A Scalable Approach for Unified Large Events Models in Soccer

Tiago Mendes-Neves[1,2] (✉), Luís Meireles[2], and João Mendes-Moreira[1,2]

[1] Faculdade de Engenharia da Universidade do Porto, Porto, Portugal
{tiago.neves@fe.up.pt}
[2] LIAAD - INESC TEC, Porto, Portugal

Abstract. Large Events Models (LEMs) are a class of models designed to predict and analyze the sequence of events in soccer matches, capturing the complex dynamics of the game. The original LEM framework, based on a chain of classifiers, faced challenges such as synchronization, scalability issues, and limited context utilization. This paper proposes a unified and scalable approach to model soccer events using a tabular autoregressive model. Our models demonstrate significant improvements over the original LEM, achieving higher accuracy in event prediction and better simulation quality, while also offering greater flexibility and scalability. The unified LEM framework enables a wide range of applications in soccer analytics that we display in this paper, including real-time match outcome prediction, player performance analysis, and game simulation, serving as a general solution for many problems in the field.

Keywords: Large Events Model · Sports Analytics · Deep Learning · Generative Model.

1 Introduction

Large Events Models (LEMs) [11] are an innovative concept in soccer analytics, drawing inspiration from the success of Large Language Models (LLMs). Just as LLMs predict the next word in a sequence based on the context of previous words, LEMs are engineered to predict the next event in a soccer match, given the current game state. These events include discrete actions such as passes, shots, fouls, and more, collectively forming the "language" of soccer.

LEMs leverage deep learning techniques and are trained on extensive datasets of soccer event data. This training enables them to identify patterns and sequences in gameplay, allowing them to simulate entire soccer matches from a specified starting point or predict the likelihood of specific events occurring next. The goal of LEMs is to solve one of the long-standing limitations of traditional soccer analytics models: existing models often lack flexibility, i.e., they are designed for specific tasks and require redevelopment for new applications.

In Mendes-Neves et. al. [11], the authors employ a chain of classifiers approach to build a LEM. This method involves a sequence of multiple classifiers working together, where each classifier focuses on predicting a specific aspect of

the upcoming event, using the outputs of the previous classifiers as additional input. The chain begins with predicting the next event type (such as a pass, shot, or foul) using the current game state (e.g., previous event, ball location, ...). The second classifier then takes the predicted event type plus the game state to determine the event's accuracy (whether it will be successful and whether it will result in a goal). Finally, the third classifier uses all prior predictions (event type, accuracy, and goal outcome) alongside the game state to forecast additional details, including the time elapsed until the event occurs, its location on the field (X and Y coordinates), and whether the home team will perform it. By structuring the prediction process this way, the chain of classifiers captures how the event type influences its likelihood of success and other attributes, leading to a more accurate model of soccer match dynamics.

Although the chain of classifiers architecture effectively captures the core elements of a soccer event, this approach exhibited several drawbacks. Firstly, the discontinuous nature made fine-tuning complex, as each classifier required individual adjustments [10], leading to a cumbersome process when adapting the model to specific teams or players. Secondly, scaling the model proved challenging. Different analytical tasks might necessitate different model sizes, but the chain structure required synchronized scaling across all modules. Finally, the parallel nature of some predictions within the architecture meant that specific components did not leverage the full context of the event sequence.

These limitations show the need for a more streamlined and integrated approach to modeling soccer event data. In this paper, we explore the potential of a unified LEM. Using a causal masking strategy, we unify the LEM by predicting the next event in a tabular format. The central idea driving this unified approach is to design a system capable of sequentially predicting each element of a soccer event. This sequential prediction process, even if it requires multiple inference steps for a single event, allows the model to incorporate the full context of preceding event elements. In the appendix of this paper, we also document a strategy to treat the problem as a language modeling task, with inferior results to our approach.

This paper is organized as follows. Section 2 reviews related work in soccer analytics and generative modeling, highlighting the gaps our approach addresses. Section 3 details our experimental setup, including data preparation and the methodology. Section 4 presents our results, evaluating model performance across prediction accuracy and simulation fidelity, with applications demonstrated in Section 5. Finally, Section 6 concludes with a summary of findings and directions for future work. Appendices provide supplementary details on datasets and alternative modeling approaches.

2 Related Work

Soccer analytics has witnessed a remarkable evolution, driven by models tailored to dissect various facets of the game, addressing a spectrum of tasks like valuing discrete actions, players, and teams. For example, expected goals (xG) models

predict the likelihood of a shot resulting in a goal based on location, angle, and other contextual factors like defensive pressure [13, 1]. Frameworks like Valuing Actions by Estimating Probabilities (VAEP) [4] and expected threat (xT) [15, 8] extend this concept by assessing the broader impact of actions on scoring or conceding probabilities. Meanwhile, models leveraging tracking data analyze player movements offering insights into off-ball contributions and team formations [17, 5]. This diversity underscores the complexity of soccer, necessitating specialized tools for distinct analytical objectives.

The advent of LLMs has demonstrated the power of large-scale, self-supervised models to address a wide range of tasks within a single framework [14, 3]. LLMs learn data representations from vast, unlabeled datasets, enabling them to generalize across applications (e.g., text generation, translation, and question answering). This paradigm shift offers a compelling analogy for sports analytics. In soccer, current methods often rely on task-specific models. The success of LLMs suggests that a sequence model could serve as a "foundation" for soccer analytics, capable of modeling event sequences and adapting to diverse downstream tasks without specialist solutions.

