TranSPORTmer: A Holistic Approach to Trajectory Understanding in Multi-Agent Sports Scenarios

Guillem Capellera[1,2], Luis Ferraz[1], Antonio Rubio[1], Antonio Agudo[2], and Francesc Moreno-Noguer[2]

[1] Kognia Sports Intelligence, Barcelona, Spain
{guillem.capellera,luis.ferraz,antonio.rubio}@kogniasports.com
[2] Institut de Robòtica i Informàtica Industrial CSIC-UPC, Barcelona, Spain
{gcapellera,aagudo,fmoreno}@iri.upc.edu

Abstract. Understanding trajectories in multi-agent scenarios requires addressing diverse tasks, including predicting future movements, imputing missing observations, inferring the status of unseen agents, and classifying different states. Traditional data-driven approaches often handle these tasks independently, utilizing specific models tailored for individual purposes. In this paper, we introduce TranSPORTmer, a unified framework capable of simultaneously addressing all these tasks at once, showcasing its application to the intricate dynamics of multi-agent soccer and basketball scenarios. TranSPORTmer builds upon a Transformer architecture and employs Set Attention Blocks sequentially to model trajectories in an equivariant manner, effectively capturing temporal dynamics and social interactions among agents. The model's task is guided by an input mask, concealing missing or yet-to-be-predicted observations. Additionally, we introduce a CLS extra agent to classify states along soccer trajectories, including *passes*, *possessions*, *out-of-play* intervals, and *uncontrolled* states, contributing to an enhancement in modeling trajectories. To validate TranSPORTmer's efficacy, we conducted a thorough evaluation on a soccer and two basketball datasets. Across various metrics, we demonstrate that our general-purpose model consistently outperforms task-specific state-of-the-art architectures in player forecasting, unified player forecasting-imputation, ball inference, and ball imputation. Supplementary video: https://youtu.be/8VtSRm8oGoE

Keywords: Multi-agent modelling · Imputation · Transformers.

1 Introduction

Multi-agent systems are prevalent in various real-world scenarios encompassing pedestrian modelling [2, 5, 25, 29, 38, 49, 50, 56, 58, 67], human pose estimation [1, 11, 24, 28, 35, 45–47], and sports analytics [3, 30, 33, 71, 72]. This paper focuses on the latter, where trajectory understanding plays a pivotal role in unraveling the inter-dependencies within multi-agent sports scenarios. This understanding opens up diverse applications such as performance evaluation [10,15,61], scouting

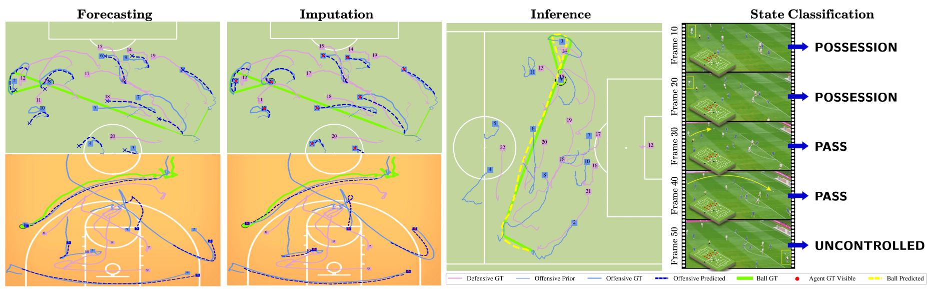

Fig. 1: TranSPORTmer is a holistic model that is able to perform multiple tasks for trajectory understanding in multi-agent sport scenarios. The images showcase examples using soccer and basketball data for the tasks of *forecasting*: predicting future trajectories given past observations; *imputation*: predicting agent trajectories given partial observations; *inference*: predicting the trajectory of an unobserved agent given the state of other ones; and *state classification*: assigning a semantic label to each frame of the sequence. Continuous and dashed lines correspond to observed states and predicted trajectories, respectively.

[52], tactical analysis [16, 65] and event detection [21, 64]. In contrast to urban contexts, the realm of sports requires the capturing of both individual player influences and comprehensive team strategies, all of which involve heightened levels of interactions and complex dynamics.

Despite the promising applications, challenges persist in this domain. Inherent inaccuracies in optical tracking data, often arising from occlusions, pose a significant hurdle. The substantial costs associated with adopting GPS technology for ball tracking [36] add an extra layer of complexity. Additionally, the intricacies introduced by off-screen players [51,68] and the nuances of broadcasting videos further contribute to the challenges in multi-agent sports scenarios. Moreover, the annotation of a match demands a significant amount of manual work due to the density of events and states that unfold during gameplay.

Previous research has proposed task-specific solutions for trajectory forecasting [13, 18, 69] and imputing missing observations [43, 51]. Some works offer unified frameworks capable of addressing both tasks [54,68]. However, a common limitation across these models is the assumption that all agents have either complete or partially observed data, overlooking scenarios involving entirely unseen agents. Furthermore, several of these models rely on recursive prediction strategies, potentially compromising efficiency in match processing and performance when modeling long-range sequences.

In the domain of unseen agent inference, recent efforts have concentrated on predicting both ball location [36] and player positions [20]. Nevertheless, these approaches require additional data beyond agent locations, including velocities and customized event data. Moreover, prior works focusing on event and state classification using trajectory data often center around a limited set of sparse scenarios [21] or specific events like passes and receptions [32, 36], without providing comprehensive annotation for every state of the game.

In this paper we introduce TranSPORTmer, a holistic and efficient approach to trajectory understanding capable of handling multiple tasks in multi-agent sports scenarios. These tasks encompass predicting future motions, imputing missing observations, inferring the status of unseen agents, and densely classifying states occurring during the game (refer to Fig. 1). Our approach is built upon transformer encoders [17, 19, 37, 62], also known as Set Attention Blocks (SABs) which operate in an equivariant manner, disregarding the specific ordering of agents. They adeptly capture both temporal dynamics and inter-agent interactions, commonly referred to as "social" interactions. In order to allow adaptability, we propose a socio-temporal mask to conceal missing or yet-to-be-predicted observations and to define semantic game tasks, such as predicting offensive players conditioned on the opponent team and the ball. That allows for a seamless transition between different problem definitions without necessitating adjustments to the network architecture. Additionally, we have devised a learnable effective mask in the loss term to model the uncertainty of visible observations near the hidden ones. Lastly, our model employs CLS tokens [17], denoted as CLS extra agent, allowing state classification for each timestep.

We train and validate our approach on a soccer dataset with player trajectories and annotated states, as well as two basketball datasets with player trajectories. A thorough validation demonstrates that our holistic approach surpasses task-specific state-of-the-art methods in forecasting player trajectories, ball trajectory inference, and imputation. Notably, in the soccer dataset TranSPORTmer achieves accurate state classification in key actions such as *passes*, *possessions*, *uncontrolled* situations, and transitions between in-play and *out-of-play* states, thereby contributing in general to improved results in trajectory modeling tasks.

Our key contributions can be summarized as follows:

- We introduce a holistic transformer-based approach for understanding trajectories in multi-agent sports scenarios, capable of conducting trajectory forecasting, imputation, inference, and simultaneous state classification using a single model.
- We validate our versatile approach on an extensive soccer dataset, demonstrating results that exceed those of state-of-the-art forecasting techniques, as well as ball imputation and inference tasks. Furthermore, we evaluate our approach on two basketball datasets, achieving results that surpass those of competing methods in player forecasting and unified forecasting-imputation tasks.
- We devise a learnable effective mask in the loss term to model uncertainty in the first and second neighbours of hidden (masked) values.
- To the best of our knowledge, no prior work has addressed the challenge of performing per-frame state classification in sports scenarios by using partial ball trajectories. We demonstrate our model's capability to achieve this with any information on the ball trajectory, by inferring it without a significant drop in performance. Additionally, we demonstrate that temporal classification, in general, enhances the trajectory prediction task.
- We elucidate how the model surpasses over 25% the state-of-the-art results in ball inference through attention maps, providing a coarse-to-fine understanding.

