Seq2Event: Learning the Language of Soccer using Transformer-based Match Event Prediction

Ian Simpson
University of Southampton
Southampton, UK
ijs1c20@soton.ac.uk

Ryan J. Beal
University of Southampton
Southampton, UK
ryan.beal@soton.ac.uk

Duncan Locke
Rugby Football Union
London, UK
duncanlocke@rfu.com

Timothy J. Norman
University of Southampton
Southampton, UK
t.j.norman@soton.ac.uk

ABSTRACT

Soccer is a sport characterised by open and dynamic play, with player actions and roles aligned according to team strategies simultaneously and at multiple temporal scales with high spatial freedom. This complexity presents an analytics challenge, which to date has largely been solved by decomposing the game according to specific criteria to analyse specific problems. We propose a more holistic approach, utilising Transformer or RNN components in the novel *Seq2Event* model, in which the next match event is predicted given prior match events and context. We show metric creation using a general purpose context-aware model as a deployable practical application, and demonstrate development of the *poss-util* metric using a *Seq2Event* model. Summarising the expectation of key attacking events (shot, cross) during each possession, our metric is shown to correlate over matches ($r = 0.91, n = 190$) with the popular *xG* metric. Example practical application of *poss-util* to analyse behaviour over possessions and matches is made. Potential in sports with stronger sequentiality, such as rugby union, is discussed.

CCS CONCEPTS

• **Applied computing**; • **Computing methodologies** → **Machine learning**; **Model development and analysis**;

KEYWORDS

Sports Analytics, Soccer, Football, Deep Learning, Neural Networks, Transformer, Recurrent Neural Networks, Metric Development

ACM Reference Format:
Ian Simpson, Ryan J. Beal, Duncan Locke, and Timothy J. Norman. 2022. Seq2Event: Learning the Language of Soccer using Transformer-based Match Event Prediction. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22), August 14–18, 2022, Washington, DC, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3534678.3539138

KDD '22, August 14–18, 2022, Washington, DC, USA.

ACM ISBN 978-1-4503-9385-0/22/08…$15.00
https://doi.org/10.1145/3534678.3539138

1 INTRODUCTION

In a team-based sport such as soccer,[1] quantitative analysis plays an integral role within the coaching process and in informing strategic and tactical decision making [9, 34]. Its importance and prevalence has accelerated in recent years as professional teams in the multi-billion dollar industry [6] strive for additional insight to support and enhance coaching philosophy. This acceleration has been driven by the breadth and availability of game event and tracking data, coupled with significant research and the integration of analysis techniques born and advanced in other industries [11]. Applying traditional techniques with sport-specific formulations has led to the development and implementation of several new metrics [10, 12, 20] which have successfully delivered greater insights into the game. However, as soccer analytics gains maturity, such formulations become harder to discover and more sophisticated techniques are required to continue advancing the frontiers in the field [9].

Recent advances in machine learning (ML) techniques, combined with a prevalence of data, presents an opportunity to make sense of complexity in the sport far beyond that of traditional statistical techniques. Whilst there is evidence of the use of sequential ML techniques in sports research [15, 34, 35], for the most part sequence invariant methods like Random Forest (RF), Multi-Layer Perceptron (MLP), or the Convolutional Neural Network (CNN) are used, with sequences of match events being analysed by pre-determined aggregation or concatenation of data on time.

Techniques inherently designed to model sequential data have been responsible for state of the art results in the Natural Language Processing (NLP) domain for around a decade, with the Recurrent Neural Network (RNN) [13, 19] family of architectures, specifically the Long Short-Term Memory (LSTM) [17] and Gated Recurrent Unit (GRU) [8], and Transformer [32] family of architectures leading the way in cutting-edge applications.

In this paper, we show that these architectures, when built into an appropriate framework, may be leveraged to solve sports analytics problems. Our novel *Seq2Event* model is presented as a working example of this, where a sequence of match event data is used to predict the next match event. Hyperparameter search experimentation finds that the model is better than baseline statistical methods, and provides some information on optimal model hyperparameters when applied to a cross-section of top tier matches. Analysis of

[1] Known synonymously as "Association football", or "football".

predictions demonstrates that context-aware predictions are made by the model.

We introduce the novel *poss-util* metric, which articulates the attacking possession utilisation of any given possession, and is formulated as an integral function of the *Seq2Event* model probability of attack. We apply these methods to teams from La Liga, demonstrating the metric's ability to characterise team attacking behaviour. Straightforward development of the metric (*poss-util* is a function of cross and shot probabilities) illustrates the potential for general purpose models (*Seq2Event* predicts six attributes of the next event) to be used to accelerate metric development and deployment.

This paper seeks to advance the state of the art in sports analytics in the following ways:

(1) The novel *Seq2Event* model allows representations of sequences of match event and contextual features to be learnt by an RNN or Transformer encoder component, from which the next match event action and location are predicted.
(2) Using real-world data from 138 matches from seven top tier competitions, models are trained to explore and identify optimal hyperparameters for such a model. Elman-RNN, LSTM, GRU and Transformer variants all yield better results than baseline Autoregressive (AR) and Markov models (>21% improvement on best baseline), with LSTM and Transformer models yielding the best predictive performance (53% and 46% improvement on best baseline).
(3) We introduce the *poss-util* metric to articulate the attacking utilisation per possession of play. Context-dependent expected probability of attack is integrated over events per possession, and multiplied by an indicator variable based on whether an attack actually occurred, or not.
(4) We present a case study based on five teams from the 2017/18 Spanish La Liga season, demonstrating the utility of the model combined with the metric in showing team possession utilisation.

This paper is organised as follows: relevant literature in the sports analysis and machine learning (ML) domains is reviewed in Section 2. The *Seq2Event* model is introduced in Section 3, with results of empirical evaluation and case study application presented in Section 4 and Section 5 respectively. Potential for utilisation of the model for metric development and decision making in the soccer industry, as well as in other professional sports is discussed in Section 6.

2 BACKGROUND

In this section, literature on techniques for modelling sequences is reviewed, followed by a summary of metrics used to identify team strategies in soccer.

2.1 Modelling Sequences

Traditional statistical are reviewed in order to form baseline models, and machine learning techniques used in the *Seq2Event* models are reviewed.

Baseline Techniques. Auto Regressive (AR) models are one component of the popular Auto Regressive Integrated Moving Average (ARIMA) family of models [16], and serve as a simple baseline model for continuous feature observations of soccer event data. The Markov family of models [14] are popularly employed for discrete Markovian data, and serve as a simple baseline model for the categorical action feature observations of soccer event data. In this domain, the system is considered autonomous as the data is being inspected retrospectively. Whilst the process ('playing soccer') is so complex that other latent parameters could be assumed to exist which cannot be observed, systematic and robust observations are made and so the simple Markov chain model based on a transition probability matrix is selected as a baseline model for action prediction. The model order, i.e. how far back the model looks, is a hyperparameter of both baseline models.

Machine Learning Techniques. The RNN [19, 25] was introduced as an ML approach to learning sequences. With a network architecture based on a feedfoward deep neural network, recurrent connections are added and during operation inputs are received and fed forward. Some output unit values are fed into persistent units, thus enabling the network to have some memory of the state. The Elman-RNN [13] is employed as the simplest ML model component in the *Seq2Event* architecture.

RNN application to long sequences had been limited by the noisy gradients problem, whereby trends over long periods become lost due to the error noise. LSTM [17] aimed to solve this problem by introducing a Constant Error Carrousel (CEC) and applying this concept through LSTM cells with gates, thus avoiding the accumulation of error and loss of signal encountered previously. LSTM was responsible for many state of the art results on many NLP tasks, but has been surpassed by the use of Transformer models in this domain [32, 33]. The GRU [8] cell was proposed as a simpler alternative to the LSTM cell, containing one third fewer learnable parameters. However, it has been observed that on more complex sequences, LSTM may provide better fit due to controlling the exposure of the memory to the network [7].