Before the introduction of LEMs, generative modeling in soccer analytics was limited to narrow scopes. For instance, Seq2Event [16] employed transformers and recurrent neural networks to predict the next event in a match sequence but limited itself to passes, dribbles, crosses, and shots. TacticAI [20] utilized graph neural networks to model player interactions, predicting outcomes and suggesting tactical adjustments, but limited to corner kicks. There are other efforts [2], but all require a limited action set to compromise. These shortcomings set the stage for developing LEMs, which aim to provide a comprehensive generative framework.

LEMs was a pioneering approach to generatively model soccer events in a holistic manner. Inspired by LLMs, LEMs sought to learn the underlying probability distribution of event sequences, enabling realistic simulation and prediction. LEMs adopted a chain of classifiers to model multiple event attributes within a unified framework. This approach covered a significant portion of the SPADL schema [4] (excluding identifiers), offering a more complete representation of soccer dynamics than its predecessors while modeling 33 event types. This was a significant improvement over existing proposals.

Despite their innovations, LEMs exhibited several drawbacks that limit their effectiveness and scalability, which we seek to address in this paper. The reliance on a chain of classifiers introduces architectural complexity, requiring the training (and post-training) to be executed three times, upon which there is a necessity to verify if the models have learned coherent patterns among themselves. The context window of the model is also limited, only using a single event to predict the following.

3 Experimental Setup

3.1 Data

For training our models, we used data from the 2015-2016 to 2021-2022 seasons for the first and second leagues of Portugal, Spain, Germany, and France, as well as the first leagues of Denmark and Belgium. We selected these leagues due to their high level of competitiveness and availability. We used the 2022-2023 season of the same leagues for validation purposes, with 100 000 randomly sampled instances reserved for validation during model training and 15% of the remaining validation set used to evaluate the models. The 2023-2024 season is used for testing applications in Section 5. A more in-depth description of the datasets is available in Table 1.

The work is also reproducible using publicly available datasets, such as Pappalardo et. al. [12] or Statsbomb Free Data[3]. However, because companies utilize different data standards to annotate events, there may be differences in how the models perceive these events. Nonetheless, the quality and depth of the underlying data has an effect on the model quality.

3.2 Deviations from the Original Dataset

In contrast to Wyscout's original grouping, with several event types under broad categories, we refined and expanded some of these event types. The changes we made are the following:

- Wyscout groups multiple types of duels under the "duels" event type. We split this category into five distinct types of duels: "defensive duel," "offensive duel," "aerial duel," "loose ball duel," and "dribble."
- We separated the "passes" category into three subcategories: "pass," "long pass," and "cross."
- For "shot," we differentiated based on the part of the body used: "right-footed shot," "left-footed shot," and "headshot."
- We differentiated the "shot against" event type by changing it to "save" when a goalkeeper made a save.
- Cards were previously associated with an "interruption" event. We now explicitly distinguish between "yellow card" and "red card" events.
- Some events also have an associated "carry." A carry occurs when a player moves the ball from where they received it to a new position before executing another action or being interrupted by a duel. To improve the accuracy of our models and given the importance of spatio-temporal aspects in soccer, we added an event each time a carry is associated with another event. The carry event follows the event with which it was initially associated.
- We also introduced two new event types: the "first half end" and "game end." These events help in modeling when game simulations should terminate, fixing the issue with the original proposal where all games ended exclusively

[3] https://statsbomb.com/what-we-do/hub/free-data/

based on time. This extends the time limit where games are forced to end from 90 to 99 minutes, with the first half extending from 45 to 49 minutes

To reduce the number of inputs in our model, we computed two new variables:

- Accurate: In the dataset, different event types have different indicators of success. Since they are independent, we merged them into a single variable to reduce the number of variables being forecasted. An event is considered accurate if it meets the following criteria:
  - the event is a pass and is accurate
  - a player is the first to touch the ball in an aerial duel
  - a player successfully progresses with the ball in an offensive duel
  - a player recovers possession in a ground duel
  - the event is a carry that leads to progression on the field
  - the event is a shot that results in a goal
- Time elapsed: Time in an event is described by two variables: minute and second. We compute the time difference in seconds between two events to reduce them to a single variable. Then, we clip the "time elapsed" variable to a maximum value of 100, ensuring that we do not require extra tokens to manage larger values. This extends from the original proposal that capped the "time elapsed" at 60 seconds.

The new "accurate" variable now carries both the information of the *isAccurate* and of *isGoal* from the original architecture [11]. This provides an improvement by reducing the number of variables to forecast from 7 to 6. While previously, in the chain of classifiers approach, this was not significant since both variables are predicted with a single step, in our new approach, reducing the number of variables is important as each variable is inferred individually.