2 Related Work

This section discusses the related work in trajectory forecasting, imputation, inference, and state classification, with a specific emphasis on multi-agent sports scenarios.

Trajectory Forecasting consists in predicting future positions based on past observations. In the context of multi-agent sports, earlier approaches [23, 71, 72] predicted long-term behaviors using Variational Recurrent Neural Networks (VRNNs) [14]. However, these methods lack equivariance properties and rely on heuristics like tree-based role alignment [44, 59] to define a specific *ordering* of the agents. The combination of VRNNs with Graph Neural Networks (GNNs) [7], results in GVRNN [60, 69], defining an equivariant model treating agents as nodes of a fully connected graph. This approach allows the aggregation of spatial interactions for final predictions. However, GVRNN treat agent dependencies equally by aggregating agent information at each timestep. To handle dependencies between different agents more effectively, [9, 18, 22, 34, 48] used a Graph Attention Network (GAT) [63], replacing fully connected graphs. Transformer-based models [62] have been used in this task [3, 4], demonstrating a superior performance compared to graph-recurrent-based methods. Nevertheless, conducting attention in both temporal and social dimensions simultaneously still incurs a notable computational cost. In contrast, TranSPORTmer employs attention in both temporal and social dimensions sequentially. This design choice results in a substantial reduction in computational cost without compromising performance. Moreover, by departing from recursive sequence construction, our model gains a significant advantage in long-term sequence prediction, thanks to its inherent look-ahead temporal property.

Trajectory Imputation involves predicting agents' behavior in unknown frames using available information, such as partial trajectories of the target agent. Previous research tackled value imputation in time series with an autoregressive RNN [12]. A bidirectional GVRNN structure proposed by [51] addressed imputing missing agent observations in soccer games. However, due to its autoregressive nature, these approaches may lead to suboptimal results in long-range sequences [27, 39]. Liu *et al.* [43] introduced a non-autoregressive imputation model exploiting the multi-resolution structure of sequential data, although it falls short in handling trajectory forecasting. Another asynchronous approach solved imputation and forecasting tasks using imitative techniques [54]. Some research leveraged GVRNN to handle both tasks simultaneously [68]. Similarly, our method is equipped to handle this unified task effectively.

Trajectory Inference aims to predict the behavior of agents across all frames based solely on information from other agents. This is often approached as ball inference [6, 36]. The fusion of Set Transformers [40] with Bi-LSTM [31] has been utilized to infer the ball trajectory and identify the ball possessor (or pass receiver). This method relies on player trajectories and their corresponding velocities [36]. As we will demonstrate later, TranSPORTmer does not require player velocities to infer the ball position. Moreover, it can be applied to any type of agent, including the goalkeeper, that exhibits very particular motion patterns.

State/Event Classification: On this context, [64] applied a rule-based framework to identify soccer events based on agent trajectories. [21] proposed a method using a variational autoencoder and support vector machine to detect events such as corner kicks, crosses, and counterattacks. Another significant work is [32], introducing a pass receiver Transformer-LSTM model that integrates visual information with player and ball trajectories. The recent work previously mentioned for ball inference [36] can also serve as a pass receiver prediction model. However, these approaches primarily focus on limited soccer context situations such as set pieces and often rely on robust and precise estimations of ball and/or player trajectories. TranSPORTmer provides a more detailed coverage of events, referred to as states, including *passes*, *possessions*, *uncontrolled* situations, and transitions between in-play and *out-of-play* states. The model also demonstrates robustness against missing observations, showcasing its ability to perform state classification even with an unseen ball.

Pedestrian Motion Modeling: We also review works on generic pedestrian motion modeling. In this context, Transformer-based architectures [40, 62] have made significant contributions to pedestrian forecasting [70]. In [26] demonstrated promising results in the TrajNet benchmark [57] when focusing solely on processing temporal dynamics without considering social interactions. Becker *et al*. [8] introduced an RNN baseline with an MLP decoder, without social encoding, yet surpassing social pooling methods [2, 29, 41], and underscoring their limitations in capturing social interactions. Subsequent approaches [1, 25] leverage the encoder component of transformers, without positional encoding, to encode social interactions among agents. Very recently, diffusion models have entered the scene in multi-agent systems for human behavior modelling [55, 66], primarily focusing on motion generation tasks.

3 Revisiting Attention Mechanisms

Attention mechanisms are effective at capturing relationships in sequences or sets. Utilizing n queries $\mathbf{Q}$ and n_v keys $\mathbf{K}$ of dimension d_k, and n_v values $\mathbf{V}$ of dimension d_v as inputs, the attention mechanism computes weighted sums of values by assessing the compatibility between queries and keys measured using dot or scaled dot products. The masked attention expression $\mathrm{A}(\mathbf{Q}, \mathbf{K}, \mathbf{V}, \mathbf{M})$, incorporating a binary mask $\mathbf{M}$, can be written as:

$$\mathrm{softmax}\left(\frac{(\mathbf{Q}\mathbf{K}^\top) + o(\mathbf{M})}{\sqrt{d_k}}\right)\mathbf{V}, \tag{1}$$

with $\mathbf{Q} \in \mathbb{R}^{n \times d_k}$, $\mathbf{K} \in \mathbb{R}^{n_v \times d_k}$, $\mathbf{V} \in \mathbb{R}^{n_v \times d_v}$, and $\mathbf{M} \in \{0,1\}^{n \times n_v}$. $\mathbf{M}$ determines which keys are employed in computing attention for each query. More specifically, entries filled with zeros in $\mathbf{M}$ indicate keys to be included, while entries filled with ones denote those to be excluded. The function $o(\cdot)$ maps $0/1$ values to $0/-\infty$. Note that the softmax operator output will assign zero weight to the latter set of keys, ensuring that similarity scores are normalized. The weighted value sum is obtained by multiplying attention weights with their corresponding values.

In practice, attention mechanism is often extended with multiple attention heads, also called *Multi-Head Attention* (MHA) [62], allowing to capture different aspects of the data. Instead of computing a single attention function, this method projects $\mathbf{Q}$, $\mathbf{K}$, $\mathbf{V}$ onto H different d_k^h, d_k^h, d_v^h dimensional vectors, respectively. Each attention head computes its own attention weights and weighted sum of values, and the outputs of the attention heads are concatenated or linearly transformed to obtain the final attention output.

$\mathrm{MHA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}, \mathbf{M})$ is inferred using $\mathrm{concat}(\mathrm{head}_1, \ldots, \mathrm{head}_h, \ldots, \mathrm{head}_H)\mathbf{W}^O$, where $\mathrm{head}_h = \mathrm{A}(\mathbf{Q}\mathbf{W}_h^Q, \mathbf{K}\mathbf{W}_h^K, \mathbf{V}\mathbf{W}_h^V, \mathbf{M})$ and $h = \{1, \ldots, H\}$ represents the h-th attention head. Therefore, MHA is a $[n \times d]$ matrix, with learnable parameters $\{\mathbf{W}_h^Q, \mathbf{W}_h^K\} \in \mathbb{R}^{d_k \times d_k^h}$, $\mathbf{W}_h^V \in \mathbb{R}^{d_v \times d_v^h}$, and $\mathbf{W}^O \in \mathbb{R}^{h d_v^h \times d}$. In this work we will use $d_k = d_v = d$ and $d_k^h = d_v^h = d_k/H$ as it is standard in literature.

