Machine Learning (2025) 114:38
https://doi.org/10.1007/s10994-024-06648-2

Interactive sequential generative models for team sports

Dennis Fassmeyer[1] · **Moritz Cordes**[1] · **Ulf Brefeld**[1]

Received: 8 October 2023 / Revised: 3 September 2024 / Accepted: 3 October 2024 /
Published online: 27 January 2025

Abstract
Understanding spatiotemporal coordination of players in team sports is key to movement models, pattern detection, and computational tactics. Existing generative models propose to capture all stochasticity by a single latent variable and may suffer from entangled representations, or aim to uncover interaction structures of players but then do not focus on their generative ability. As a remedy, we propose a hierarchical latent variable model for predicting trajectories of multiple players. In the generative model, both, discrete role assignments and a latent interaction graph are sampled to allow for different models in subsequent node updates and message passing operations between nodes, where standard Gaussian latent variables are employed per agent and timestep. We cast our approach as a variational autoencoder that provides a disentangled latent space to capture variability in team sport movements and propose a neural architecture for its optimization. We empirically evaluate our approach on tracking data from basketball and soccer and observe that our contribution outperforms the state-of-art in all experiments.

Keywords Soccer · Generative models · Graph recurrent neural networks · Socio-temporal dependencies · Trajectory forecasting

1 Introduction

Understanding the behavior of players and teams over time and space is crucial in sports, for example to address questions regarding their coordination, implementation of game plans, optimality of movements, etc. An interesting facet in this conglomerate constitute *social structures* or roles of players. In a football game, a defender may act as a striker

Editors: Philippe Lopes, Werner Dubitzky, Daniel Berrar, Jesse Davis.

✉ Dennis Fassmeyer
dennis.fassmeyer@leuphana.de

✉ Ulf Brefeld
ulf.brefeld@leuphana.de

Moritz Cordes
moritz.cordes@leuphana.de

[1] Machine Learning Group, Leuphana Universität Lüneburg, Universitätsallee 1, 21335 Lüneburg, Germany

and aim to create chaos and/or concede a goal while somebody else takes on her role as defender. Hence, roles of players change over time and may help to predict or explain a player's future position.

However, modeling the dynamics of team sports (Omidshafiei et al., 2022; Le et al., 2017; Yue et al., 2014; Liu et al., 2020) is challenging since player behavior and interaction patterns are often multi-modal and underlie temporal shifts (Makansi et al., 2022). Although most current methods for modeling sports data rely on graph encoding strategies (Kipf & Welling, 2016), they usually rely on only a single latent variable that needs to capture all stochasticity including social information (Zhan et al., 2019; Yeh et al., 2019; Sun et al., 2019; Omidshafiei et al., 2022). These methods are usually formulated as some form of variational autoencoder (Kingma & Welling, 2013; Rezende et al., 2014; Sohn et al., 2015) and may aim at learning latent edge types of an (assumed) underlying graph structure (Kipf et al., 2018; Graber & Schwing, 2020; Löwe et al., 2022). However, approaches that explicitly infer edge types neglect other potential factors of uncertainty that do not originate from mere interaction categories but equally affect multi-modal agent behavior. This usually results in limited generative capacity. Hence, there is a need for an approach that can expliclity represent the intricate relationship structures and characteristics of players in team sports (Passos & Davids, 2015; Radke & Orchard, 2023).

This paper aims to bridge the gap between generative ability and modeling interaction structures. We propose a novel approach for modeling multi-agent trajectories that enhances existing graph latent variable models by explicitly encoding discrete latent variables reflecting social structures. The idea grounds on the hypothesis that incorporating social dependencies into a model can provide a more comprehensive understanding of player-player relationships, ultimately leading to a more accurate generative model. More specifically, the key contributions of our paper are threefold: Firstly, we derive a novel variational objective by proposing a disentangled latent space that explicitly captures inherent characteristic traits. The latent space includes graph-based variables representing categorical agent roles and pairwise interactions that provide structure to an autoregressive graph neural network with further continuous latent variable. Secondly, we propose a tailored network architecture including inductive biases to be able to disentangle information in latent space. Finally, our hierarchical latent model outperforms existing state-of-art methods in trajectory prediction tasks in soccer and basketball.

The remainder is structured as follows. Section 2 briefly reviews related work and Sect. 3 introduces the problem setting. We introduce our main contribution in Sect. 4 and report on empirical results in Sect. 5. Section 6 concludes.