3.3 Statistical Description of the Dataset

For the rest of this paper, we will use the following abbreviations for the variables included in the models.

hy → Home team yellow cards.
ay → Away team yellow cards.
hr → Home team red cards.
ar → Away team red cards.
hg → Home team goals scored.
ag → Away team goals scored.
p → True if period is second half.
m → Minute.
s → Second.
h → True if the event was made by the home team.
e → Event type.
x → x-coordinate of the event on the field.
y → y-coordinate of the event on the field.
t → Time elapsed since previous event.
a → Accurate.

Table 1. Descriptive statistics of the processed Wyscout dataset.

		Set		
Variable		**Train**	**Validation**	**Test**
General				
Events	#	39,580,286	6,036,590	5,955,852
Matches	#	22,773	3,352	3,256
Events per Match	Mean	1,738	1,800	1,829
Home Team (h)				
Majority Class		1	1	1
Majority Class	%	0.51	0.51	0.51
Event Type (e)				
Unique Values	#	32	32	32
Majority Class		pass	pass	pass
Majority Class	%	0.36	0.38	0.38
Coordinates (x, y)				
Unique Values	#	101	101	101
x	Mean	47.58	47.13	47.07
y	Mean	49.63	50.33	50.19
Time (t)				
Unique Values	#	101	101	101
Majority Class		2	2	2
Majority Class	%	0.22	0.22	0.23
t	Mean	3.27	3.19	3.20
Action Type (a)				
Majority Class		1	1	1
Majority Class	%	0.51	0.5	0.51

3.4 Reshaping

We reshaped the dataset to fit in a tabular format. We created six copies of each event, each with a target for each of the h, e, x, y, t, and a variables. Some inputs will be masked as -1 depending on the target variable to hide future information from the model. For example, to predict the y variable, we mask y and all subsequent variables (t and a). When we aim to predict the first variable h, we mask all variables. Listing 1.1 shows a sample of the data in this format.

In addition to the event variables we aim to predict, we have contextual variables such as the current goals scored for home and away team (hg and ag), along with red and yellow cards (hr, ar, hy, ay). All other variables preceded by a c refer to contextual variables extracted from previous events.

Listing 1.1. A tabular dataset sample for sequence size 3. The first row indicates column headers while subsequent rows show example event records.

```
id,h,e,x,y,t,a,p,m,s,hg,ag,hr,ar,hy,ay,c1_h,c1_e,
   c1_x,c1_y,c1_t,c1_a,c2_h,c2_e,c2_x,c2_y,c2_t,
   c2_a,c3_h,c3_e,c3_x,c3_y,c3_t,c3_a,target
235200910,1,0,64,71,3,-1,0,40,42,0,0,0,0,0,0,0,6,
   13,36,2,0,1,18,100,100,29,0,1,12,100,68,9,0,1
192308161,1,0,29,25,-1,-1,1,69,13,0,1,0,0,1,1,1,0,
   10,61,5,1,1,24,0,0,5,0,0,0,27,43,3,0,4
...
```

3.5 Tabular Modeling with Multilayer Perceptrons

In our proposal, we model soccer events as a tabular problem, where each row represents an event and columns represent features of that event. The core idea is to autoregressively predict each event in a sequence, using the context of previous events and the current game state. We will refer to this approach as Multilayer Perceptrons Large Events Model (MLP LEM).

Data Flow The data flow in MLP LEMs is presented in Figure 1.

The input data for LEMs consists of three main components:

1. Game Context: represents the global state of the game at any given time. These features provide crucial context for the model, reflecting each team's overall performance and situation and influencing the likelihood of different events.
   It includes (hg, ag, hy, ay, hr, ar).
2. Previous Event Sequence: provides a localized context, capturing the immediate history leading to the current event. The model receives a fixed-length sequence of the n most recent events. Each event in the sequence is represented using a six-token format, just like the current event.

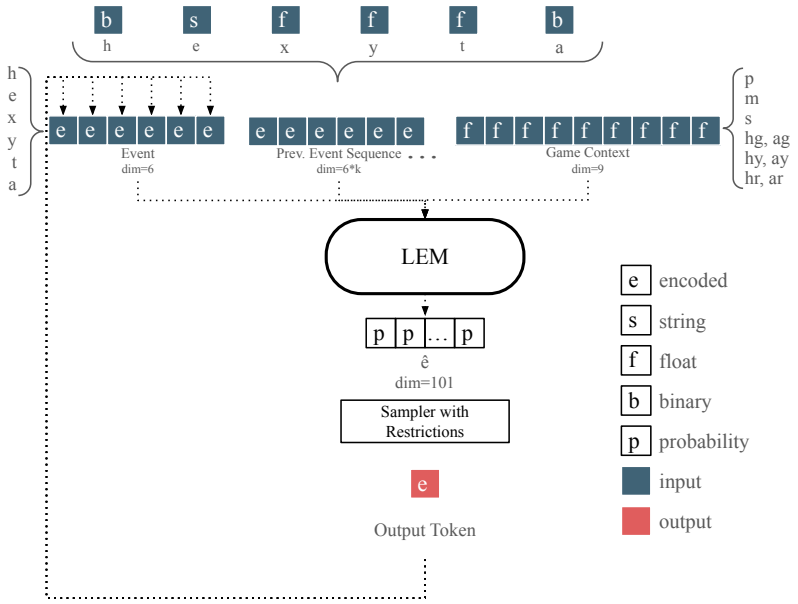

Fig. 1. Data flow in MLP LEMs. The model inputs the game context, previous events, and the current event to predict the next event in the sequence. Each event is represented by six tokens: Team, Event Type, Elapsed Time, Start X, Start Y, and Accuracy. The model predicts each token sequentially, updating the current event representation with each prediction.