The MHA operation was extended [40] to operate on sets by defining the SAB. Given one set of d-dimensional vectors and one mask determining which vectors are used to compute the attention, denoted by $\mathbf{X}$ and $\mathbf{M}$, respectively, the SAB is defined as:

$$\mathrm{SAB}(\mathbf{X}, \mathbf{M}) = \mathrm{LayerNorm}(\mathbf{H} + \mathrm{rFFN}(\mathbf{H})), \tag{2}$$

where $\mathbf{H} = \mathrm{LayerNorm}(\mathbf{X} + \mathrm{MHA}(\mathbf{X}, \mathbf{X}, \mathbf{X}, \mathbf{M}))$ and $\mathrm{rFFN}(\mathbf{H})$ denotes the row-wise feed-forward neural network applied to $\mathbf{H}$. Note that SAB is an adaptation of the encoder block of the transformer but lacks the positional encoding. The MHA operation itself provides the property of permutation equivariance, allowing SAB to effectively capture relationships in the absence of positional information. When no mask is provided, the SAB operation is denoted as $\mathrm{SAB}(\mathbf{X}) = \mathrm{SAB}(\mathbf{X}, \mathbf{0})$, where $\mathbf{0}$ denotes an all-zero-values mask.

4 Method

4.1 Problem Statement

Let us consider a set of $N \in \mathbb{N}$ agent observations by including players and a ball in our case, denoted as $X = \{\mathbf{x}^1, \ldots, \mathbf{x}^n, \ldots, \mathbf{x}^N\}$ with $n = \{1, \ldots, N\}$, where each observation contains the (x, y) pitch locations. We can now collect T observations along time for every agent, defining the tensor $\mathbf{X}_{1:T}$ where all $\mathbf{x}_t^n$ with $t = \{1, \ldots, T\}$ are considered. Trajectory completion aims at inferring missing or unobserved entries of a data structure based on the visible ones. Given partial observations $\mathbf{X}_{1:T}^U$ and a $[T \times N]$ binary mask $\mathbf{M}$ to encode by 0 the visible observations and by 1 the unobserved ones, the goal is to find a function $f_1(\cdot)$ to infer the full observations such that:

$$f_1(\mathbf{X}_{1:T}^U, \mathbf{M}) = \mathbf{X}_{1:T}. \tag{3}$$

Based on that idea, we can define three sub-tasks by imposing specific constraints on the mask $\mathbf{M}$:

Trajectory forecasting: We assume a timestep $\hat{t} < T$, and full observability for all agents from 1 to $\hat{t}$, *i.e.*, all the entries related to these observations in $\mathbf{M}$ are set to 0.

Trajectory imputation: In this case, it is assumed that at least one observation per agent is available, *i.e.*, there is at least one null entry per row in $\mathbf{M}$.

Trajectory inference: This task is the most challenging, as it involves at least one agent having no observations throughout the entire duration, meaning that at least one entire row of the matrix $\mathbf{M}$ lacks null entries.

In addition, we delve into the classification of states within the game, seeking another function that takes the same input as in Eq. (3) but generates an output corresponding to a specific state for each timestep. These states involve the actions *pass*, *possession*, *uncontrolled*, and *out of play*, 4 in total, all of which are pertinent in a soccer context. Specifically, our objective is to estimate a classification function $f_2(\cdot)$ such that:

$$f_2(\mathbf{X}^U_{1:T}, \mathbf{M}) = \mathbf{s}_{1:T}, \tag{4}$$

where $\mathbf{s}_{1:T}$ is a $[T \times 4]$ dimensional tensor that represents the probability distribution over each game state for each timestep.

4.2 TranSPORTmer

We next present TranSPORTmer our holistic and versatile approach to address trajectory forecasting, imputation, inference and state classification. Figure 2 depicts its main components.

Input processing: The input tensor $\mathbf{X}^U_{1:T}$ contains partial observations, indicated by the input mask matrix $\mathbf{M}$. We can append additional known information to this tensor, such as the *agent type*, which is an integer representing 0 for ball, 1 for offensive team player, and 2 for defensive team player. Therefore, the dimension of $\mathbf{X}^U_{1:T}$ is $[T \times M \times 3]$ with $(x, y, \text{agent type})$. Initially, $\mathbf{X}^U_{1:T}$ is transformed by a row-wise feed-forward network (rFFN), becoming an embedding tensor of dimension d.

CLS extra agent: Then a CLS tensor of dimension $[T \times d]$ is appended as an extra agent along the social axis, resulting in a $[T \times (N+1) \times d]$ tensor $\mathbf{J}$. To ensure consistency, the mask $\bar{\mathbf{M}}$ of dimensions $[T \times (N+1)]$ extends $\mathbf{M}$, setting all entries corresponding to the CLS extra agent to one to indicate hidden observations. This extended mask is used in the initial SAB operations to make a first approximation of the hidden observations using temporal information.

Coarse-to-fine encoders: The next block comprises two encoders, applied sequentially and that operate in a coarse-to-fine manner. Formally:

$$\mathbf{J}' = \text{Encoder}_c(\mathbf{J}, \bar{\mathbf{M}}) = \text{SAB}_S\left(\text{SAB}_T\left(\text{SAB}_T(\mathbf{J} + \text{PE}, \bar{\mathbf{M}}), \bar{\mathbf{M}}\right)\right), \tag{5}$$

$$\text{Encoder}_f(\mathbf{J}') = \text{SAB}_S(\text{SAB}_T(\text{SAB}_T(\mathbf{J}' + \text{PE}))), \tag{6}$$

where PE corresponds to the original positional encoder [62] to preserve temporal ordering. SAB_T and SAB_S are temporal and social set attention blocks, respectively. SAB_T processes individual temporal dynamics through the temporal embeddings of each agent, while SAB_S addresses social interactions by encoding the embeddings of all agents at each timestep. The sequential configuration of SAB_T followed by SAB_S enables the implicit integration of information

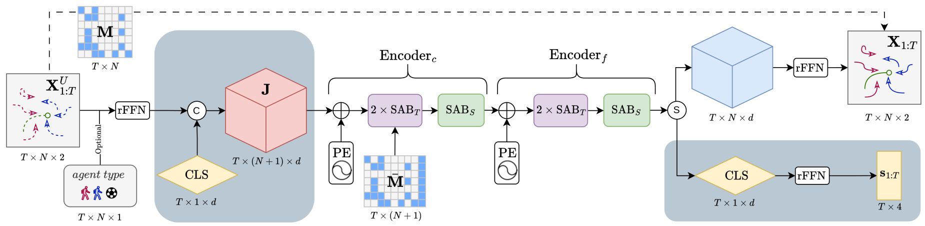

Fig. 2: TranSPORTmer. The architecture uses sequential Set Attention Blocks for attention in both temporal (SAB_T) and social (SAB_S) axes. A Positional Encoder (PE) precedes each encoder to maintain the temporal sequence. The mask $\mathbf{M}$ identifies the values to be predicted (dashed arrow), forming the complete observation tensor $\mathbf{X}_{1:T}$. The extended mask $\bar{\mathbf{M}}$ is applied to the $2 \times \mathrm{SAB}_T$ of the first $\mathrm{Encoder}_c$, conveying information about hidden and visible states. Gray segments are involved in state classification, including the CLS extra agent and the final classification head to rank the state classes per frame. (c) operation stands for concatenation and (s) for split.

from both future and past time steps through temporal attention, enhancing the model's ability to consider a broader temporal context in social attention.

Output construction: After passing through the encoder blocks, the output tensor retains the dimensions of $\mathbf{J}$. This output is then split into two tensors: the encoded trajectory embeddings and the encoded CLS extra agent. The former undergoes a rFFN operation to yield a tensor of dimension $[T \times N \times 2]$ corresponding to the predicted (x, y) pitch locations. The binary mask $\mathbf{M}$ is then employed to directly propagate the visible (x, y) values from the input tensor, $\mathbf{X}^U_{1:T}$, resulting in the full observation tensor $\mathbf{X}_{1:T}$. On the other side, the encoded CLS extra agent is reshaped and processed through another rFFN operation to obtain probability scores for each class at each time, $s_{1:T}$, of dimension $[T \times 4]$.