2 Related work

Forecasting human trajectories has traditionally been addressed as a deterministic regression problem, typically by minimizing negative log-likelihoods and presuming bi-variate Gaussian output distributions (Mohamed et al., 2020; Rudenko et al., 2020). However, assuming unimodal Gaussians is inappropriate in the presence of multi-modal densities which are common in team sports (Brefeld et al., 2019). Consequently, contemporary methods have suggested the use of latent variable models, such as conditional VAEs (Sohn et al., 2015), to account for the inherent stochasticity of trajectories. These methods can be grouped by their approach to model temporal and social dependencies as well as their latent representation.

Some methods, for example, enhance VRNNs to manage agent-agent relations using graph neural networks (Yeh et al., 2019; Sun et al., 2019; Zhan et al., 2019; Monti et al., 2021) while others generate trajectories in a non-autoregressive manner by collapsing the time axis into a single dimension (Casas et al., 2020; Salzmann et al., 2020; Bhattacharyya et al., 2021). There are also approaches that utilize transformer-based architectures to encode spatial and temporal relations (Girgis et al., 2021; Yuan et al., 2021). Various studies aim to enhance forecasting trajectories by augmenting latent information containing long-term goals of the involved agents (Mangalam et al., 2020, 2021; Zhao et al., 2021; Monti et al., 2021; Zhan et al., 2019; Fan et al., 2021; Choi et al., 2021; Girase et al., 2021). These approaches tend to model interactions implicitly by aggregating messages along social dimensions into spatiotemporal representations.

By contrast, our contribution explicitly infers agent interactions at prediction time. This inductive bias was firstly used by Kipf et al. (2018) who propose a variational autoencoder where a discrete latent code is learned for relations between agents in a causal graph. While the original formulation is confined to learning a single graph for an entire multi-agent episode, Graber and Schwing (2020) and Li et al. (2021a) propose dynamic generalizations where the interaction graph can change between time steps. Recently, Löwe et al. (2022) infer causal relations for distinct interaction graphs but shared dynamics while Li et al. (2021b) extend the approach by Kipf et al. (2018) by employing a continuous latent space to separate interactive factors from agent intentions.

3 Preliminaries

Given N sequences $\mathcal{D} = \{\boldsymbol{x}_{\leq T}^{(i)}\}_{i=1}^{N}$ with $\boldsymbol{x}_{\leq T} = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_T]$, encoding the movement of agents over time $1 \leq t \leq T$, our goal is to estimate the underlying data distribution of agent locations via maximizing the likelihood $p_\theta(\boldsymbol{x}_{\leq T})$. In practice, $p_\theta(\boldsymbol{x}_{\leq T})$ is often highly multi-modal, which renders a direct application of maximum likelihood inappropriate. Hence, we resort to variational methods, introduce latent variables and aim to optimize the variational lower bound on the marginal log-likelihood (Kingma & Welling, 2013; Rezende et al., 2014; Sohn et al., 2015).

Existing conditional variational models for generating highly-structured sequential data $\boldsymbol{x}_{\leq T}$ usually associate a latent variable $z_1, \dots, z_T$ with each timestep of the segment to describe the generative process (Bayer & Osendorfer, 2014; Goyal et al., 2017; Fraccaro et al., 2016). The variational RNN (Chung et al., 2015) arrives at the following lower bound on $\log p_\theta(\boldsymbol{x}_{\leq T})$, given by

$$\mathbb{E}_{q_\phi(\boldsymbol{z}_{\leq T}|\boldsymbol{x}_{\leq T})}\left[\sum_{t=1}^{T} \log p_\theta(\boldsymbol{x}_t|\boldsymbol{z}_{\leq t}, \boldsymbol{x}_{<t}) - \mathcal{KL}\big[q_\phi(\boldsymbol{z}_t|\boldsymbol{x}_{\leq t}, \boldsymbol{z}_{<t}) \parallel p_\theta(\boldsymbol{z}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{<t})\big]\right], \tag{1}$$

where the history over $\boldsymbol{x}_{<t}$ and $\boldsymbol{z}_{<t}$ is approximated via a recurrent neural network $\boldsymbol{h}_t = f_{\text{RNN}}(\boldsymbol{x}_t, \boldsymbol{z}_t, \boldsymbol{h}_{t-1})$ and $\mathcal{KL}$ is the KL-divergence between proposal distribution q_ϕ and prior p_θ over the latent variables. Given the temporal and multi-modal notion of human movement, sequential generative models constitute a promising starting point for designing a framework tailored to multi-agent trajectories. However, such approaches only account for the temporal aspect of the problem and neglect potential social interactions at each timestep. As a remedy, sequential data can be augmented by a social dimension

$\boldsymbol{x}_{\leq T} = \{\boldsymbol{x}^{(a)}_{\leq T}, \forall a \in \mathcal{A}\}$, where $\boldsymbol{x}^{(a)}_t \in \mathbb{R}^d$ denotes a d-dimensional feature representation of agent $a \in \mathcal{A}$ at time t (e.g., her 2D position).