The Transformer [32] aims to solve sequential learning problems without the use of recursion, instead using positional embedding and attention to learn the sequential nature of the data, with an encoder-decoder architecture. By eliminating the need to iterate over the data each epoch these models can be much faster than RNN equivalents, as long as the larger Transformer model can fit in computational memory. The full model, comprising an encoder stack followed by a decoder stack has been used to set state of the art results in sequence to sequence NLP tasks, e.g. English to German translation with a sequence of English text as the input and a sequence of German text as the output. In the sequence to event soccer prediction task, since only a single event is being predicted rather than a full sequence, the decoder stack is not required, and therefore only the Transformer encoder stack is employed.

Historically, the use of LSTM and Transformer models in soccer has focused on for match video processing tasks such as annotation [30] and summarisation [2], although recently other authors have also begun to look at team behaviour prediction [15, 35].

2.2 Metrics for Identifying Team Strategies in Soccer

Expected Goals (*xG*) was originally introduced to the sport of ice hockey [21]. It is identified that a limitation of ice hockey is the low scoring rate compared to other sports such as basketball. The paper identifies that the randomness and scarcity of goals limits the ability to properly judge current performance and to predict future performance of teams using goals alone. Sparsity of scoring is a difficulty shared with soccer [9]. In order to provide a more continuous context, the *xG* model allows an estimate of the number of goals a team ought to have scored based on their performance on key metrics such as shots on goal, missed shots, blocked shots, turnovers, face-offs, and actual goals scored. Ordinary Least Squares (OLS) regression and Ridge regression were used on a time-series of these metrics to build models by *xG* [21]. *xG* was first recorded in soccer in 2016 [12], with models based on random forest and Adaboost [12].

A limitation of *xG* is that it is a function of instantaneous shot actions only. It does not account for events that deliver opportunities that do not result in significant match events from which *xG* could be accrued. Several metrics aim to account for this, such as the Off-Ball Scoring Opportunity [31], Expected Assists (*xA*) [23], Expected Goal Chain (*xGChain*) [20], Expected Goal Build-Up (*xGBuildup*) [29], Expected Threat (*xT*) [29] metrics.

A more holistic approach was taken by the Valuing Actions by Estimating Probabilities (VAEP) metric [10], which is still only a function of on-ball actions, but estimates the probability of scoring a goal in the near future across all actions. The metric has proven successful and succinctly accounts for over and under achievement in both scoring and conceding aspects of the game. Production of VAEP scores is underpinned by the need for a model for the scoring and conceding probabilities, for which Decroos et al. [10] use a Catboost gradient boosted model on the past three actions to generate probabilities for whether a goal was scored or conceded next.

Estimating the reward of team tactics is the focus of Beal et al. [3], and, building on this, the long term optimisation of decision making [4]. 'Fluent objectives' are defined for a team, which are a function of 'objective variables' which correspond to different points in the planning horizon of an agent. Markov Chain Monte Carlo (MCMC) is performed in order to make predictions about the future performance of strategies against different objectives, and this is used to inform the selection of optimal objectives by the agent. See Beal et al. [5] for a more complete review.

3 SEQ2EVENT MODEL FOR MATCH EVENT PREDICTION

In this section the architecture, and associated loss function, of our novel *Seq2Event* model for match event prediction are presented.

3.1 Loss Function

Defining a loss function that characterises the error is necessary to correctly measure the success of a model. Whilst all of the features in the dataset are relevant as model inputs, not all provide differentation of styles of play. For example, score advantage is an important contextual input, but changes infrequently and is therefore not of interest to directly predict. Action type and resultant spatial location in x, y of on-the-ball actions were chosen as target variables, as they were the most fundamental features of interest, and are directly observed and recorded in the source data.

Cross Entropy Loss (CEL) is popularly used for equivalent multi-class classification tasks in NLP, and is used here to measure error of the prediction of the next action. Root-Mean-Square-Error (RMSE) loss was identified as suitable for the x, y co-ordinate error, particularly due to this metric's intuition as Euclidean distance on a spatial metric. Applying an unweighted sum was found to lead to a bias towards models that prioritised reducing the RMSE loss, and so the final loss function is presented as a weighted sum with the weights determined empirically. The resultant loss function is shown in Equation 1.

$$\mathcal{L}(\hat{y}, y) = 5 \cdot CEL(\hat{y}_{action}, y_{action}) + RMSE(\hat{y}_{loc}, y_{loc}) \quad (1)$$

Event action type has a class imbalance, so to resolve this, the CEL is itself weighted on the reciprocal class occurrence proportion. This means, for example, that the strong true positive of a 'pass' will not yield as low a loss as an equivalently strong true positive of a 'shot', since shots occur less frequently (1.4% vs 56%). Scoring goals, change of possession events, and end-of-match events are purely contextual and are not considered indicators of style for this task, so are given zero weights within the CEL function.

3.2 Seq2Event Model Architecture

The *Seq2Event* model comprises seven main stages, with RNN and Transformer variants, as shown in Figure 1. Inputs are match events from $t - seqlen$ to $t - 1$, with output an estimate of the event at t.

Stage 1: Learnable Embedding. For this task, we could simply treat the actions as words. However, we know that the context of an event has meaning and wish to capture the ten continuous variables in the model. An embedding layer is used to embed the actions and a dense layer is used to transform the continuous variables.

Hyperparameters:

- *seqlen*: sequence length; how far back the model looks.
- *act_embed_dim*: action embedding dimensionality.
- *cont_embed_dim*: continuous feature dense layer dimensionality.

Stage 2: Concatenation. We need to pass one matrix, composed of vectors of sequential events to the RNN/Transformer, and so two outputs from the previous stage are concatenated. In the Transformer variant, positional embedding is also applied at this stage.

Stage 3: RNN/Transformer Component. Either an RNN or a Transformer encoder component operates in this stage.

RNN variant hyperparameters:

- *RNN type*: one of Elman-RNN, LSTM, or GRU is specified.
- *hidden size*: the number of hidden units within each RNN cell.
- *number of layers*: the number of stacked RNN cells.
- *dropout rate*: if number of layers > 1, the proportion of final layer weights stochastically ignored during training.
- *directionality*: unidirectional: this stage is a function of ordered inputs from start to end of input sequence; else bidirectional: this stage is a function of ordered inputs from start to end, then end to start of input sequence.
- *activation function*: for Elman RNN, the activation function (e.g. ReLU) applied prior to updating the hidden state at each time iteration.