3. Event: represents the event being predicted at each step, containing the information of which tokens are masked and require prediction.

The LEM predicts each event autoregressively, one token at a time. Each token is a part of an event which is encoded to enable modeling. This means the model predicts each token of the current event sequentially, conditioning each prediction on the game context, previous events, and the previously predicted tokens. Initially, when all tokens of the current event are unknown, the current event vector is initialized with an "unknown" token (represented as -1 in our implementation) for each token. The prediction process then unfolds sequentially as follows: h, e, x, y, t, and a.

Like the original proposal [11], the model does not simply classify the most likely token at each step. Instead, it outputs a probability distribution over the possible values for each token. This allows us to incorporate randomness into the event-generation process through sampling.

Training Procedure The raw event data is transformed into a numerical format suitable for the neural network using tokenization, where each feature of an event is mapped to a discrete numerical token. This tokenization process resulted in a vocabulary size of 101, tokenized as follows:

- The spatial features, x and y, representing the coordinates of an event on the field, were discretized into 101 tokens each, representing values from 0 to 100.
- The temporal feature, t was also discretized into 101 tokens. Since the original data could go up to 30 minutes without an event occurring, we condensed the range to reduce the vocabulary size while maintaining sufficient temporal resolution for normally occurring games.
- The e feature was encoded using 34 tokens, representing the different types of events in the dataset. We reused the tokens 0-33 to avoid increasing the output space.
- The t and a features were represented with two tokens (0 and 1).

We experimented with MLP architectures of varying depths and widths to explore the impact of model size on performance. After preliminary experiments, we selected 7 architectures that scale their parameter count exponentially (10k, 30k, 100k, 300k, 1M, 3M, and 10M). The configurations presented in Listing 1.2 were extensively tested.

Listing 1.2. Model architectures for MLP LEMs.

```
MLP(input_size=get_size(seq_len), hidden_sizes=[  80,           ], output_size=101)
MLP(input_size=get_size(seq_len), hidden_sizes=[  96,   96,   96], output_size=101)
MLP(input_size=get_size(seq_len), hidden_sizes=[ 196,  196,  196], output_size=101)
MLP(input_size=get_size(seq_len), hidden_sizes=[ 360,  360,  360], output_size=101)
MLP(input_size=get_size(seq_len), hidden_sizes=[ 682,  682,  682], output_size=101)
MLP(input_size=get_size(seq_len), hidden_sizes=[1200, 1200, 1200], output_size=101)
MLP(input_size=get_size(seq_len), hidden_sizes=[2220, 2220, 2220], output_size=101)
```

The models were trained using the Adam optimizer [9], with an initial learning rate of 0.01. We employed the binary cross-entropy with logits loss function, with a batch size of 1024. A dropout rate of 0.3 was used on every layer of all models for regularization. During training, we measured the model's performance in the validation set. Due to the computational cost of evaluating the entire validation set, a representative sample of 100 000 data points was used for evaluation. For scaling law experiments, we use 25% of the train data. Subsequently, the best models were trained using the entire dataset for 4 epochs. The code to reproduce our training process is available on Github[4].

4 Results

4.1 Scaling Laws

We conducted initial experiments to investigate the relationship between model size, sequence length, and performance using a subset of the training data. We trained a series of MLP models as described in Section 3.5. The validation loss curves for these models are presented in Figure 2.

The results demonstrate that a sequence length of 3 consistently yielded the lowest validation loss across different model sizes. This suggests that capturing the immediate context of the three preceding events provides the most valuable information for predicting the subsequent event.

[4] https://github.com/nvsclub/LargeEventsModel/

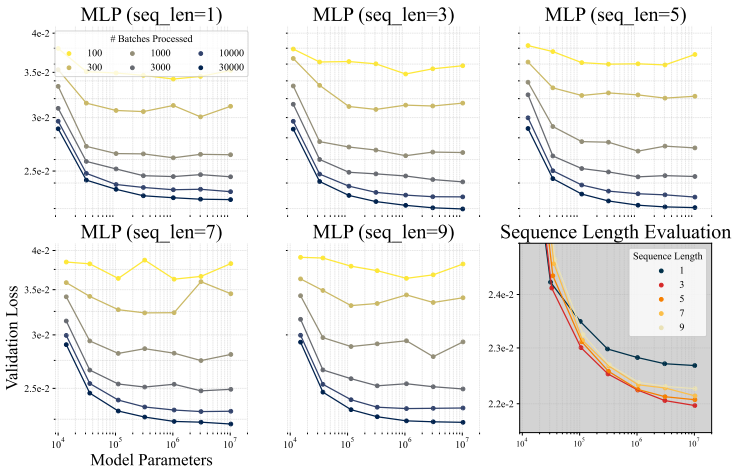

Fig. 2. Validation loss of MLPs trained with varying sequence lengths and model parameters. Each plot represents the test performed at a different sequence length, and each line represents the loss at different points during the training process.