Permutation invariant models are a prerequisite for processing sequential sets of agents with potentially divergent cardinality (e.g., due to red cards). Thus, one straight forward application of the VRNN objective in Eq. (1) would implicitly impose social independence across the involved agents and their trajectories. This assumption is clearly inappropriate for team sports; however, a possible solution is offered by graph-based approaches to explicitly capture agent interactions.

Yeh et al. (2019) introduce a graph-based generalization of the VRNN (Chung et al., 2015) to representing agents and their interactions as nodes and edges, respectively. Their VRNN components are computed by

$$p_\theta(\boldsymbol{z}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{<t}) = \mathcal{N}(\boldsymbol{z}_t; \mathrm{GNN}_{\mathrm{prior}}(\boldsymbol{h}_{t-1})) \tag{2}$$

$$q_\phi(\boldsymbol{z}_t|\boldsymbol{x}_{\leq t}, \boldsymbol{z}_{<t}) = \mathcal{N}(\boldsymbol{z}_t; \mathrm{GNN}_{\mathrm{enc}}([\boldsymbol{x}_t, \boldsymbol{h}_{t-1}])) \tag{3}$$

$$p_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{\leq t}) = \mathcal{N}(\boldsymbol{x}_t; \mathrm{GNN}_{\mathrm{dec}}([\boldsymbol{z}_t, \boldsymbol{h}_{t-1}])), \tag{4}$$

where $\mathrm{GNN}_{\mathrm{prior}}$, $\mathrm{GNN}_{\mathrm{enc}}$ and $\mathrm{GNN}_{\mathrm{dec}}$ in general may be any specific graph neural network variant (cf. Battaglia et al., 2018) and where $\boldsymbol{h}_t$ is the set of recurrent agent states $\boldsymbol{h}^{(a)}_t = f_{\mathrm{RNN}}(\boldsymbol{x}^{(a)}_t, \boldsymbol{z}^{(a)}_t, \boldsymbol{h}^{(a)}_{t-1})$. We emphasize that, although factorized, the latent space is not marginally independent across agents since each $\boldsymbol{z}^{(a)}_t$ is conditioned on information of all other entities via a (possibly fully-connected) interaction graph. However note that, in most game patterns, the majority of agents may be irrelevant or even distracting a successful pattern detection since the specific composition of relevant factors can change rapidly over time. A better strategy is thus to explicitly detect semantic classes that capture the underlying structures before aggregating social information into entangled variables.

4 Main contribution

We propose a novel way to capture the joint distribution of spatiotemporal data $p_\theta(\boldsymbol{x}_{\leq T})$ using variational methods. We provide conceptual ideas on how we should think about variational methods for interactive sequential sports data and a technical formulization of this line of thought, including an inductive bias for structuring the latent space.

4.1 The objective function

Structural dependencies in team-sports are highly dynamic over time and usually only implicitly represented. As an example, consider again the defender who is carrying the ball forward while a teammate takes over her original role in this situation. Once the situation is over, the two change their roles again and fall back into their standard roles and locations. To account for these structures, we propose to introduce latent graphs $\mathcal{G}_t = \{\mathcal{K}_t, \mathcal{E}_t\}$, where $\mathcal{K}_t = \{\mathcal{K}^{(a)}_t, a \in \mathcal{A}\}$ represents the set of discrete agent roles at time t (e.g. *attacker* or *defender*), and $\mathcal{E}_t = \{\mathcal{E}^{(a,j)}_t, (a,j) \in \mathcal{A} \times \mathcal{A}\}$ encodes their (discrete) pairwise interaction types (e.g. the interaction *ball-handler-to-intended-pass-receiver* etc.). Both agent roles and interaction types are latent variables that are inferred without supervision. How many roles and

interaction types are assumed in the model is determined by hyperparameters. We include a dedicated *no interaction* class as one type of interaction in $\mathcal{E}_t$ to allow the model to explicitly exclude some interactions.

