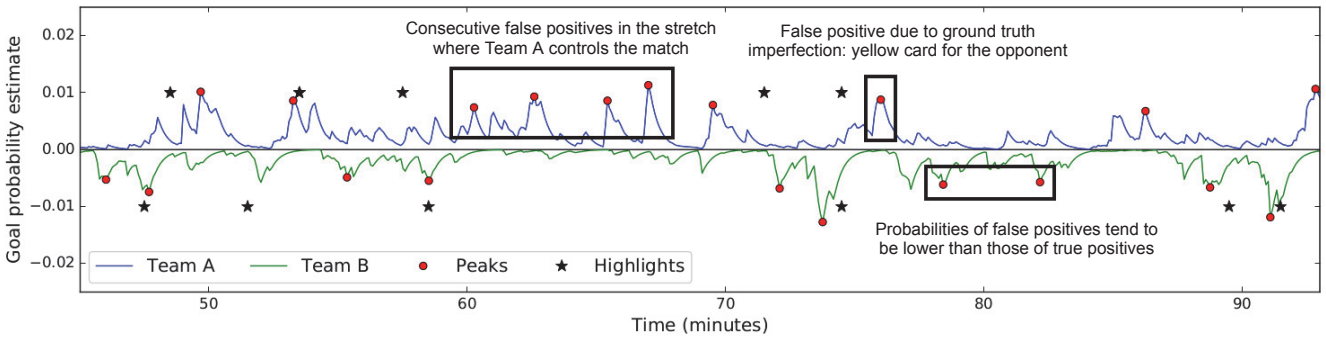

Figure 3: POGBA's goal probability estimates for both teams, detected peaks, and the ground truth highlights for the second half of a match. Although the match was a 0:0 draw, the ground truth contains more than twenty highlights. POGBA's precision and recall for highlight prediction on this match are 39% and 44% respectively.

estimates the probability of a game state leading to a goal attempt, and then that of a possible attempt resulting in a goal. On a real-world dataset, POGBA outperforms the baseline algorithms in terms of precision and recall.

Acknowledgments

Tom Decroos is supported by the KU Leuven Research Fund (C22/15/015) and FWO-Vlaanderen (G.0356.12). Vladimir Dzyuba is supported by FWO-Vlaanderen. Jan Van Haaren is supported by the Agency for Innovation by Science and Technology in Flanders (IWT). Jesse Davis is partially supported by the KU Leuven Research Fund (C22/15/015) and FWO-Vlaanderen (G.0356.12, SBO-150033).

References

Bagnall, A., and Lines, J. 2014. An experimental evaluation of nearest neighbour time series classification. *CoRR* abs/1406.4757.

Baydogan, M. G.; Runger, G.; and Tuv, E. 2013. A bag-of-features framework to classify time series. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(11):2796–2802.

Berndt, D. J., and Clifford, J. 1994. Using Dynamic Time Warping to find patterns in time series. In *Proc. of AAAI KDD Workshop*, 359–370.

Caley, M. 2015. Premier League projections and new expected goals. http://cartilagefreecaptain.sbnation.com/.

Cervone, D.; D'Amour, A.; Bornn, L.; and Goldsberry, K. 2014. POINTWISE: Predicting points and valuing decisions in real time with NBA optical tracking data. In *Proc. of MIT Sloan Sports Analytics Conference*.

Eastwood, M. 2015. Expected goals and Support Vector Machines. http://pena.lt/y/2015/07/13/expected-goals-svm/.

Gupta, M.; Gao, J.; Aggarwal, C.; and Han, J. 2014. *Outlier Detection for Temporal Data*. Morgan & Claypool.

Ijtsma, S. 2015. A close look at my new expected goals model. http://11tegen11.net/2015/08/14/.

Lin, J.; Keogh, E.; Wei, L.; and Lonardi, S. 2007. Experiencing SAX: A novel symbolic representation of time series. *Data Mining and Knowledge Discovery* 15(2):107–144.

Lucey, P.; Bialkowski, A.; Monfort, M.; Carr, P.; and Matthews, I. 2014. "Quality vs quantity": Improved shot prediction in soccer using strategic features from spatiotemporal data. In *Proc. of MIT Sloan Sports Analytics Conference*.

Müller, M. 2007. *Dynamic time warping*. Information retrieval for music and motion. Springer. chapter 4, 69–84.

Rakthanmanon, T., and Keogh, E. 2013. Fast shapelets: A scalable algorithm for discovering time series shapelets. In *Proc. of SDM*, 668–676.

Routley, K., and Schulte, O. 2015. A Markov game model for valuing player actions in ice hockey. In *Proc. of UAI*, 782–791.

Shokoohi-Yekta, M.; Wang, J.; and Keogh, E. 2015. On the non-trivial generalization of Dynamic Time Warping to the multi-dimensional case. In *Proc. of SDM*, 289–297.

Vidal, E.; Casacuberta, F.; Benedi, J. M.; Lloret, M. J.; and Rulot, H. 1988. On the verification of triangle inequality by Dynamic Time Warping dissimilarity measures. *Speech Communication* 7(1):67–79.

Wang, X.; Mueen, A.; Ding, H.; Trajcevski, G.; Scheuermann, P.; and Keogh, E. 2013. Experimental comparison of representation methods and distance measures for time series data. *Data Mining and Knowledge Discovery* 26(2):275–309.

Xi, X.; Keogh, E.; Shelton, C.; Wei, L.; and Ratanamahatana, C. A. 2006. Fast time series classification using numerosity reduction. In *Proc. of ICML*, 1033–1040.

Yianilos, P. N. 1993. Data structures and algorithms for nearest neighbor search in general metric spaces. In *Proc. of SODA*, 311–321.