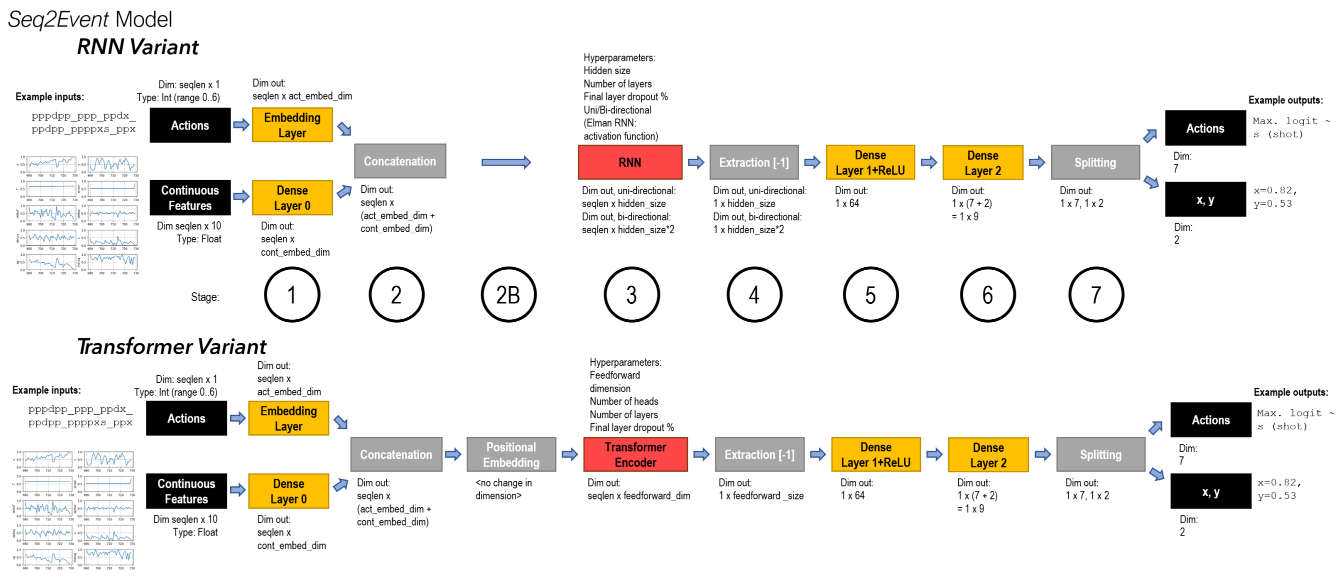


Figure 1: *Seq2Event* model architecture. Predictions of next action and position are made given a source of $11 \times seqlen$ **comprising ten continuous features and one categorical action feature. Learnable components are shaded orange and red.**

Transformer variant hyperparameters:

- *feedforward dimension*: analogous to the RNN hidden size.
- *number of heads*: the number of heads in the multi-head attention component.
- *number of layers*: the number of parallel layers in the stack.
- *dropout rate*: analogous to the RNN dropout rate.

Stage 4: Extraction of final prediction. The RNN/Transformer component makes one-ahead predictions across the whole set. However, for this task we are only interested in the final prediction of the set, representing the next event.

Stage 5: Dense Layer with ReLU Activation. The focus from Stage 5 onwards is on mapping the representation to action and x, y outputs. A dense layer with rectified linear unit (ReLU) [22] activation function is applied. Initial prototypes excluded this layer, but Javid et al. [18] refer to the potential for such a structure to aid with learning particularly when larger hidden unit sizes are used, since neuron outputs may be ignored when they are below the $x = 0$ activation threshold. Application to prototype models showed a modest improvement in training loss (circa 1%).

Hyperparameter:

- *output dimensionality*: dense layer dimensionality.

Stage 6: Dense Layer. The final learnable layer is a dense layer mapping the representation to the final outputs. Seven action types are predicted, although three of these types are redundant since due to being given zero weighting in the loss function, and therefore could be reduced to four action types in future versions.

Stage 7: Splitting. The vector of length 9 is split into a vector of action logits of length 7 (for the four actions plus three change of possession characters), and a vector of x, y position co-ordinates of length 2. Under training, Equation 1 is applied as the loss function to these outputs. For practical application, simplified feature engineered actions pass ('p'), dribble ('d'), cross ('x'), and shot ('s') were used, as elaborated upon in Appendix A. The softmax of the four action logits is taken, yielding next action prediction probabilities.

Each individual prediction is made given the previous 40 events in the example of model outputs shown in Figure 2. Gaps in the x, y plots are periods of play where the team is not in possession, which are shown as '_' in the action sequence. The model does not predict turnovers as they are given zero weight in the loss function, which is by design, since the intended purpose is to analyse attacking behaviour where turnover prediction is of less relevance. However the loss function and/or the final stage of the model could easily be modified to operate on other tasks that could include turnover prediction.

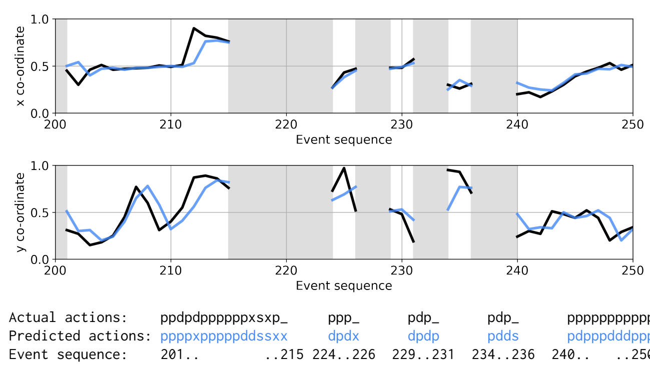


Figure 2: *Seq2Event* model predictions (blue) of next event location and action vs actual (black), given context of previous 40 events at each time step. Gaps in location (grey) indicate possession turnovers.

4 EMPIRICAL EVALUATION

To evaluate our model, match event data from the WyScout Open Access Dataset [24] was used, covering the 2017/18 season of the English, French, German, Italian and Spanish men's first divisions, plus the international Euro 2016 and World Cup 2018 competitions. Comprising 1,941 matches in total, a sample of 138 matches representative across success and competitions was used. Further details on the reproducibility of our results can be found in Appendix A.

4.1 Experiment 1: Hyperparameter Selection

Hyperparameter search was conducted by initial experimentation across a wide range of values on all parameters to identify plausibly optimal values on which hyperparameter grid search was conducted. A total of 145 different models were fitted, using data streamed as shown in Figure 3. All models' performance by test loss is shown in Figure 4.

Continuous features were engineered from on-the-ball x, y, and time data to provide features of known contextual significance in soccer analytics [1, 10] whilst being somewhat independent in representation. The resultant set comprised x, y, T, and their respective delta between observations; distance covered between observations; angle and distance to opposition goal; and score advantage (with negative values indicating a goal deficit). A categorical action feature was engineered by mapping the 106 possible action types on to four generalised action types: pass, dribble, cross, and shoot. Reduction was made according to three principles: (1) category support had to be sufficient so as to permit learning on the modestly sized dataset; (2) actions had to be sufficiently distinct in their meaning in the context of the soccer domain; (3) the simplified encoding had to yield sequences of reasonable sequential diversity.

With the task of predicting the next offensive action given a team's previous offensive actions, only the offensive actions of the team of interest were kept per match, but with a change of possession character added. Thus, observations of a specified length and step can be made across the data, as shown in Figure 3.

We fitted 12 baseline Autoregressive and Markov chain models of orders 1-5 and on x, y and all ten continuous variables. The best baseline model was of order 1 on all ten continuous variables (test loss 0.704). The worst *Seq2Event* model bettered this score, a Transformer variant with 17 heads and a hidden unit size 4096 (test loss 0.548). The best performing was an LSTM variant with sequence length 100, 1 layer, hidden unit size 8, and unidirectional order (test loss 0.332), with its bidirectional equivalent coming second (test loss 0.344). A Transformer variant with sequence length 40, 1 layer, and hidden unit size 8 came third (test loss 0.362).

A PyTorch implementation of the model [28] was run on Google Colab Pro hardware, with acceleration provided by an Nvidia Tesla P100 GPU. In general, for equivalent settings, Transformer models were quicker to train than RNN variants. The best LSTM model took 15.5h to train. Reducing the sequence length to 40 reduced the model training time to between 3.5 to 5.5h depending on other hyperparameters, but this was still much longer than the comparable best Transformer model which took 1.4h to train. This validates one of the known benefits of Transformer models, in that the architecture avoids iterating over the data, thus speeding up fitting as long as the whole source data fits into memory; whereas RNN models, including LSTM, must operate in series on the data, with computational complexity of order linear to the source sequence length [26].