Note that our model exhibits permutation equivariance under agent permutations, as the operations along the social axis inherently maintain this property. In this architecture, an additional mask can be employed in all SAB blocks to ignore corrupt or NaN inputs observations not to predict, or to facilitate padding during batching with length-varying sequences and different numbers of agents involved. We denote this mask as NaN-mask.

4.3 Loss Functions

We introduce a learnable effective mask $\mathbf{M_{eff}}$, with the same dimension as $\mathbf{M}$ to represent observation uncertainty. Here, ${\mathbf{M_{eff}}}^n_t = 1$ where $\mathbf{M}^n_t = 1$, indicating areas of maximum uncertainty (hidden observations). Along the time axis, we use two learnable weights: $w_1 \in (0, 1)$ bounded using a sigmoid function, and $w_2 := 1 - w_1$. These weights are applied to the immediate neighbors of 1's (${\mathbf{M_{eff}}}^n_t = w_1$) and to the second neighbors if they are not immediate neighbors (${\mathbf{M_{eff}}}^n_t = w_2$). All other values are set to null entries, signifying visible observations and, consequently, a lack of uncertainty. Extending the mask in the loss for boundary observations to reflect uncertainty enables the model to reconstruct them, leading to more accurate overall predictions. Figure 3 illustrates the dif-

Fig. 3: Binary mask ($\mathbf{M}$) and learnable effective mask ($\mathbf{M_{eff}}$) for a single agent. Null values indicate visible observations.

ferences between the binary mask ($\mathbf{M}$) and the learnable effective mask ($\mathbf{M_{unc}}$) for a single agent over time.

The loss function for trajectory completion uses the Average Displacement Error (ADE) with the learnable effective mask, assessing the disparity between the predictions and the ground truth:

$$\mathcal{L}_{\text{ADE}} = (\sum_{n=1}^{N}\sum_{t=1}^{T} {\mathbf{M_{eff}}}_t^n)^{-1} \sum_{n=1}^{N}\sum_{t=1}^{T} \|\hat{\mathbf{x}}_t^n - \mathbf{x}_t^n\|_2 \, {\mathbf{M_{eff}}}_t^n \,, \tag{7}$$

where $\hat{\mathbf{x}}_t^n$ denotes our estimation of the n-th agent at time t, $\mathbf{x}_t^n$ corresponds to the ground truth. We also utilize a standard Cross Entropy (CE) loss as the training metric for the state classification task:

$$\mathcal{L}_{\text{CE}} = -\frac{1}{T}\sum_{t=1}^{T}\sum_{c=1}^{4} s_t^c \log(\hat{s}_t^c), \tag{8}$$

where s_t^c represents the ground truth probability of game being in state c at time t, and $\hat{s}_t^c$ is the predicted probability. The overall loss is $\mathcal{L} = \mathcal{L}_{\text{ADE}} + \lambda\mathcal{L}_{\text{CE}}$, where λ is a weighting factor set to $\lambda = 4$ when classifying states. In the supplementary (*suppl*), we detail the training procedure and the chosen hyperparameters.

We will study the importance of each part of the method by considering: *Ours w/o CLS* (without utilizing state classification), *Ours w/o* $\mathbf{M_{eff}}$ (using the binary mask $\mathbf{M}$ instead of $\mathbf{M_{eff}}$ in the loss term), and *Ours w/o SOC* (without employing SAB_S nor state classification). Additionally, we depict combinations of these variations.

5 Experimental Evaluation

We next present experimental results on trajectory completion and state classification, comparing our approach with competing methods. For quantitative evaluation, we utilize the ADE metric in Eq. (7) but considering the binary mask $\mathbf{M}$ instead of $\mathbf{M_{eff}}$. For trajectory forecasting, we use the Final Displacement Error (FDE) to measure the final prediction deviation. We also consider the Maximum Error (MaxErr) to capture the largest discrepancies:

$$\text{MaxErr} = \frac{1}{D}\sum_{n=1}^{N} \max_{t\in\{1,\ldots,T\}} \left(\|\mathbf{x}_t^n - \hat{\mathbf{x}}_t^n\|_2 \cdot \mathbf{M}_t^n\right),$$

where $D = \sum_{n=1}^{N} \mathbb{1}\left(\sum_{t=1}^{T} \mathbf{M}_t^n\right)$, with $\mathbb{1}(\cdot)$ as the unit step function. For state classification, being $\mathbb{I}(\cdot)$ the indicator function, Accuracy (Acc) is computed as:

$$\text{Acc} = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}\left[\underset{c}{\text{argmax}}(s_t^c) = \underset{c}{\text{argmax}}(\hat{s}_t^c)\right].$$

5.1 Datasets

Soccer: This dataset comprises real soccer match data from LaLiga's 2022-2023 season, including 283 matches. The matches are split into sequences of $T = 60$ frames, representing 9.6 seconds sampled at 6.25 Hz. Each frame contains 23 observations (x, y) for each one of the agents (22 players and the ball). The *agent type* is known in this dataset. Goalkeepers may contain NaNs if they are not visible. To ensure consistency with prior research, the agent order is standardized. The dataset is split into 82,954 / 7,500 / 6,258 sequences for training, validation and testing, respectively, with each split using different matches. For the state classification task, the dataset is complemented with one state label per frame, considering the states *pass*, *possession*, *uncontrolled* and *out-of-play*.
Basketball-VU: This dataset consists of basketball player tracking data provided by STATS SportVU from the 2016 NBA season. To evaluate our model on player forecasting, we use the same splits as in [48]. Each sequence consists of 50 timesteps representing 10 seconds sampled at 5Hz, where each one contains the (x, y) observations for 10 players and the ball.
Basketball-TIP: We also consider another basketball dataset from the 2012 NBA season. Following the same splits as [71], Xu *et al.* [68] pre-processed it allowing to evaluate in both player imputation and forecasting tasks, renaming it as Basketball-TIP. This dataset employs two distinct strategies to simulate the appearance and disappearance of players: the "circle mode" and the "camera mode". Each sequence consists on 50 frames representing 8 seconds sampled at 6.25Hz, each one containing the (x, y) observations for 10 players and the ball.
ETH-UCY: For completeness, we conducted an experiment using the ETH-UCY pedestrian dataset [42, 53]. Our approach performs comparably to deterministic SOTA architectures [56, 67], achieving a 4.3% improvement in ADE on the ETH subset. Detailed information and results are available in the *suppl.*

5.2 Player Forecasting and Imputation

First, we assess our model's effectiveness in **(i) soccer player forecasting and imputation**. The predicted players, referred to as agents of interest P, are predicted using all future visible observations of conditioning agents, like the ball and/or an opponent team. In the forecasting task, the model observes 20 timesteps (3.2s) and predicts the next 40 timesteps (6.4s) of P. The imputation task is similar but with the final location of each player of interest set as visible. As in previous studies [69], goalkeepers are excluded from this analysis.