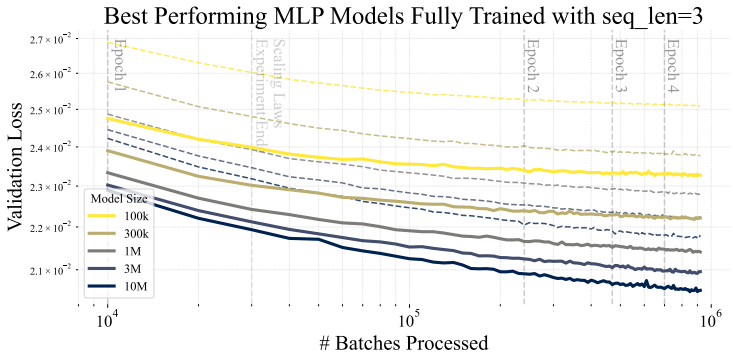

Fig. 3. Validation loss curves for the best-performing MLP models, all using a sequence length of 3. The validation loss continues decaying as the data increases past the previous scaling laws experiment, although the rate of decrease is shrinking. Dotted lines represent train losses.

4.2 Training on the Full Dataset

Based on the insights gained from the scaling law experiments, we selected the 100k, 300k, 1M, 3M, and 10M models with a sequence length of 3 and trained them on the full dataset. The learning rate was reduced to 0.001 for this stage of training to prevent overshooting the optimal solution. The validation loss curves for these models are shown in Figure 3.

The validation loss curves indicate that the models continued improving, achieving lower loss values than the scaling law experiments. The trend in the

validation loss curves suggests that the models might benefit from further training. It is possible that, with more training epochs, the models could achieve even better performance, as the slope of the curves indicates that we are yet to convergence. Nonetheless, the returns are diminishing as the scale of resources to increase performance increases exponentially. This decay in performance increase is observable in the performance metrics presented in Table 2.

Table 2. Performance metrics for MLP models on the validation set. Our new approach outperforms the original approach (OG LEM) at nearly every scale. All variables where the performance increase is not visible reflect significant modifications on the new model that put our new proposal at a disadvantage.

	OG LEM	**MLP 100k**	**MLP 300k**	**MLP 1M**	**MLP 3M**	**MLP 10M**
Accuracy						
h (Team)	0.938[a]	0.855	0.864	0.871	0.873	0.874
e (Event Type)	0.557	0.644	0.656	0.664	0.667	0.670
a (Accuracy)	0.817[b]	0.852	0.856	0.860	0.860	0.861
F1-Score						
h (Team)	0.938[a]	0.855	0.864	0.871	0.873	0.874
e (Event Type)	0.499	0.577	0.591	0.603	0.608	0.612
a (Accuracy)	0.873[b]	0.852	0.856	0.859	0.860	0.861
R^2						
x	0.636	0.812	0.836	0.851	0.857	0.858
y	0.292	0.435	0.571	0.603	0.625	0.651
t (Time Elapsed)	0.552[c]	0.153	0.408	0.420	0.447	0.464
MAE						
x	8.5	6.945	6.308	5.919	5.775	5.695
y	15.6	13.526	11.330	10.648	10.233	9.804
t (Time Elapsed)	2.6[c]	1.735	1.515	1.475	1.427	1.401
Inference Time (seconds per 150 000 tokens)						
Any variable	0.023	0.011	0.019	0.048	0.130	0.419

[a] In the original proposal, the h variable was predicted in the last step and, in this iteration, is the first variable to be predicted, having less information to work with. Therefore, this value is inflated.

[b] In the original proposal, the a variable was predicted in the second step and, in this iteration, is the last variable to be predicted, having more information to work with. Integrating the goal variable into a also inflates the accuracy of our newer models.

[c] The time elapsed variable ranged between 0 and 60 in the original proposal, while it now ranges from 0 to 100.

MLP models can learn meaningful patterns from event data better than our original proposal. The larger models generally outperform the smaller ones, but the gains diminish with increasing size. The best model choice depends on the specific application and the acceptable trade-offs between performance, inference time, and computational resources. The largest model, 10M, provides the best overall performance across the board but at the cost of significantly higher inference times. A smaller model might be more suitable for applications where speed is critical, offering a good balance between performance and efficiency.

4.3 Benchmarking Large Events Models

Evaluating the quality of our LEMs based on their accuracy in predicting individual tokens is not enough. Token-level accuracy alone does not fully capture the capabilities of a generative model, particularly for downstream tasks like simulating entire soccer matches. While a high token prediction accuracy is desirable, it does not guarantee that a model can generate coherent sequences of events that accurately reflect the dynamics of a real game. We evaluate our models' ability to simulate full soccer matches and compare the statistical properties of these simulated matches (10,000 simulations from kickoff to the end of the match) to real-world data, focusing on three key metrics: goals scored by the home team (Home Goals), by the away team (Away Goals) and the difference between the scores of each team (Goal Difference).

We compute a distance metric based on the element-wise differences to quantify the similarity. Specifically, for each metric (home goals, away goals, goal difference), we calculate the absolute difference between the corresponding elements of the predicted and expected distributions and then sum these differences. Formally, let $[p_1, p_2, ..., p_n]$ be the array representing the predicted distribution and $[e_1, e_2, ..., e_n]$ be the array representing the expected distribution. The distance D calculation is formalized in Equation 1.

$$D = \sum_{i=1}^{n} |p_i - e_i| \tag{1}$$

Benchmarking Results Table 3 presents the results on our benchmark.