LSTM models yielded the best results by test loss, and outperformed the Transformer model with equivalent configuration (test loss 0.332 vs 0.379), demonstrating an enhanced ability to learn patterns in longer sequences which has also been noted with this architecture in other domains [7, 8].

GRU models have fewer learnable parameters than LSTM and this resulted in a marginally faster training time, but test performance was not as good as LSTM or Transformer models. Elman-RNN models are the earliest type of RNN and generally provided the worst test performance of all *Seq2Event* variants.

Models with sequence lengths of 5, RNN models with more than 2 layers, and Transformer models with more than 2 heads all performed poorly (best model losses 0.423, 0.440, 0.384 respectively).

We highlight LSTM as being promising for future research especially involving sequence lengths of 100 or more, where the computational resources can be justified. For the application of identifying team strategies by examining differences against what 'the average team' would do, looking back at too many actions can be undesirable. The model may pick up the 'style' of the team and start to benefit by predicting the next event using that, providing predictions against what 'the current team' would do. In the next sections of this paper, the Transformer model with the third best performance and sequence length of 40 was used to generate results due to its superior speed of computation.

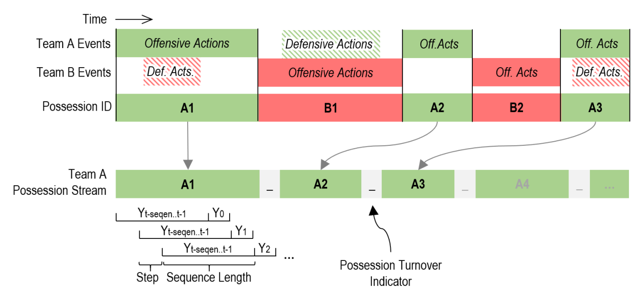


Figure 3: WyScout data is rearranged to form streams for *Seq2Event* modelling. From source data ordered on time (top two rows), periods of possession for each team are identified (third row). Events from each possession are aggregated by team with a turnover indicator added between possessions.

Figure 4: Train (blue), validation (yellow) and test (black) loss over all models trained during hyperparameter search.

4.2 Experiment 2: Analysis of Action Prediction Probability

Actual occurrence of actions was assessed against the predicted probability of action occurrence, for all four action types (pass, dribble, cross, shot). Focusing here on shot prediction, Figure 5 shows the spatial distribution of relevant summary statistics.

Figure 5 (a) demonstrates that shot prediction occurs approximately in line with actual data (see Figure 10 for comparison). Figure 5 (b) demonstrates that mean shot prediction probability is generally higher closer the opposition goal, given a shot is predicted. Higher intensities in the left hand side have very low sample size and are insignificant. Figure 5 (c) demonstrates model diversity of shot prediction probability, over given locations. Diverse shot probabilities are made even in areas of high sample size on the right hand side. Similar diversity can be observed in Figure 9. These are important observations because they demonstrate that although the model is indeed predicting higher shot probabilities where we would expect given the empirical spatial distribution, their generation is neither solely an independent function of spatial source nor of predicted features.

The multiclass confusion matrix for this model is shown in Table 1. For all actions the model, on average, correctly assigns the highest probability, with the exception of pass where it predicts dribble ($P = 0.38$) more than pass ($P = 0.36$). This could potentially be resolved by fine tuning the weights in the CEL portion of the loss function. Given the low actual class occurrence of shot (2.2%), cross (3.9%), and dribble (10.2%), the results are reasonable.

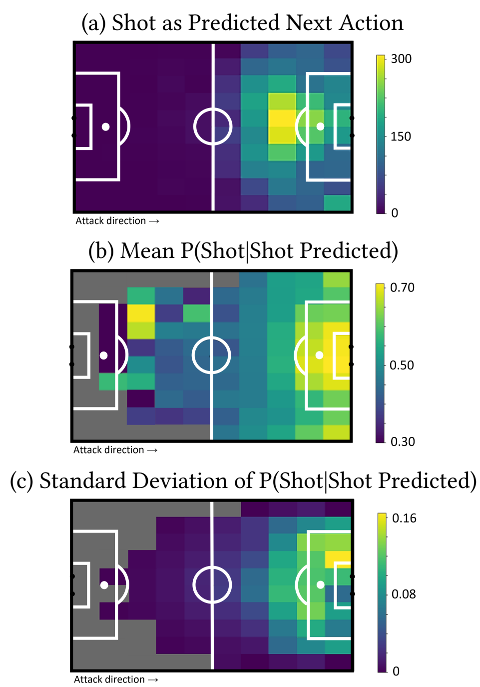


Figure 5: Spatial distribution of shot prediction statistics (n=5,666).

Table 1: Multiclass Confusion Matrix (Mean Probability)

	Predicted Action			
Actual Action	Pass	Dribble	Cross	Shot
Pass	0.36	0.38	0.16	0.10
Dribble	0.32	0.35	0.20	0.13
Cross	0.13	0.19	0.40	0.29
Shot	0.07	0.12	0.27	0.54

4.3 Experiment 3: Analysis of Possession Utilisation

Attack metric development. Of the four actions predicted, shot and cross may be considered to be 'attacking' in intent. A cross is defined as a ball played from the offensive flanks aimed towards a teammate in the area in front of the opponent's goal, and a shot is defined as an attempt towards the opposition's goal with the intention of scoring a goal.[2] Summing shot and cross predicted probabilities thus gives an 'attack' probability. Further summing per possession gives a measure of the weight of expectation of attack accumulated during each possession.

In order to distinguish between attacking and non-attacking possessions, the summed attack probabilities are multiplied by -1 when no attack event (shot or cross) occurred in a possession. The reasons for not attacking vary, and this approach allows some insights to be gained. A weaker team may simply have not managed to create attacking opportunities (thereby incurring a low magnitude cumulative expectation, with negative sign). Conversely, a stronger team may have been seeking the optimal opportunity to commence an attack (incurring a high magnitude cumulative expectation, with negative sign, in the process). Additionally, contextual factors such as game state (winning, losing, drawing) and period of the match may also influence a team's attacking strategy.

For ease of interpretibility, percentile rank is applied to both positive and negative groups, to provide a metric for each possession with range $[-1, 1]$, and the resultant statistic is termed *poss-util*.

Metric application. The metric is applied to selected teams for the 2017/18 season of La Liga, Barcelona (league champions), Atlético Madrid (finished 2nd place), Real Madrid (3rd), Girona (10th) and Málaga (20th). Distribution over possession by team is shown in Figure 6 (a) and can be seen to be lower for positive values of *poss-util*, since a minority of possessions result in an attack (23%). Values of high magnitude, i.e. close to -1 or 1, indicate possessions where a high probability of attack was accrued during the possession, but with only those with positive sign having resulted in an attack (shot or cross).

Barcelona and Real Madrid are noteworthy for having distinct distributions compared to the other teams. They tend to generate possessions with higher attack expectation, as shown by the higher density at high magnitude positive and negative values. Atlético Madrid, Girona, and Málaga have similar distributions to each other. Atlético Madrid's similarity to these two teams is perhaps surprising given the differences in final league standings, from second position to mid-table and last place. However, Atlético Madrid are known to play with an unusually defensive style for such a successful

[2] https://dataglossary.wyscout.com

team.[3] They conceded the fewest goals per match (0.6) which is significantly below the league average (1.6). By contrast, in attack, they scored just below the average number of goals per match (1.5 vs 1.6), which validates our finding that Atlético Madrid are similar to more average teams in attack. Girona scored 1.3 goals per match and finished mid-table. Málaga scored only 0.6 goals per match, despite having a similar distribution of *poss-util* to Atlético Madrid and Girona, although as last place finishers this metric highlights the importance of technical proficiency in the final third and the ability of players to execute specific skills (shot, cross) to a high level of accuracy and/or for players in these positions to make optimal decisions on when to perform the action(s). Barcelona and Real Madrid both accounted for the highest number of goals scored in the league (2.61 and 2.47 mean goals per match respectively).