In the forecasting task, we compare against the following implemented baselines: *Velocity* extrapolation, projecting agent predictions linearly based on observed velocity; *RNN* encoder with LSTM, using shared weights, and MLP decoder for prediction [8]; *GRNN* as the non-variational version of GVRNN [69]; *GRNN+Att* which is the previous baseline but using GAT instead of GNNs; *Transformer* which mirrors our pipeline but uses SAB to perform attention across all timesteps of all agents simultaneously [3], as opposed to decoupling attention in SAB_T and SAB_S. Further details of these implementations can be

Predict P (Condition)		Forecasting								Imputation	
		Velocity	RNN	Ours w/o SOC	GRNN	GRNN+Att	Transformer	Ours w/o CLS	Ours	Ours w/o CLS	Ours
	Social				✓	✓	✓	✓	✓	✓	✓
Players (Ball)	$\text{ADE}_P \downarrow$	5.96	4.36	4.08	4.02	3.67	2.66	2.53	**2.42**	**1.14**	1.15
	$\text{MaxErr}_P \downarrow$	13.49	8.95	8.60	7.43	7.02	5.28	5.12	**4.97**	**2.21**	2.22
	$\text{FDE}_P \downarrow$	13.33	8.59	8.25	6.85	6.49	4.78	4.65	**4.50**	-	-
	Acc (%) $\uparrow$	-	-	-	-	-	-	-	87.35	-	89.00
Offense (Defense+Ball)	$\text{ADE}_P \downarrow$	5.76	4.23	3.96	3.76	3.30	2.26	2.10	**2.06**	**0.99**	1.02
	$\text{MaxErr}_P \downarrow$	13.04	8.72	8.39	6.84	6.31	4.45	4.27	**4.21**	**1.92**	1.97
	$\text{FDE}_P \downarrow$	12.89	8.39	8.07	6.32	5.80	3.96	3.82	**3.77**	-	-
	Acc (%) $\uparrow$	-	-	-	-	-	-	-	88.91	-	89.69
Defense (Offense+Ball)	$\text{ADE}_P \downarrow$	6.16	4.49	4.20	3.47	3.22	2.14	2.01	**1.98**	**1.03**	1.04
	$\text{MaxErr}_P \downarrow$	13.94	9.18	8.81	6.29	5.98	4.17	4.04	**3.98**	**1.99**	2.00
	$\text{FDE}_P \downarrow$	13.78	8.79	8.44	5.69	5.36	3.63	3.55	**3.49**	-	-
	Acc (%) $\uparrow$	-	-	-	-	-	-	-	89.92	-	90.47

Table 1: Evaluation in (i) soccer player forecasting and imputation. Predictions are generated with a time horizon of 6.4s using a prior of 3.2s. P denotes agents of interest. For the imputation task, the last observation of each agent is visible. All metrics, except Acc, are in meters.

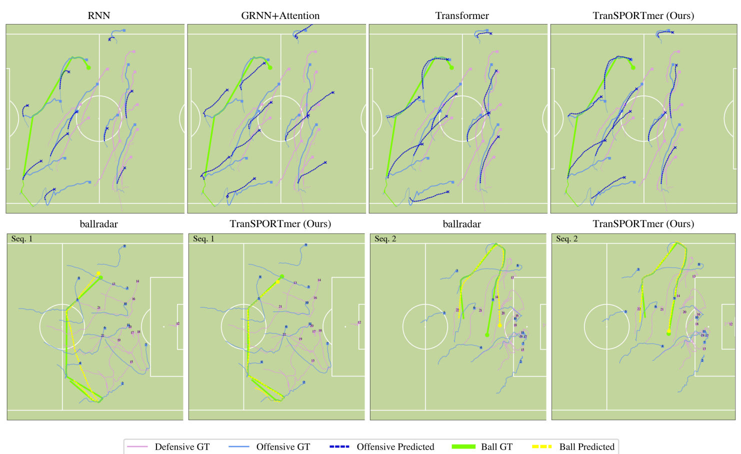

Fig. 4: Qualitative evaluation in soccer player forecasting and ball inference. **Top:** Offensive player trajectory forecasting with a time horizon of 6.4s using a prior of 3.2s. **Bottom:** Ball inference through the full 9.6s sequence.

found in the *suppl.* It is important to note that *Velocity*, *RNN*, and *Ours w/o SOC* operate independently for each agent, making the agents ordering irrelevant and preventing them from utilizing any social conditioning.

Table 1 shows the results of (i) soccer player forecasting and imputation with conditioning agents indicated in parentheses. As expected, socially aware architectures exhibit superior performance in all metrics, particularly when the number of conditioning agents is increased. Results for *Ours w/o SOC* underscore the clear significance of SAB_S. *Transformer* achieves slightly inferior results compared to *Ours w/o CLS*, likely due to its flattened attention mechanism, which may cause confusion with the higher number of non-correlated observations. Additionally, *Transformer* is approximately four times slower at inference time (340 vs. 88 milliseconds). Furthermore, TranSPORTmer (*Ours*) outperforms *Ours w/o CLS* in forecasting, but in the imputation task, *Ours w/o CLS* achieves slightly better results, possibly due to sub-optimal λ compared to forecasting. Figure 4-top shows an example on the task of forecasting offensive players (second task-row in Table 1). The *RNN* baseline tends to gener-

Predict P	STGAT [34]	Social-Ways [5]	GVRNN [69]	GMAT [71]	AC-VRNN [9]	DAG-Net [48]	U-MAT [22]	S-PatteRNN [49]	Ours w/o CLS
	ICCV'19	CVPRW'19	CVPR'19	ICLR'19	CVIU'21	ICPR'21	NeurIPS'22	IROS'22	
Players	-	-	-	-	-	8.55/12.37	-	8.13/12.34	**7.75**/**11.65**
Offense	9.94/15.80	9.91/15.19	9.73/15.89	9.47/16.98	9.32/14.91	**8.98**/14.08	9.01/**13.28**	-	9.19/14.24
Defense	7.26/11.28	7.31/10.21	7.29/10.62	7.05/10.56	7.01/10.16	6.87/9.76	6.88/**9.04**	-	**6.31**/**9.04**

Table 2: Evaluation in (ii) basketball player forecasting using Basketball-VU dataset ($\mathbf{ADE}_P$/$\mathbf{FDE}_P$). Predictions have a time horizon of 8s using a prior of 2s. Results are extracted from the original works, and no agent future condition is considered in this task. P denotes agents of interest. All metrics are in feet.

Model		$r = 3$ft		$r = 5$ft		$r = 7$ft		$\theta = 10^{\circ}$		$\theta = 20^{\circ}$		$\theta = 30^{\circ}$	
		I-ADE	**P**-ADE	**I**-ADE	**P**-ADE	**I**-ADE	**P**-ADE	**I**-ADE	**P**-ADE	**I**-ADE	**P**-ADE	**I**-ADE	**P**-ADE
Mean		9.07 (10.36)	-	9.53 (9.44)	-	9.51 (9.21)	-	8.83 (8.56)	-	8.64 (8.73)	-	8.47 (8.92)	-
Median		9.32 (10.55)	-	9.82 (9.64)	-	9.81 (9.44)	-	9.16 (8.84)	-	8.96 (9.02)	-	8.75 (9.21)	-
GMAT [71]	ICLR'19	7.36	-	6.89	-	6.73	-	6.42	-	5.99	-	6.01	-
NAOMI [43]	NeurIPS'19	7.68	-	7.08	-	7.04	-	6.33	-	6.11	-	5.91	-
LSTM [31]	NeurComp	7.33	20.07	6.73	14.91	6.51	10.07	6.28	9.34	6.01	7.52	5.67	6.10
VRNN [14]	NeurIPS'15	7.43	12.26	6.90	11.38	6.68	10.07	6.38	8.49	6.09	7.47	5.92	7.36
INAM [54]	CVPR'20	7.35	8.93	6.93	8.24	6.80	7.68	6.50	7.32	6.13	7.10	5.92	6.96
GC-VRNN [68]	CVPR'23	7.03	8.93	6.93	8.24	6.80	7.68	5.86	6.29	5.56	4.74	5.39	4.28
Our w/o CLS/$\mathbf{M_{eff}}$		(5.32)	(5.91)	(4.71)	(5.56)	(4.16)	(4.91)	(3.60)	(**4.77**)	(3.29)	(4.13)	(**3.08**)	(**3.60**)
Our w/o CLS		(**5.24**)	(**5.89**)	(**4.48**)	(**5.29**)	(**4.14**)	(**4.90**)	(**3.59**)	(4.78)	(**3.26**)	(**4.09**)	(**3.08**)	(**3.60**)

Table 3: Evaluation in (iii) basketball player unified imputation and forecasting using the Basketball-TIP dataset [68]. The imputation task is performed over 6.4 seconds, and forecasting over 1.6 seconds. All metrics are in feet. Our implementation results are presented in parentheses.

ate shorter predicted trajectories, emphasizing the need for social interactions to enhance performance. The *GRNN+Att* baseline exhibits improved performance with conditioning in long-term predictions. However, TranSPORTmer outperforms these baselines, yielding more realistic results aligned with ground truth positions (see video of our results in the *suppl*).