In general, the representational capacity of $\mathcal{G}_t$ may be insufficient to holistically capture the full latent spectrum of multi-modal agent behavior arising from both individual and social perspectives. We thus augment the latent space with agent-wise continuous variables $z_t^{(a)} \in z_t$. This allows the latent representation to incorporate the characteristics of all remaining sources of uncertainty that are not represented by the inferred causal graphs. In addition, the z_t may vary as a function of $\mathcal{G}_t$. We are now able to formalize the generative process by incorporating the latent concepts $\{\mathcal{G}_t, z_t\}$ into the marginal likelihood,

$$p_\theta(\boldsymbol{x}_{\leq T}) = \sum_{G_{\leq T}} \int_{z_{\leq T}} p_\theta(\boldsymbol{x}_{\leq T}, z_{\leq T}, \mathcal{G}_{\leq T}) dz_{\leq T}$$

$$= \sum_{G_{\leq T}} \int_{z_{\leq T}} \prod_{t=1}^{T} \Big[p_{\theta_1}(\boldsymbol{x}_t|\boldsymbol{x}_{<t}, z_{\leq t}, \mathcal{G}_{\leq t}) p_{\theta_2}(z_t|\boldsymbol{x}_{<t}, z_{<t}, \mathcal{G}_{\leq t}) p_{\theta_3}(\mathcal{G}_t|\boldsymbol{x}_{<t}, z_{<t}, \mathcal{G}_{<t}) \Big] dz_{\leq T}$$

where the model likelihood is parameterized using a *decoder* p_{θ_1}, and *prior* distributions over the specified latent factors p_{θ_2} and p_{θ_3}.

Given the generative model, learning representations from spatiotemporal data can be posed as learning a variational approximation of the posterior using an *encoder* that generates the distribution of latent values for a given observation $\boldsymbol{x}_{\leq T}$. We propose to factorize the variational posterior $q_\phi(z_{\leq T}, \mathcal{G}_{\leq T}|\boldsymbol{x}_{\leq T})$ by

$$q_\phi = \prod_{t=1}^{T} q_{\phi_1}(\mathcal{G}_t|\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{<t}) q_{\phi_2}(z_t|\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{\leq t}).$$

Note that, as opposed to the VRNN objective in (1), we condition q_{ϕ_1} and q_{ϕ_2} also on future inputs $\boldsymbol{x}_{t+1:T}$. This has been shown to be empirically beneficial in sequential settings (Goyal et al., 2017; Fraccaro et al., 2016). To train the introduced concepts p_θ and q_ϕ, we derive the objective function $\mathcal{L}(\boldsymbol{x}_{\leq T}; \boldsymbol{\phi}, \theta)$ as a a lower-bound on the log-likelihood $\log p_\theta(\boldsymbol{x}_{\leq T})$ given by

$$\log p_\theta(\boldsymbol{x}_{\leq T}) \geq \mathbb{E}_{q_\phi(z_{\leq T}, \mathcal{G}_{\leq T}|\boldsymbol{x}_{\leq T})} \Bigg[\sum_{t=1}^{T} \log p_{\theta_1}(\boldsymbol{x}_t|\boldsymbol{x}_{<t}, z_{\leq t}, \mathcal{G}_{\leq t}) \tag{5}$$

$$- \mathcal{KL}\big[q_\phi(z_t, \mathcal{G}_t|\boldsymbol{x}_{\leq T}) \parallel p_\theta(z_t, \mathcal{G}_t|\boldsymbol{x}_{<t}, z_{<t}, \mathcal{G}_{<t})\big] \Bigg] \tag{6}$$

$$= \mathbb{E}_{q_\phi(z_{\leq T}, \mathcal{G}_{\leq T}|\boldsymbol{x}_{\leq T})} \Bigg[\sum_{t=1}^{T} \log p_{\theta_1}(\boldsymbol{x}_t|\boldsymbol{x}_{<t}, z_{\leq t}, \mathcal{G}_{\leq t}) \tag{7}$$

$$+ \log p_{\theta_2}(z_t|\boldsymbol{x}_{<t}, z_{<t}, \mathcal{G}_{\leq t}) + \log p_{\theta_3}(\mathcal{G}_t|\boldsymbol{x}_{<t}, z_{<t}, \mathcal{G}_{<t}) \tag{8}$$

$$- \log q_{\phi_1}(\mathcal{G}_t|\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{<t}) - \log q_{\phi_2}(z_t|\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{\leq t}) \Bigg]. \tag{9}$$