Using *poss-util* to predict goals. Goals scored was selected as a comparison metric, as a key and well defined statistic. Sum, mean, and median of all, of only positive and of only negative *poss-util* were analysed for correlation (nine experiments). Median positive *poss-util* had the highest Pearson correlation ($r = 0.47$). Summed positive *poss-util* was expected to be the most intuitive, but had only a weak correlation ($r = 0.17$). Mean positive *poss-util* showed only a slightly lower correlation ($r = 0.46$) than median, and is easier for a sports analyst to comprehend, so was the chosen method. Finally, a linear transform of the mean positive *poss-util* over possessions per match ($p \in M$) against actual goals over the season was used to yield predicted goals ($\hat{g}$), as shown in Equation 2.

$$\hat{g} = 6.5 \cdot \frac{\sum_p p_{poss-util}}{|p|} - 1.5 \; \forall \, p \in M : p_{poss-util} > 0 \qquad (2)$$

Validation against actual goals and *xG*. Predicted goals by *poss-util* was found to moderately correlate with goals scored ($r = 0.46$ over matches), and found to strongly correlate with *xG*[4] ($r = 0.91$). *xG* shows marginally stronger correlation with actual goals than our metric ($r = 0.57$ vs 0.46). Figure 7 demonstrates further similarity between the two metrics, although again *xG* performs marginally better (RMSE 1.29 vs 1.42). Table 2 shows a very high correlation of both metrics against actuals when mean aggregated over the season, and with our metric performing marginally better (*poss-util* $r = 0.98$ vs *xG* $r = 0.97$). Facets of team under and over-performance against *xG* are visible in our metric also. Since *xG* is a popular and relied upon model for the specific task of goal scored prediction[31], the similarity in predictive performance validates our metric. By induction, this also validates the underlying *Seq2Event* model from which it is derived, and which was trained on the more general task of next event prediction.

5 MODEL APPLICATION TO LA LIGA

In this section, we show how the *Seq2Event* model can be applied practically as a team profiling method using the *poss-util* metric to deepen game understanding and derive additional insights into attacking behaviour over the course of a match.

Match timeline view. Computing the metric per possession, Figure 8 shows the evolution of *poss-util* over two matches. The first

[3]https://bleacherreport.com/articles/2589852-analysing-atletico-madrids-defensive-structure-under-diego-simeone
[4]https://understat.com/league/la_liga/2017

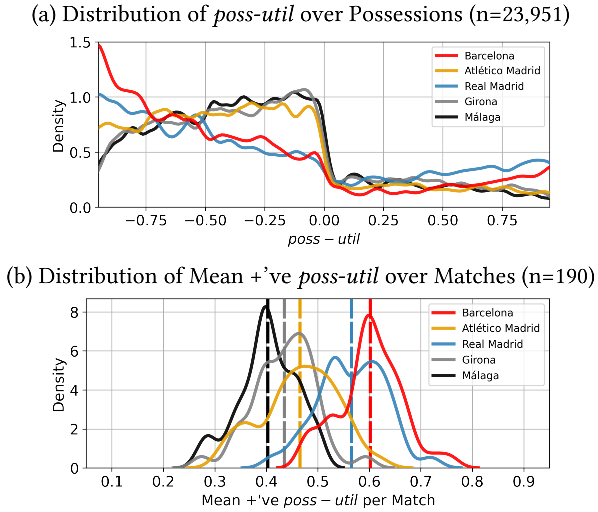


Figure 6: Analysis of behaviour by team using *poss-util*.

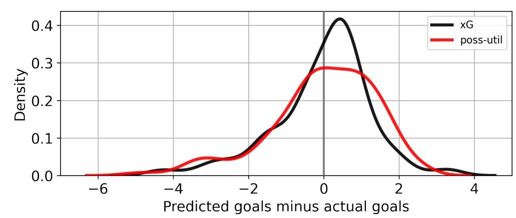


Figure 7: Distribution of error of *xG* and predicted goals by mean *poss-util* over matches (n=190).

Table 2: La Liga 2017/18 Mean Goals Scored per Match

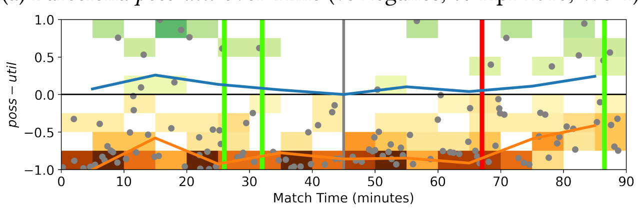
Team	Actual Goals	*poss-util*	*poss-util* Goals (Δ)	*xG* Goals (Δ)
Barcelona	2.61	0.60	2.40 *(-0.21)*	2.38 *(-0.23)*
Atlético Madrid	1.53	0.46	1.49 *(-0.04)*	1.32 *(-0.21)*
Real Madrid	2.47	0.57	2.21 *(-0.26)*	2.40 *(-0.07)*
Girona	1.32	0.44	1.36 *(+0.04)*	1.37 *(+0.05)*
Málaga	0.63	0.40	1.10 *(+0.47)*	0.94 *(+0.31)*

plot shows Barcelona winning 3-1 against Leganes on 7 April 2018. Overall, Barcelona's mean positive *poss-util* was 0.61, corresponding to 2.5 predicted goals (which is similar to *xG* metric 2.7). As depicted by the orange 10-minute rolling mean line in the negative region, Barcelona can be seen to generate lots of high attacking potential possessions that do not convert into attack (line closer to -1) throughout the match. When they do convert to attack, the median in the positive region can be seen to be high, although as denoted by the rolling mean line, attacks occur infrequently. Looking at other relevant match statistics, it is of note that Barcelona took and maintained a lead after goals in the 26th and 31st minutes; and Leganes achieved an *xG* of only 0.65. Two main insights can be gained from the use of the model in this way: Barcelona had a high predicted goals and performed approximately as predicted in that regard; they did not attack often (denoted by the blue line) despite accumulating a lot of high potential possessions.

The second plot shows Real Madrid losing 0-1 to Villarreal on 13 January 2018. Overall, Real Madrid's mean positive *poss-util* was 0.49, corresponding to 2.4 predicted goals (similar to *xG* metric 2.35). Real Madrid can be seen to consistently generate a mixture of possessions of varying attacking potential and conversion to attack throughout the match. Villarreal scored their only goal in the 86th minute.

Possession overview. Selecting possessions of interest based on the metric, Figure 9 (a) shows one of Barcelona's highest possessions by *poss-util*. Brighter colors denote higher attack expectation, and this long possession can be seen to accumulate over many instances of moderate expectation of attack at the edge of the opposition third, followed by several instances of high expectation of attack in the opposition left corner and penalty area. Ultimately, a cross was taken which did not lead to a goal, but a patient build-up of possession followed by an attack recorded a high metric score. By contrast, the second plot shows a possession with a moderate metric score of 0.39. A direct play, initially expecting a long-ball cross from the goalkeeper, followed by an unsuccessful shot.