Interestingly, we observe that the best performing MLP models (e.g., MLP 10M) according to Table 2 exhibit poor simulation capabilities despite achieving the best performance in token prediction. Their distance scores are significantly higher than small models, indicating a divergence between token-level accuracy and the ability to generate realistic match outcomes. On the other hand, smaller models exhibit the best overall simulation performance. These models show a good balance between token-level accuracy and generative capacity. Furthermore, the results suggest that, in some cases, earlier training epochs can yield better simulation results than later epochs. This might be because earlier epochs retain more "uncertainty" or stochasticity in their predictions, which can be beneficial for generating diverse and realistic sequences of events.

Table 3. The simulation error of MLP models at their best epochs. The error is measured as the distance between simulated and actual soccer match outcomes. Lower distances indicate better simulation accuracy, reflecting a closer alignment.

Model	Epoch	Goal Delta Distance	Home Goal Distance	Away Goal Distance	Total Distance
MLP 100k	3	0.081	0.057	0.050	0.188
MLP 300k	3	0.076	0.022	0.067	0.165
MLP 1M	1	0.061	0.082	0.034	0.177
	2	0.088	0.045	0.136	0.269
MLP 3M	3	0.067	0.079	0.084	0.230
MLP 10M	1	0.117	0.354	0.322	0.793

Figure 4 visualizes the distribution of goal differences for real matches and compares it to the simulated distributions. We observe a close alignment between almost all distributions, indicating that any model can generate realistic simulations of events in soccer matches.

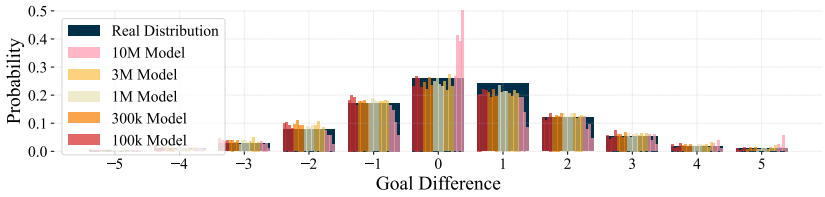

Fig. 4. Comparison of goal difference distributions between real matches and MLP simulations across different epochs (in order epoch 1, 2, 3, and 4). Each bar represents the number of simulations ending with the respective goal difference.

5 Applications of Large Events Models

The potential of LEMs lies in their broad applicability to various problems in soccer analytics. This section demonstrates applications, showcasing how LEMs can provide novel insights. All applications presented here utilize the MLP 100k model, the smallest fully-trained model introduced in this paper. These examples demonstrate that even a relatively small LEM can offer significant value.

5.1 Measuring Performance

Estimating Shot Efficiency While xG models have become common in football analytics, they primarily focus on a single aspect of the shot. LEMs offer a

more granular approach, allowing us to analyze more aspects of shots, like the shot efficiency. This metric assesses how effectively players convert their involvement in the game into shots taken, providing a complementary perspective to traditional expected goals analysis.

We mask the e and subsequent variables (x, y, t, and a) to estimate the probability of each event being a shot. By attributing this probability to each event, we calculate the chance of each event being a shot. Note that no fine-tuning is required in this approach. We are using the simulator to ask, "What is the probability that this situation leads to a shot?" for all actions in a soccer match. The answer contains a probability calculated using the average behaviors of our training set. To measure the behavior of specific players, we aggregate the probabilities across all events where the player was involved and compare it with the actual number of shots taken. These results are presented in Table 4, which analyzes individual player shot efficiency, contrasting the expected shot count with the actual shots taken during the 2023/24 Portuguese First League season. Jota Silva and Pedro Gonçalves exhibit the most significant positive deviations, with a delta of +35.38 and +32.97, suggesting an exceptionally high propensity to take shots. Similarly, V. Gyökeres takes approximately 16 more shots than anticipated, which is particularly noteworthy given his high line for expected shots. In opposition, players such as Pepê Aquino and João Mário have negative deltas of -11.34 and -13.01, respectively. This could indicate a more selective approach to shooting, possibly prioritizing higher-quality opportunities or reflecting a tendency to opt for passes or dribbles over shots in certain situations. It may also be associated with a lack of confidence in their finishing abilities.

Table 4. Player shot data for the 2023/24 season of the Portuguese First League. The delta column represents the difference between actual shots taken and the expected number of shots based on LEM predictions. A positive delta (highlighted in blue) suggests a player is taking more shots than expected, given their involvement in the game, potentially indicating a shoot-first mentality. Conversely, a negative delta (highlighted in red) could suggest a more hesitant approach to shooting.