Table 1 also reports the accuracy of state classification while addressing trajectory prediction and imputation tasks. Achieving approximately 90%, these results demonstrate the robust and consistent classification power of our model, primarily attributed to the ball's visibility in all tasks. The confusion matrix in Fig. 5-left-left specifically illustrates the state classification while forecasting offensive players, achieving an overall accuracy of 88.91%. It is worth noting that the *uncontrolled* class exhibits less accurate predictions due to its challenging subjective nature in annotations and an imbalance compared to the other classes.

Next, we evaluate the effectiveness of our model in **(ii) basketball player forecasting** using the Basketball-VU dataset. The task at hand consists of observing 10 time-steps (2s) and predicting the following 40 (8s) of players without future conditioning agents (refer to *suppl* for additional conditioning-based experiments). We compare against the state-of-the-art results already published in previous works, as shown in Table 2. Our model is trained to predict both offensive and defensive players simultaneously. Other baselines, like DAG-Net [48], need separate training to achieve better results. The ADE_P and FDE_P metrics depicted in the table demonstrate that our method outperforms in predicting trajectories for all players and defense, using only one model trained with the same weights. In both Soccer and Basketball-VU datasets, it can be seen that in general, forecasting offensive players is more challenging than defensive ones.

Additionally, we assess our model's capability in **(iii) basketball player unified imputation and forecasting** tasks using the Basketball-TIP dataset.

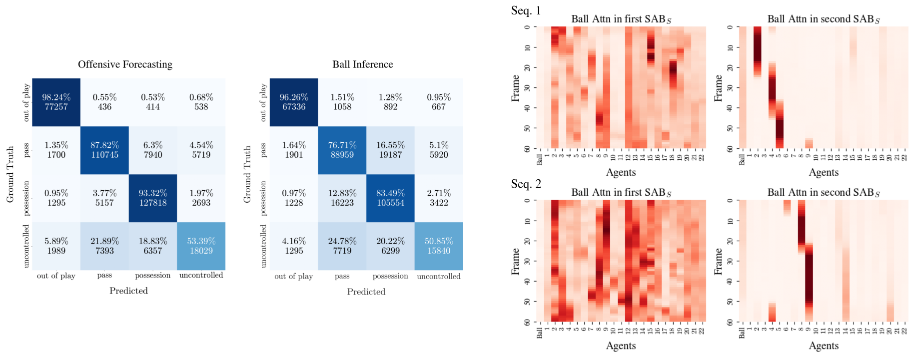

Fig. 5: Left: Confusion matrix in state classification. Offensive player trajectory forecasting (left) and ball inference (right). **Right: Attention maps for the ball.** Visualization of attention maps in each social SAB$_S$ across agents and time for the sequences #1 and #2 in Fig. 4-bottom (see the animations in *suppl* video).

This task involves observing the initial 40 timesteps (6.4s), imputing agents outside the circle/camera view, and forecasting their locations during the subsequent 10 frames (1.6s). In "circle mode", three radii $r \in \{3, 5, 7\}$ ft are considered, centered on the ball location. In "camera mode", a fixed field of view (FOV) tracks the ball from the center of the pitch, with three possible angles $\theta \in \{10, 20, 30\}^\circ$. Following Xu *et al.* [68], we predict players who have at least one observation in the initial 40 timesteps, potentially varying numbers of agents across sequences. Our method incorporates the additional NaN-mask to exclude non-interest agents within each sequence. We add our results in Table 3, showing a clear effectiveness of our method against the SOTA approaches in all six scenarios. **I-**ADE denotes the error in the initial 40 timesteps (imputation error) and **P-**ADE signifies the error in the final 10 timesteps (forecasting error). Our method performs notably well in imputation tasks compared to GC-VRNN [68], due to its unidirectional recurrent nature, which affects forecasting reliability based on imputed data. Refer to the *suppl* for detailed information and figures.

5.3 Ball Imputation and Inference

We evaluate **(iv) soccer ball imputation and inference** tasks. The inference task involves predicting all observations of the ball, masking 100% of them. The imputation task involves predicting a lower percentage of the ball observations while setting the others as visible. Players' observations serve as conditioning agents in all tasks. We benchmark against the state-of-the-art method *ballradar* [36], which employs a hierarchical approach involving possessor classification followed by ball trajectory regression. Additionally, we compare against its non-hierarchical version, *ballradar w/o POS*, which performs ball regression without possessor classification. Due to the requirements of *ballradar*, our dataset is augmented with ground-truth possessor information, player's velocities and goalkeeper locations using our method (further details can be found in the *suppl*).

Table 4 presents a comparative analysis of ball trajectory imputation and inference. Our methods consistently outperform the state-of-the-art, achieving

	ballradar w/o POS	ballradar (KDD'23)			Ours w/o CLS/$\mathbf{M_{eff}}$			Ours w/o CLS			Ours w/o $\mathbf{M_{eff}}$			Ours		
Mask	100%	80%	90%	100%	80%	90%	100%	80%	90%	100%	80%	90%	100%	80%	90%	100%
ADE ↓	5.43	0.97	1.51	3.89	0.88	1.18	2.89	0.88	1.12	2.89	**0.80**	1.23	**2.71**	0.84	**1.09**	**2.71**
MaxErr ↓	10.98	3.73	5.16	8.79	3.47	4.59	7.78	3.44	4.48	7.78	3.25	4.48	**7.39**	**3.24**	**4.39**	**7.39**
Acc (%) ↑	-	-	-	-	-	-	-	-	-	-	85.51	83.38	**80.84**	**85.59**	**83.55**	**80.84**

Table 4: Evaluation in (iv) soccer ball imputation and inference. Predictions are generated through the full 9.6s sequence. All metrics, except Acc, are in meters.

over a 25% improvement in ADE for trajectory inference. Qualitative differences in two test samples are illustrated in Fig. 4-bottom. For imputation, we showcase results by masking 80% and 90% of total ball observations for each sequence, highlighting the superior performance of our method. Notably, employing state classification in TranSPORTmer helps achieve generally better results, showcasing its holistic nature. However, *Ours w/o CLS* surpasses *ballradar* without requiring additional data beyond player (x, y) locations.

In terms of state classification accuracy (see Table 4), there is an anticipated decline compared to the soccer player trajectory forecasting and imputation section (see Table 1), likely attributed to the non-visibility of the ball, our target. Surprisingly, the method still achieves an accuracy of 80.84% in state classification showcasing that the game states can be inferred using only the movement of players (refer to Fig. 5-left-right for the detailed confusion matrix). Figure 5-right shows the attention maps generated by the SAB_S for the ball across all agents and timesteps in the two examples of Fig. 4-bottom. Computed by averaging contributions from each head, these maps reveal the model's awareness. In the first SAB_S, a broad, general awareness of other agents is observed, resembling a coarse social perspective. The second SAB_S focuses attention on the possessor player or the anticipated recipient of the ball in the event of a pass. This highlights the coarse-to-fine nature of the two encoders in our model. Refer to the *suppl* for additional ablation study regarding the coarsening-to-fine.

In both Tables 3 and 4, we include an ablation study regarding the usage of $\mathbf{M_{eff}}$ in the loss term, which generally leads to improved results. For the first neighbors, the recorded values are $w_1 \in [0.7, 0.85]$, and for the second ones $w_2 \in [0.15, 0.3]$, reflecting the expected level of uncertainty.