(a) Barcelona *poss-util* over Time (vs Leganes, 07 Apr 2018, W3-1)



(b) Real Madrid *poss-util* over Time (vs Villarreal, 13 Jan 2018, L0-1)

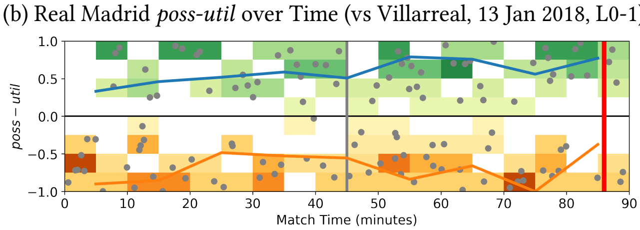
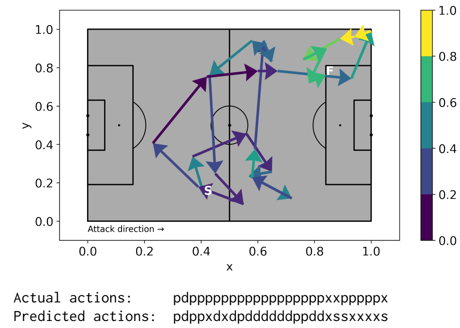


Figure 8: Evolution of *poss-util* over time. Grey points represent individual possessions; shaded cells give a visual indication of point density and magnitude; 10 minute positive and negative rolling mean lines shown in blue and orange. Vertical lines indicate goals for (green) and against (red).

(a) Barcelona Possession (vs Leganes, 07 Apr 2018, W3-1)
poss-util=0.98



(b) Real Madrid Possession (vs Villarreal, 13 Jan 2018, L0-1)
poss-util=0.39

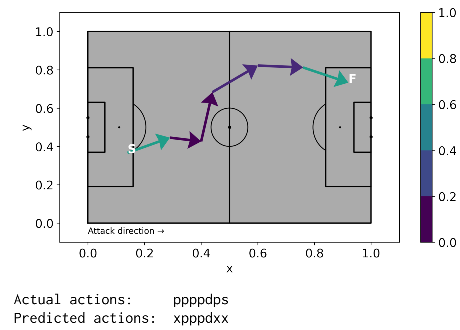


Figure 9: *poss-util* model predicted probability of attack as next action over actual x, y and actual actions vs predicted most likely action over two possessions. Color by probability of attack; 'S' and 'F' denote possession start and finish. For action decode, Table 3 refers.

6 DISCUSSION

Team Behaviour as Dialect. Events in most sports generally happen sequentially. Patterns appear, and strategy decisions are made, over multiple temporal scales. In this way, sport can be modelled and learnt using RNN and Transformer components to effectively 'learn the language of sports'. In much the same way that dialects can be detected by analysing differences in vocabulary from that expected, different tactics can be detected by analysing differences in actions from that expected. In our soccer dataset, there were 1,401 events recorded on average per match, and each action can be classified to a learnt tactic. By analysing the occurrence of tactics at a macro-level both in aggregate by team and sequentially, insights into tactical utilisation, strategy, and impact are achieveable.

General Purpose ML Models vs Specific Statistical Models. Putting effort into training more advanced but general purpose probabilistic models is highlighted as a potential benefit to the professional sports industry with the potential to provide teams with versatile modelling capabilities which can adjust and adapt to derive contemporary metrics. As sports evolve, these may be used to help determine the influence of law changes or season structures.

The model we trained was given a general task of predicting the next match event as parameterised according to the general attacking features that were engineered. From that basis, aggregation and linear regression were then readily performed on the model probabilities to generate a useful metric which we have shown correlates with a popular specific metric. From the other probabilities presented by the *Seq2Event* model, formulation of other metrics should be readily achieveable, and by modifying in particular the final layers of the model, and the event streaming, the general principle could be put to use on tasks other than examining attacking styles.

6.1 Future Work

Further Leveraging Model Outputs. Initial work was conducted to utilise the spatial features that are predicted. Analysis of prediction error in x location showed no significant difference in means across the five teams analysed in Section 5. However, the standard deviation of error correlated weakly with standings and goals scored (standard deviations of 0.157, 0.176, 0.164, 0.190, 0.193 for Barcelona to Málaga in 17/18 La Liga final standing respectively). Essentially, the model was found to be able to predict the next x position of Barcelona and Real Madrid more precisely than the other teams.

Further initial work was conducted to cluster on the difference between model predicted action logits and actual actions, with five styles of play identified which could then be counted by team and used to help to identify team behaviours. This is something we would explore more in future work, working closely with domain experts.

Application to Other Sports. Other sports such which have a stronger sequentiality than soccer could also benefit from these techniques and our model. For example, in rugby union, more distinct state transitions occur from rucks, mauls, line-outs and scrums. In rugby league, play resets after each tackle with possessions limited to six tackles before possession is turned over. In American football, running, passing or kicking plays are made with possession turned over if a team has not progressed ten yards over four plays.

7 CONCLUSION

The novel *Seq2Event* framework for applying sequential machine learning techniques to predict the next match event data in soccer has been presented, accompanied by results of experimentation to find optimal hyperparameters, and evidence of contextual event prediction and validation against *xG*.

In terms of practical application, we have demonstrated the ability for a general purpose probabilistic model to aid in rapid metric prototyping, as evidence for which we have presented the *poss-util* metric with application to La Liga. As a caveat, we note that specific models such as *xG* are likely to out-perform general models, but assert that general models have a place in the professional sport analytics industry. We suggest that sports with stronger sequentiality may particularly benefit from this framework, and also suggest that initial and final layers may be adjusted for other tasks.

REFERENCES

[1] David Adams, Ryland Morgans, Joao Sacramento, Stuart Morgan, and Morgan D Williams. 2013. Successful short passing frequency of defenders differentiates between top and bottom four English Premier League teams. *International Journal of Performance Analysis in Sport* 13, 3 (2013), 653–668.

[2] Rockson Agyeman, Rafiq Muhammad, and Gyu Sang Choi. 2019. Soccer video summarization using deep learning. In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE, 270–273.

[3] Ryan Beal, Georgios Chalkiadakis, Timothy J Norman, and Sarvapali D Ramchurn. 2020. Optimising Game Tactics for Football. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*. 141–149.

[4] Ryan Beal, Georgios Chalkiadakis, Timothy J Norman, and Sarvapali D Ramchurn. 2021. Optimising Long-Term Outcomes using Real-World Fluent Objectives: An Application to Football. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*. 196–204.

[5] Ryan Beal, Timothy J Norman, and Sarvapali D Ramchurn. 2019. Artificial intelligence for team sports: a survey. *The Knowledge Engineering Review* 34 (2019).

[6] Daniel Berrar, Philippe Lopes, Jesse Davis, and Werner Dubitzky. 2019. Guest editorial: special issue on machine learning for soccer. *Machine Learning* 108, 1 (2019), 1–7.

[7] Roberto Cahuantzi, Xinye Chen, and Stefan Güttel. 2021. A comparison of LSTM and GRU networks for learning symbolic sequences. (2021).

[8] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*.

[9] Tom Decroos. 2020. *Soccer Analytics Meets Artificial Intelligence: Learning Value and Style from Soccer Event Stream Data.* Ph. D. Dissertation. KU Leuven.

[10] Tom Decroos, Lotte Bransen, Jan Van Haaren, and Jesse Davis. 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*. 1851–1861.

[11] Deepmind. 2021. *Advancing sports analytics through AI research.* Retrieved December 27, 2021 from https://deepmind.com/blog/article/advancing-sports-analytics-through-ai

[12] H Eggels, R van Elk, and M Pechenizkiy. 2016. Expected goals in soccer: Explaining match results using predictive analytics. In *The machine learning and data mining for sports analytics workshop*, Vol. 16.