Team Name	Player Name	Expected Shots	# Shots	Delta
Sporting CP	V. Gyökeres	86.18	102	15.82
Porto	Francisco Conceição	70.70	76	5.30
Benfica	Rafa Silva	66.34	83	16.66
Benfica	Á. Di María	65.00	92	27.00
Porto	Pepê Aquino	62.34	51	-11.34
Porto	Galeno	56.20	75	18.80
Vitória Guimarães	Jota Silva	55.62	91	35.38
Sporting CP	Pedro Gonçalves	54.03	87	32.97
Benfica	João Mário	54.01	41	-13.01
Vízela	S. Essende	51.81	76	24.19

Quantifying Accumulated Pass Risk Traditional pass accuracy metrics often fail to capture the true value of a player's passing ability. A player can achieve high pass accuracy by playing safe passes in his half that contribute little to advancing the team's attack. Metrics like ball possession have been criticized as poor indicators of proactive or effective play.

LEMs offer a way to quantify pass risk by estimating the probability of success for each pass based on its contextual factors. This allows us to calculate a player's *Expected Passes Completed* (EPC), representing the number of passes a player is expected to complete successfully, given the difficulty of the passes attempted. By comparing a player's actual completed passes to their EPC, we obtain a more informative measure of their passing ability and risk-reward assessment that accounts for the risk and value associated with each pass.

Table 5. EPC results for the 2023/24 season of the Portuguese First League. EPC represents the number of passes a player is expected to complete, given the difficulty of their attempts. Delta highlights the difference between accurate passes and EPC.

Player Name	Team Name	EPC	Accurate Passes	Delta
João Neves	Benfica	1550	1688	138
A. Varela	Porto	1326	1446	120
António Silva	Benfica	1403	1519	116
João Mário	Benfica	1279	1395	116
João Moutinho	Sporting Braga	1299	1410	111
Gonçalo Inácio	Sporting CP	1920	2022	102
Diogo Nascimento	Vizela	1030	1132	102
N. Otamendi	Benfica	1456	1555	99
F. Aursnes	Benfica	1346	1440	94
O. Diomandé	Sporting CP	1425	1513	88

Table 5 presents the EPC and delta for players in the 2023/24 Portuguese League season. Players in this table consistently exceed their EPC, indicating they are completing more passes than expected for the risk they are taking. For instance, João Neves of Benfica has a delta of +138, demonstrating his ability to successfully execute passes on a risk-adjusted basis. The metric offers a more nuanced and insightful approach to evaluating passing performance, highlighting players who effectively balance risk and reward in their passing game.

5.2 Game Simulation

As introduced in the original proposal [11], a key application of LEMs is the simulation of soccer matches. We distinguish between two primary types of simulations: short-term and long-term. Short-term simulations focus on predicting events within a limited horizon, such as forecasting the likelihood of a goal within the following ten events. Conversely, long-term simulations aim to model an en-

tire match from a specific point until its conclusion, enabling the estimation of match outcomes like the final score or the probability of a win.

Short-term Short-term simulations allow for a granular analysis of game dynamics by predicting the immediate consequences of specific in-game situations. At the most basic level, we can simulate a single subsequent event to understand the likely progression of play. By simulating the specified number of subsequent events numerous times and calculating the percentage of simulations in which a goal was scored, we can calculate the probability of a goal in the short term. Figure 5 demonstrates this capability by visualizing the short-term goal-scoring probability throughout a match. Note that these probabilities are crucial for action valuation methods such as VAEP [4].

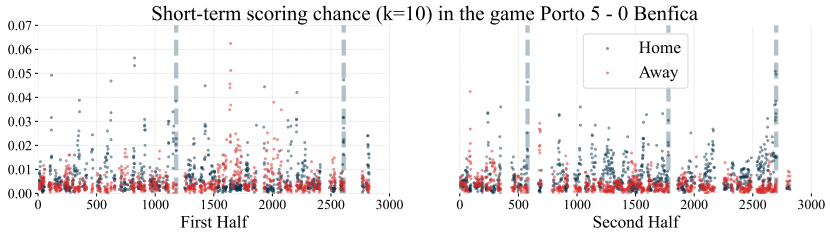

Fig. 5. Short-term probability forecasting, i.e., the chance of scoring within 10 events during the Porto - Benfica, 5 - 0, March 3, 2024.

Long-term LEMs can simulate full matches by iteratively feeding the model's output as input, allowing it to generate the next event in the sequence until a terminal state (e.g., the end of the match or a certain number of events). This opens a wide range of analytical possibilities. The generated event sequences are fully compatible with existing event data analysis workflows, provided they operate within the feature space encompassed by LEMs. A key application of long-term simulations is the real-time estimation of match outcome probabilities. In Figure 6, we present examples of such estimations for three different matches. For each game, we initialized 10,000 simulations from each event. Each point in the plot represents the probability of each outcome at the time of each event.