6 Conclusions

In this paper, we introduced TranSPORTmer, a holistic approach capable of handling multiple tasks (forecasting, imputation, inference, and state classification) for trajectory understanding in multi-agent sports scenarios. Unlike existing state-of-the-art methods, TranSPORTmer can address all tasks using a single model, eliminating the need for task-specific models. Our extensive evaluation on a large soccer dataset and two basketball datasets demonstrates competitive performance across all tasks. Notably, our approach excels in player forecasting, player imputation, ball imputation and inference tasks, while combined with state classification tasks allows to improve the results. Additionally, the learnable effective mask is utilized to model the uncertainty of neighboring hidden values, contributing to result enhancement. We believe this has the potential to pave the way for a deeper understanding of the semantic aspects of sports games.

Acknowledgment. This work has been supported by the project GRAVATAR PID2023-151184OB-I00 funded by MCIU/AEI/10.13039/501100011033 and by ERDF, UE and by the Government of Catalonia under 2023 DI 00058.

References

1. Aksan, E., Kaufmann, M., Cao, P., Hilliges, O.: A spatio-temporal transformer for 3d human motion prediction. In: 2021 International Conference on 3D Vision (3DV). pp. 565–574. IEEE (2021) 1, 5
2. Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., Savarese, S.: Social lstm: Human trajectory prediction in crowded spaces. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 961–971 (2016) 1, 5
3. Alcorn, M.A., Nguyen, A.: baller2vec++: A look-ahead multi-entity transformer for modeling coordinated agents. arXiv preprint arXiv:2104.11980 (2021) 1, 4, 10
4. Alcorn, M.A., Nguyen, A.: baller2vec: A multi-entity transformer for multi-agent spatiotemporal modeling. arXiv preprint arXiv:2102.03291 (2021) 4
5. Amirian, J., Hayet, J.B., Pettré, J.: Social ways: Learning multi-modal distributions of pedestrian trajectories with gans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. pp. 0–0 (2019) 1, 12
6. Amirli, A., Alemdar, H.: Prediction of the ball location on the 2d plane in football using optical tracking data. Academic Platform Journal of Engineering and Smart Systems **10**(1), 1–8 (2022) 4
7. Battaglia, P.W., Hamrick, J.B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., et al.: Relational inductive biases, deep learning, and graph networks. arXiv preprint arXiv:1806.01261 (2018) 4
8. Becker, S., Hug, R., Hubner, W., Arens, M.: Red: A simple but effective baseline predictor for the trajnet benchmark. In: Proceedings of the European Conference on Computer Vision (ECCV) Workshops. pp. 0–0 (2018) 5, 10
9. Bertugli, A., Calderara, S., Coscia, P., Ballan, L., Cucchiara, R.: Ac-vrnn: Attentive conditional-vrnn for multi-future trajectory prediction. Computer Vision and Image Understanding **210**, 103245 (2021) 4, 12
10. Brito Souza, D., López-Del Campo, R., Blanco-Pita, H., Resta, R., Del Coso, J.: Association of match running performance with and without ball possession to football performance. International Journal of Performance Analysis in Sport **20**(3), 483–494 (2020) 1
11. Cai, Y., Huang, L., Wang, Y., Cham, T.J., Cai, J., Yuan, J., Liu, J., Yang, X., Zhu, Y., Shen, X., et al.: Learning progressive joint propagation for human motion prediction. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16. pp. 226–242. Springer (2020) 1
12. Cao, W., Wang, D., Li, J., Zhou, H., Li, L., Li, Y.: Brits: Bidirectional recurrent imputation for time series. Advances in neural information processing systems **31** (2018) 4
13. Capellera, G., Ferraz, L., Rubio, A., Agudo, A., Moreno-Noguer, F.: Footbots: A transformer-based architecture for motion prediction in soccer. arXiv preprint arXiv:2406.19852 (2024) 2

14. Chung, J., Kastner, K., Dinh, L., Goel, K., Courville, A.C., Bengio, Y.: A recurrent latent variable model for sequential data. Advances in neural information processing systems **28** (2015) 4, 12
15. 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 (2019) 1
16. Decroos, T., Van Haaren, J., Davis, J.: Automatic discovery of tactics in spatio-temporal soccer match data. In: Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining. pp. 223–232 (2018) 2
17. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018) 3
18. Ding, D., Huang, H.H.: A graph attention based approach for trajectory prediction in multi-agent sports games. arXiv preprint arXiv:2012.10531 (2020) 2, 4
19. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020) 3
20. Everett, G., Beal, R.J., Matthews, T., Early, J., Norman, T.J., Ramchurn, S.D.: Inferring player location in sports matches: Multi-agent spatial imputation from limited observations. arXiv preprint arXiv:2302.06569 (2023) 2
21. Fassmeyer, D., Anzer, G., Bauer, P., Brefeld, U.: Toward automatically labeling situations in soccer. Frontiers in Sports and Active Living **3**, 725431 (2021) 2, 5
22. Fassmeyer, D., Fassmeyer, P., Brefeld, U.: Semi-supervised generative models for multiagent trajectories. Advances in Neural Information Processing Systems **35**, 37267–37281 (2022) 4, 12
23. Felsen, P., Lucey, P., Ganguly, S.: Where will they go? predicting fine-grained adversarial multi-agent motion using conditional variational autoencoders. In: Proceedings of the European conference on computer vision (ECCV). pp. 732–747 (2018) 4
24. Fragkiadaki, K., Levine, S., Felsen, P., Malik, J.: Recurrent network models for human dynamics. In: Proceedings of the IEEE international conference on computer vision. pp. 4346–4354 (2015) 1
25. Girgis, R., Golemo, F., Codevilla, F., Weiss, M., D'Souza, J.A., Kahou, S.E., Heide, F., Pal, C.: Latent variable sequential set transformers for joint multi-agent motion prediction. arXiv preprint arXiv:2104.00563 (2021) 1, 5
26. Giuliari, F., Hasan, I., Cristani, M., Galasso, F.: Transformer networks for trajectory forecasting. In: 2020 25th international conference on pattern recognition (ICPR). pp. 10335–10342. IEEE (2021) 5
27. Gu, J., Bradbury, J., Xiong, C., Li, V.O., Socher, R.: Non-autoregressive neural machine translation. arXiv preprint arXiv:1711.02281 (2017) 4
28. Guo, W., Du, Y., Shen, X., Lepetit, V., Alameda-Pineda, X., Moreno-Noguer, F.: Back to mlp: A simple baseline for human motion prediction. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 4809–4819 (2023) 1
29. Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., Alahi, A.: Social gan: Socially acceptable trajectories with generative adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2255–2264 (2018) 1, 5