[13] Jeffrey L Elman. 1990. Finding structure in time. *Cognitive science* 14, 2 (1990), 179–211.

[14] Eric Fosler-Lussier. 1998. Markov models and hidden Markov models: A brief tutorial. *International Computer Science Institute* (1998).

[15] Lucas Queiroz Gongora. 2021. *Estimating Football Position from Context (Master's thesis).* Master's thesis. KTH Royal Institute of Technology, Stockholm, Sweden.

[16] Steven Craig Hillmer and George C Tiao. 1982. An ARIMA-model-based approach to seasonal adjustment. *J. Amer. Statist. Assoc.* 77, 377 (1982), 63–70.

[17] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[18] Alireza M Javid, Sandipan Das, Mikael Skoglund, and Saikat Chatterjee. 2021. A relu dense layer to improve the performance of neural networks. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2810–2814.

[19] Michael I Jordan. 1997. Serial order: A parallel distributed processing approach. In *Advances in psychology*. Vol. 121. Elsevier, 471–495.

[20] Thom Lawrence. 2018. *Introducing xGChain and xGBuildup.* Retrieved December 27, 2021 from https://statsbomb.com/2018/08/introducing-xgchain-and-xgbuildup/

[21] Brian Macdonald. 2012. An expected goals model for evaluating NHL teams and players. In *Proceedings of the 6th MIT Sloan Sports Analytics Conference*.

[22] Rajeev Murgai, Robert K Brayton, and Alberto Sangiovanni-Vincentelli. 1991. On clustering for minimum delay/ara. In *1991 IEEE international conference on computer-aided design digest of technical papers*. IEEE Computer Society, 6–7.

[23] Opta. 2021. *Opta Advanced Metrics.* Retrieved December 27, 2021 from https://www.statsperform.com/opta-analytics/

[24] Luca Pappalardo, Paolo Cintia, Alessio Rossi, Emanuele Massucco, Paolo Ferragina, Dino Pedreschi, and Fosca Giannotti. 2019. A public data set of spatio-temporal match events in soccer competitions. *Scientific data* 6, 1 (2019), 1–15.

[25] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1986. Learning representations by back-propagating errors. *nature* 323, 6088 (1986), 533–536.

[26] Haşim Sak, Andrew Senior, and Françoise Beaufays. 2014. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. (2014).

[27] Thomas Seidl, Aditya Cherukumudi, Andrew Hartnett, Peter Carr, and Patrick Lucey. 2018. Bhostgusters: Realtime interactive play sketching with synthesized NBA defenses. In *Proceedings of the 12th MIT Sloan Sports Analytics Conference*.

[28] Ian Simpson. 2021. *Code Repository for Seq2Event PyTorch Implementation for Soccer.* https://github.com/statsonthecloud/Soccer-SEQ2Event

[29] Karun Singh. 2021. *Introducing Expected Threat (xT).* Retrieved December 27, 2021 from https://karun.in/blog/expected-threat.html

[30] Danilo Sorano, Fabio Carrara, Paolo Cintia, Fabrizio Falchi, and Luca Pappalardo. 2020. Automatic Pass Annotation from Soccer Video Streams Based on Object Detection and LSTM. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 475–490.

[31] William Spearman. 2018. Beyond expected goals. In *Proceedings of the 12th MIT Sloan Sports Analytics Conference*.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[33] Zhiwei Wang, Yao Ma, Zitao Liu, and Jiliang Tang. 2019. R-transformer: Recurrent neural network enhanced transformer. (2019).

[34] Neil Watson, Sharief Hendricks, Theodor Stewart, and Ian Durbach. 2021. Integrating machine learning and decision support in tactical decision-making in rugby union. *Journal of the Operational Research Society* 72, 10 (2021), 2274–2285.

[35] Qiyun Zhang, Xuyun Zhang, Hongsheng Hu, Caizhong Li, Yinping Lin, and Rui Ma. 2021. Sports match prediction model for training and exercise using attention-based LSTM network. *Digital Communications and Networks* (2021).

A REPRODUCIBILITY

A.1 Data Preparation

Use of the WyScout Open Access Dataset. The competitions, matches, teams and events JSON files were used for this research. Since this research was focused on team performance, player files were not used, but it is noted that this information is readily available to be incorporated into models for future research on player behaviour.

The events file records match events, and are classified according to 21 'event' categories and 78 'subevent' categories, with several hundred possible tags providing further detail. Tag types vary significantly in their scope of application across the dataset, in that some are specific and only used to elaborate one event/subevent type, whilst others are interpretable more generically.

Action Feature Engineering. A simplified encoding of match event attributes was necessary in order to define an associated categorical action label of acceptable dimensionality. If training on a far larger set, such reduction might not be necessary; the event, subevent, and tag information could be passed directly to the embedding layer from which the relevance of this information could be learnt by the model. However, for the relatively compact set used in this research, a reduction in dimensionality was considered necessary.

Qualitative analysis was first conducted to filter and group match events that appropriately characeterise attacking play. Making reference to the encodings performed by [9] [27] and reflecting on the objective of modelling attacking styles, an initial encoding was made as shown in Table 3. This encoding provided a rich sequential diversity, and gave an intuitive picture to the human analyst when observing data encoded in this way. However, the category support was too imbalanced or too weak in many cases, e.g. pass accounts for as much as 49.3% of events and penalty kick accounts for only 0.02% of events. Thus, a coarser encoding was deemed necessary.

For the 'final' project encoding, events were grouped into four categories: pass ('p'), dribble ('d'), cross ('x'), and shot ('s'). Three additional event types were added for possession and match context: goal scored ('g'), possession end ('_'), and match end ('@').

Continuous Feature Engineering. Event location co-ordinates and match time are provided in the source data. From these, additional features were engineered to help training by providing information on known sources of variance [9], and by providing context across different temporal scales (e.g. x, $thetag$, and $scrad$ change over time at different orders of magnitude). This resulted in ten normalised features, with spatial distribution shown in Figure 10, and defined as follows:

- $x, y \in [0, 1]$: event location co-ordinates.
- $deltax, deltay \in [-1, 1]$: difference in x, y since previous.
- $s \in [0, \sqrt{2}]$: distance since previous event.
- $sg \in [0, \sqrt{1.5}]$: distance to centre of opposition goal.
- $thetag \in [\pi/2, \pi]$: angle from centre of opposition goal.
- $T \in [0, 1]$: event match time.
- $deltaT$: time (normalised) since previous event.
- $scrad \in [-6, 6]$: current score advantage: number of goals ahead (positive) or behind (negative).

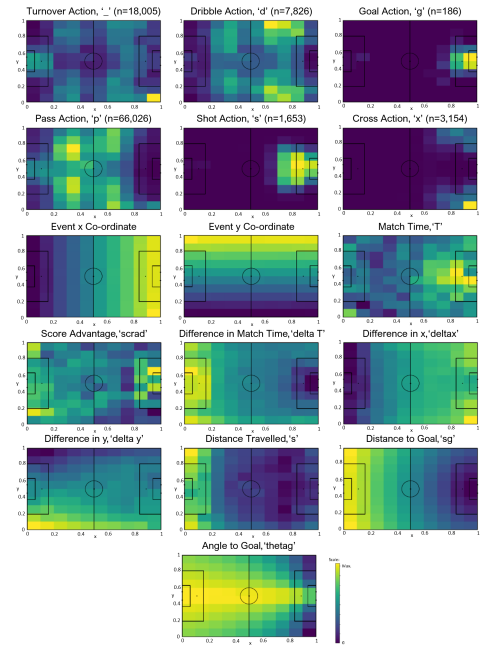


Figure 10: Spatial distribution of occurrence for all engineered source features (n=96,850). For action features, density of occurrence is shown. For continuous features, mean value is shown. Attacking direction is from left to right.