6 Conclusion

We introduced a unified LEM framework that advances soccer analytics by replacing the original chain-of-classifiers approach with a tabular autoregressive model. Our framework delivers superior predictive accuracy and simulation quality. Key findings show that our unified LEM not only outperforms its predecessor in event prediction and simulation fidelity but also achieves an optimal

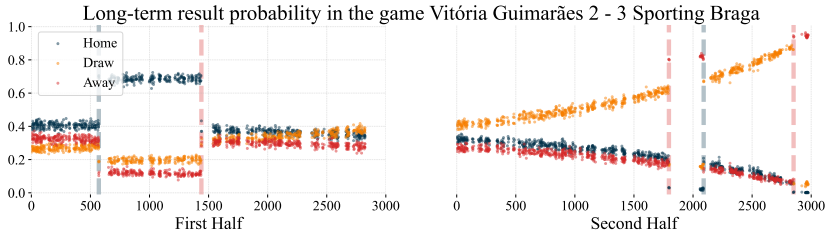

Fig. 6. Real-time long-term probabilities of home, draw, and away win outcomes for a selection of matches, generated using long-term simulations with LEMs. Vertical lines indicate the timing of goals. Vitória Guimarães - Sporting Braga, 2 - 3, May 11, 2024.

balance of performance and computational efficiency. The versatility of this approach shines through its practical applications, enabling real-time match outcome prediction, detailed player performance analysis and game simulations. By addressing previous limitations like synchronization and scalability with a single, flexible model, the unified LEM lays a robust foundation for the future of soccer analytics. Its adaptability suggests potential applications beyond soccer, extending to other sports or sequential event-driven domains. Future research on transformer-based architectures for further performance gains, and developing fine-tuning techniques to tailor the model to specific tasks or teams will increase the use cases of LEMs even more. This work marks a significant step forward in creating scalable, impactful tools for data-driven sports analysis.

Acknowledgments. This work is financed by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia, within project UIDB/50014/2020.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Anzer, G., Bauer, P.: A Goal Scoring Probability Model for Shots Based on Synchronized Positional and Event Data in Football (Soccer). Frontiers in Sports and Active Living **3**, 624475 (Mar 2021). https://doi.org/10.3389/fspor.2021.624475
2. Baron, E., Hocevar, D., Salehe, Z.: A Foundation Model for Soccer (Jul 2024). https://doi.org/10.48550/arXiv.2407.14558
3. Open AI: Language Models are Few-Shot Learners. (2020)
4. Decroos, T., Bransen, L., Van Haaren, J., Davis, J.: Actions Speak Louder than Goals: Valuing Player Actions in Soccer. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 1851–1861. ACM, Anchorage AK USA (Jul 2019). https://doi.org/10.1145/3292500.3330758
5. Fernández, J., Bornn, L., Cervone, D.: A framework for the fine-grained evaluation of the instantaneous expected value of soccer possessions. Machine Learning **110**(6), 1389–1427 (Jun 2021). https://doi.org/10.1007/s10994-021-05989-6

6. Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D.d.L., Hendricks, L.A., Welbl, J., Clark, A., Hennigan, T., Noland, E., Millican, K.: Training Compute-Optimal Large Language Models. 36th Conference on Neural Information Processing Systems (NeurIPS 2022) (2022)
7. Kaplan, J., McCandlish, S., Henighan, T., Brown, T.B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., Amodei, D.: Scaling Laws for Neural Language Models (Jan 2020). https://doi.org/10.48550/arXiv.2001.08361
8. Karun Singh: Introducing Expected Threat (xT) (2018)
9. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization (Jan 2017). https://doi.org/10.48550/arXiv.1412.6980
10. Mendes-Neves, T., Meireles, L., Mendes-Moreira, J.: Estimating Player Performance in Different Contexts Using Fine-tuned Large Events Models (Apr 2024). https://doi.org/10.48550/arXiv.2402.06815
11. Mendes-Neves, T., Meireles, L., Mendes-Moreira, J.: Towards a foundation large events model for soccer. Machine Learning (Sep 2024). https://doi.org/10.1007/s10994-024-06606-y
12. Pappalardo, L., Cintia, P., Rossi, A., Massucco, E., Ferragina, P., Pedreschi, D., Giannotti, F.: A public data set of spatio-temporal match events in soccer competitions. Scientific Data **6**(1) (2019). https://doi.org/10.1038/s41597-019-0247-7
13. Pollard, R., Ensum, J., Taylor, S.: Estimating the probability of a shot resulting in a goal: The effects of distance, angle and space. International Journal of Soccer and Science **2**(1) (2004)
14. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language Models are Unsupervised Multitask Learners. Open AI (2019)
15. Rudd, S.: A Framework for Tactical Analysis and Individual Offensive Production Assessment in Soccer Using Markov Chains. New England Symposium on Statistics in Sports (2011)
16. Simpson, I., Beal, R.J., Locke, D., Norman, T.J.: 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. ACM (Aug 2022). https://doi.org/10.1145/3534678.3539138
17. Spearman, W., Basye, A., Dick, G., Hotovy, R., Pop, P.: Physics-Based Modeling of Pass Probabilities in Soccer (2017)
18. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., Lample, G.: LLaMA: Open and Efficient Foundation Language Models (Feb 2023). https://doi.org/10.48550/arXiv.2302.13971
19. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is All you Need. ArXiV (2017)
20. Wang, Z., Veličković, P., Hennes, D., Tomašev, N., Prince, L., Kaisers, M., Bachrach, Y., Elie, R., Wenliang, L.K., Piccinini, F., Spearman, W., Graham, I., Connor, J., Yang, Y., Recasens, A., Khan, M., Beauguerlange, N., Sprechmann, P., Moreno, P., Heess, N., Bowling, M., Hassabis, D., Tuyls, K.: TacticAI: an AI assistant for football tactics. Nature Communications **15**(1), 1906 (Mar 2024). https://doi.org/10.1038/s41467-024-45965-x