30. Hauri, S., Djuric, N., Radosavljevic, V., Vucetic, S.: Multi-modal trajectory prediction of nba players. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1640–1649 (2021) 1
31. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997) 4, 12
32. Honda, Y., Kawakami, R., Yoshihashi, R., Kato, K., Naemura, T.: Pass receiver prediction in soccer using video and players' trajectories. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3503–3512 (2022) 2, 5
33. Hu, B., Cham, T.J.: Entry-flipped transformer for inference and prediction of participant behavior. In: European Conference on Computer Vision. pp. 439–456. Springer (2022) 1
34. Huang, Y., Bi, H., Li, Z., Mao, T., Wang, Z.: Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6272–6281 (2019) 4, 12
35. Jain, A., Zamir, A.R., Savarese, S., Saxena, A.: Structural-rnn: Deep learning on spatio-temporal graphs. In: Proceedings of the ieee conference on computer vision and pattern recognition. pp. 5308–5317 (2016) 1
36. Kim, H., Choi, H.J., Kim, C.J., Yoon, J., Ko, S.K.: Ball trajectory inference from multi-agent sports contexts using set transformer and hierarchical bi-lstm. arXiv preprint arXiv:2306.08206 (2023) 2, 4, 5, 13
37. Kong, Q., Xu, Y., Wang, W., Plumbley, M.D.: Sound event detection of weakly labelled data with cnn-transformer and automatic threshold optimization. IEEE/ACM Transactions on Audio, Speech, and Language Processing **28**, 2450–2460 (2020) 3
38. Kosaraju, V., Sadeghian, A., Martín-Martín, R., Reid, I., Rezatofighi, H., Savarese, S.: Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. Advances in Neural Information Processing Systems **32** (2019) 1
39. Lee, J., Mansimov, E., Cho, K.: Deterministic non-autoregressive neural sequence modeling by iterative refinement. arXiv preprint arXiv:1802.06901 (2018) 4
40. Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., Teh, Y.W.: Set transformer: A framework for attention-based permutation-invariant neural networks. In: International conference on machine learning. pp. 3744–3753. PMLR (2019) 4, 5, 6
41. Lee, N., Choi, W., Vernaza, P., Choy, C.B., Torr, P.H., Chandraker, M.: Desire: Distant future prediction in dynamic scenes with interacting agents. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 336–345 (2017) 5
42. Lerner, A., Chrysanthou, Y., Lischinski, D.: Crowds by example. In: Computer graphics forum. vol. 26, pp. 655–664. Wiley Online Library (2007) 10
43. Liu, Y., Yu, R., Zheng, S., Zhan, E., Yue, Y.: Naomi: Non-autoregressive multiresolution sequence imputation. Advances in neural information processing systems **32** (2019) 2, 4, 12
44. Lucey, P., Bialkowski, A., Carr, P., Morgan, S., Matthews, I., Sheikh, Y.: Representing and discovering adversarial team behaviors using player roles. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2706–2713 (2013) 4
45. Mao, W., Liu, M., Salzmann, M.: History repeats itself: Human motion prediction via motion attention. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16. pp. 474–489. Springer (2020) 1

46. Mao, W., Liu, M., Salzmann, M., Li, H.: Learning trajectory dependencies for human motion prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9489–9497 (2019) 1
47. Martinez, J., Black, M.J., Romero, J.: On human motion prediction using recurrent neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2891–2900 (2017) 1
48. Monti, A., Bertugli, A., Calderara, S., Cucchiara, R.: Dag-net: Double attentive graph neural network for trajectory forecasting. In: 2020 25th International Conference on Pattern Recognition (ICPR). pp. 2551–2558. IEEE (2021) 4, 10, 12
49. Navarro, I., Oh, J.: Social-patternn: Socially-aware trajectory prediction guided by motion patterns. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 9859–9864. IEEE (2022) 1, 12
50. Ngiam, J., Caine, B., Vasudevan, V., Zhang, Z., Chiang, H.T.L., Ling, J., Roelofs, R., Bewley, A., Liu, C., Venugopal, A., et al.: Scene transformer: A unified architecture for predicting multiple agent trajectories. arXiv preprint arXiv:2106.08417 (2021) 1
51. Omidshafiei, S., Hennes, D., Garnelo, M., Wang, Z., Recasens, A., Tarassov, E., Yang, Y., Elie, R., Connor, J.T., Muller, P., et al.: Multiagent off-screen behavior prediction in football. Scientific reports **12**(1), 8638 (2022) 2, 4
52. Pappalardo, L., Cintia, P., Ferragina, P., Massucco, E., Pedreschi, D., Giannotti, F.: Playerank: data-driven performance evaluation and player ranking in soccer via a machine learning approach. ACM Transactions on Intelligent Systems and Technology (TIST) **10**(5), 1–27 (2019) 2
53. Pellegrini, S., Ess, A., Schindler, K., Van Gool, L.: You'll never walk alone: Modeling social behavior for multi-target tracking. In: 2009 IEEE 12th international conference on computer vision. pp. 261–268. IEEE (2009) 10
54. Qi, M., Qin, J., Wu, Y., Yang, Y.: Imitative non-autoregressive modeling for trajectory forecasting and imputation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12736–12745 (2020) 2, 4, 12
55. Rempe, D., Luo, Z., Bin Peng, X., Yuan, Y., Kitani, K., Kreis, K., Fidler, S., Litany, O.: Trace and pace: Controllable pedestrian animation via guided trajectory diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13756–13766 (2023) 5
56. Saadatnejad, S., Gao, Y., Messaoud, K., Alahi, A.: Social-transmotion: Promptable human trajectory prediction. arXiv preprint arXiv:2312.16168 (2023) 1, 10
57. Sadeghian, A., Kosaraju, V., Gupta, A., Savarese, S., Alahi, A.: Trajnet: Towards a benchmark for human trajectory prediction. arXiv preprint (2018) 5
58. Salzmann, T., Ivanovic, B., Chakravarty, P., Pavone, M.: Trajectron++: Multi-agent generative trajectory forecasting with heterogeneous data for control. arXiv preprint arXiv:2001.03093 **2** (2020) 1
59. Sha, L., Lucey, P., Zheng, S., Kim, T., Yue, Y., Sridharan, S.: Fine-grained retrieval of sports plays using tree-based alignment of trajectories. arXiv preprint arXiv:1710.02255 (2017) 4
60. Sun, C., Karlsson, P., Wu, J., Tenenbaum, J.B., Murphy, K.: Stochastic prediction of multi-agent interactions from partial observations. arXiv preprint arXiv:1902.09641 (2019) 4
61. Teranishi, M., Tsutsui, K., Takeda, K., Fujii, K.: Evaluation of creating scoring opportunities for teammates in soccer via trajectory prediction. In: International Workshop on Machine Learning and Data Mining for Sports Analytics. pp. 53–73. Springer (2022) 1

62. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017) 3, 4, 5, 6, 7
63. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. arXiv preprint arXiv:1710.10903 (2017) 4
64. Vidal-Codina, F., Evans, N., El Fakir, B., Billingham, J.: Automatic event detection in football using tracking data. Sports Engineering **25**(1), 18 (2022) 2, 5
65. Wang, Z., Veličković, P., Hennes, D., Tomašev, N., Prince, L., Kaisers, M., Bachrach, Y., Elie, R., Wenliang, L.K., Piccinini, F., et al.: Tacticai: an ai assistant for football tactics. arXiv preprint arXiv:2310.10553 (2023) 2
66. Xie, Y., Jampani, V., Zhong, L., Sun, D., Jiang, H.: Omnicontrol: Control any joint at any time for human motion generation. arXiv preprint arXiv:2310.08580 (2023) 5
67. Xu, C., Tan, R.T., Tan, Y., Chen, S., Wang, Y.G., Wang, X., Wang, Y.: Eqmotion: Equivariant multi-agent motion prediction with invariant interaction reasoning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1410–1420 (2023) 1, 10
68. Xu, Y., Bazarjani, A., Chi, H.g., Choi, C., Fu, Y.: Uncovering the missing pattern: Unified framework towards trajectory imputation and prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9632–9643 (2023) 2, 4, 10, 12, 13
69. Yeh, R.A., Schwing, A.G., Huang, J., Murphy, K.: Diverse generation for multi-agent sports games. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4610–4619 (2019) 2, 4, 10, 12
70. Yuan, Y., Weng, X., Ou, Y., Kitani, K.M.: Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9813–9823 (2021) 5
71. Zhan, E., Zheng, S., Yue, Y., Sha, L., Lucey, P.: Generating multi-agent trajectories using programmatic weak supervision. arXiv preprint arXiv:1803.07612 (2018) 1, 4, 10, 12
72. Zheng, S., Yue, Y., Hobbs, J.: Generating long-term trajectories using deep hierarchical networks. Advances in Neural Information Processing Systems **29** (2016) 1, 4