Defining Modelling Sets Sampling of all seven available competitions on high/moderate/low placing teams was performed to capture a representative sample of success, and is shown in Table 6. A 50/6/44 train/validation/test ratio was used with teams exclusive to train/validation and test sets. A high test weighting was used to ensure a representative cross-section of teams, season states and leagues.

A.2 Hyperparameter Search

Table 4 outlines the scheme used for hyperparameter search. An initial wide manual search was conducted, after which a focused combinatorial grid search was conducted on the hyperparameters shown in bold. Table 5 shows the hyperparameters of the top 10 *Seq2Event* models. The third best model, a Transformer variant, which had been trained on the data shown in Table 6 was then used to predict the events for the five La Liga teams over the whole season, for which results are presented in the main paper.

A.3 PyTorch Implementation of Model

Modelling was performed using PyTorch, and code for model reproduction is shared by us [28]. Two notebooks are presented: a data preparation and feature engineering notebook, and a modelling notebook.

Table 6: Scheme for Modelling Sets

Competition/Team/Position	Trg	Val	Tst
FIFA WC 2018	**12**	**1**	**11**
France (C)	3	1	-
Croatia (F)	-	-	2
Colombia (R16)	3	-	-
Denmark (R16)	-	-	3
South Korea (G3)	3	-	-
Tunisia (G3)	-	-	3
Panama (G4)	3	-	-
Poland (G4)	-	-	3
UEFA Euro 2016	**12**	**1**	**11**
Portugal (F)	3	-	-
Wales (SF)	-	-	3
Hungary (R16)	3	1	-
Switzerland (R16)	-	-	2
Republic of Ireland (R16)	3	-	-
Albania (G3)	-	-	3
Czech Republic (G4)	3	-	-
Sweden (G4)	-	-	3
English PL 2017/18	**9**	**1**	**8**
Manchester City (1)	3	-	-
Manchester United (2)	-	-	3
Newcastle United (10)	-	-	3
Crystal Palace (11)	3	1	-
Stoke City (19)	-	-	2
West Bromwich Albion (20)	3	-	-
French Ligue 1 2017/18	**9**	**1**	**8**
Paris Saint-Germain (1)	3	1	-
Monaco (2)	-	-	2
Montpellier (10)	-	-	3
Dijon (11)	3	-	-
Troyes (19)	-	-	3
Metz (20)	3	-	-
German Bundesliga 2017/18	**9**	**1**	**8**
Bayern Munich (1)	3	-	-
Schalke 04 (2)	-	-	3
Borussia Mönchengladbach (9)	-	-	2
Hertha BSC (10)	3	1	-
Hamburger SV (17)	-	-	3
1. FC Köln (18)	3	-	-
Spanish La Liga 2017/18	**9**	**1**	**8**
Barcelona (1)	3	-	-
Atlético Madrid (2)	-	-	3
Girona (10)	-	-	3
Espanyol (11)	3	-	-
Las Palmas (19)	-	-	2
Málaga (20)	3	1	-
Italian Serie A 2017/18	**9**	**1**	**8**
Juventus (1)	3	1	-
Napoli (2)	-	-	2
Sampdoria (10)	-	-	3
Sassuolo (11)	3	-	-
Hellas Verona (19)	-	-	3
Benevento (20)	3	-	-
Total	69	7	62

Table 3: WyScout Event to Project Encoding Mapping

WyScout Event/SubEvent Description	Project Encoding (initial)	Project Encoding (final)	Proportion of all events	
Pass (Hand pass)	p		0.43%	
Pass (Head pass)	p		2.98%	
Pass (High pass)	p		4.01%	
Pass (Launch)	p		1.41%	
Pass (Simple pass)	p	p	39.55%	
Pass (Smart pass)	p	(Pass)	0.93%	56.1%
Others on the ball (Clearance)	o		1.75%	
Free Kick (Goal kick)	0		0.98%	
Free Kick (Throw in)	1		2.62%	
Free Kick (Free Kick)	3		1.40%	
Duel (Ground attacking duel)	d	d	8.61%	
Others on the ball (Acceleration)	t	(Dribble)	0.80%	14.8%
Others on the ball (Touch)	t		5.37%	
Pass (Cross)	x	x	1.92%	
Free Kick (Corner)	2	(Cross)	0.59%	2.8%
Free Kick (Free kick cross)	4		0.27%	
Shot (Shot)	s	s	1.32%	
Free Kick (Free kick shot)	5	(Shot)	0.07%	1.4%
Free Kick (Penalty)	6		0.02%	
Foul (Foul)	f		1.45%	
Foul (Hand foul)	f		0.06%	
Foul (Late card foul)	f		0.01%	
Foul (Out of game foul)	f		0.02%	
Foul (Protest)	f		0.02%	
Foul (Simulation)	f		0.00%	
Foul (Time lost foul)	f		0.01%	
Foul (Violent Foul)	f	n/a	0.00%	
Offside (no SubEvents)	f	(Foul or	0.25%	25.0%
Duel (Air duel)	n/a	Defensive	5.18%	
Duel (Ground defending duel)	n/a	Actions)	8.57%	
Duel (Ground loose ball duel)	n/a		4.67%	
Gkpr leaving line (Gkpr leaving line)	n/a		0.19%	
Interruption (Ball out of the field)	n/a		3.97%	
Interruption (Whistle)	n/a		0.03%	
Save attempt (Reflexes)	n/a		0.33%	
Save attempt (Save attempt)	n/a		0.21%	

Table 4: Scheme for Hyperparameter Search

Hyperparameter	Values *(bold=focused search)*
Model Type	Elman-RNN, **LSTM**, **GRU**, **Transformer**
Sequence Length	5, **10**, **40**, **100**
Step length	1, **20**
Action Embed. Dim.	1, 5, **7**, 20
Cont. Feat. Embed. Dim.	1, 3, 5, **9**, 10, 20
Number of layers	**1**, **2**, **5**
Hidden/Feedforward Dim.	**8**, **16**, 256, 1024, 2048, **4096**, **8192**, 16384, 32768
RNN Directionality	**Uni-Directional**, **Bi-Directional**
Transformer Num. Heads	**1**, 2, **17**

Table 5: Models by Test Loss: Top 10 plus Selected Baselines

Variant	Seq. Len.	Step	Act. Emb. Dim.	Cont. Emb. Dim.	Num. Lyrs.	Hdn. Dim.	Dir. /Num Heads	Loss (Rank)
LSTM	100	1	7	10	1	8	UNIDIR	0.332 *(1)*
LSTM	100	1	7	10	1	8	BIDIR	0.344 *(2)*
Transformer	40	1	7	10	1	8	1	0.362 *(3)*
Transformer	40	1	7	10	1	16	1	0.364 *(4)*
Transformer	10	20	3	5	1	8192	1	0.368 *(5)*
Transformer	10	20	1	20	1	8192	1	0.37 *(6)*
Transformer	10	20	1	5	1	8192	1	0.372 *(7)*
Transformer	10	20	5	3	1	8192	1	0.373 *(8)*
Transformer	10	20	5	5	1	8192	1	0.373 *(9)*
Transformer	10	20	3	3	1	8192	1	0.373 *(10)*
Baseline: AR+MC Order 1	1	1	7	2	-	-	-	0.667 *(146)*
Baseline: AR+MC Order 5	5	1	7	2	-	-	-	0.679 *(152)*
Baseline: Lag 1 (t=t-1)	1	1	7	2	-	-	-	0.708 *(156)*