The derivation is analogously to the lower bound for VRNNs (Chung et al, 2015, compare Equation (1)). We impose a further factorization of the prior on $\mathcal{G}_t$ with

$$p_{\theta_3}(\mathcal{G}_t|\boldsymbol{x}_{<t}, z_{<t}, \mathcal{G}_{<t}) = p_{\theta_3}(\mathcal{K}_t|\boldsymbol{x}_{<t}, z_{<t}, \mathcal{G}_{<t}) p_{\theta_3}(\mathcal{E}_t|\boldsymbol{x}_{<T}, z_{<t}, \mathcal{G}_{<t})$$

and analogously on the variational posterior, where we set

$$q_{\phi_1}(\mathcal{G}_t|\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{<t}) = q_{\phi_1}(\mathcal{K}_t|\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{<t}) q_{\phi_1}(\mathcal{E}_t|\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{<t}).$$

4.2 Architectural components

Given past realizations $\boldsymbol{x}_{<t}, z_{<t}, \mathcal{G}_{<t}$, the introduced structure requires to first infer social component $\mathcal{G}_t$ before subsequently generating variables z_t and $\boldsymbol{x}_t$. Hence, we divide the training procedure at each timestep into two distinct phases. The first phase addresses the computation of the interaction graph $\mathcal{G}_t$ (that is, p_{θ_3} and q_{ϕ_1}), while the subsequent second phase infers components over z_t and $\boldsymbol{x}_t$ (hence involving p_{θ_1}, p_{θ_2} and q_{ϕ_2}). We describe the computations in more detail below.

Learning the Interaction Graph We begin with the encoder and prior estimating the latent graph $\mathcal{G}_t$ for $1 \leq t \leq T$. Since $\mathcal{G}_t$ is defined over discrete agent roles $\mathcal{K}_t$ and their interaction $\mathcal{E}_t$, we model encoder and prior by (conditionally independent) Categorical distributions

$$q_{\phi_1}(\mathcal{K}_t|\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{<t}) = \prod_{a \in \mathcal{A}} \mathrm{Cat}\Big(\mathcal{K}_t^{(a)}|f_{\mathrm{enc}}^{\mathrm{int}}(\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{<t}; \phi_1)\Big) \tag{10}$$

$$q_{\phi_1}(\mathcal{E}_t|\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{<t}) = \prod_{(i,j) \in \mathcal{A} \times \mathcal{A}} \mathrm{Cat}\Big(\mathcal{E}_t^{(i,j)}|f_{\mathrm{enc}}^{\mathrm{int}}(\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{<t}; \phi_1)\Big), \tag{11}$$

where $f_{\mathrm{enc}}^{\mathrm{int}}$ is an encoder neural network. Since the interaction space contains a label for *no interaction*, the resulting probability values over $\mathcal{E}_t$ can be interpreted as *importance values* that encode the level of influence of interacting neighbors on a agent of interest.

While elements in $\mathcal{K}_t$ are assigned to all agents on the pitch (at each time step), interactive patterns $\mathcal{E}_t$ are assigned non-symmetrically to all pairs of agents. Thus, $f_{\mathrm{enc}}^{\mathrm{int}}$ needs to be able to learn agent-specific as well as agent-agent representations. Inspired by Kipf et al. (2018), we employ a variant of GNNs similar to interaction networks (Gilmer et al., 2017) that operate by updating node and edge representations through an iterative process where the node and edge representations are calculated via neural networks.[1] The computations in a single layer of the GNN variant are given by

$$v \to e: \quad \boldsymbol{e}^{(i,j)} = f_e([\boldsymbol{v}_i, \boldsymbol{v}_j]), \tag{12}$$

[1] Other commonly used graph encoding strategies for multi-agent scenarios such as graph attention networks (Veličković et al., 2017) or transformer architectures (Vaswani et al., 2017) update node embeddings via learned attention weights and consequently do not explicitly learn vectorial edge representations, rendering them inappropriate for the problem at hand.

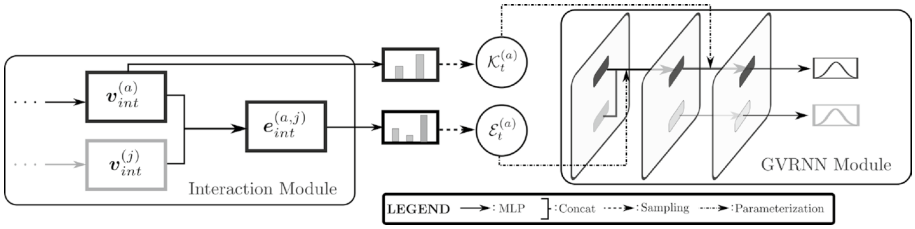

Fig. 1 The left hand side shows the output layer of an interaction module (encoder q_{ϕ_1} or prior p_{θ_2}) for target agent $a \in \mathcal{A}$ and neighboring agent $j \in \mathcal{A}$, where $\boldsymbol{v}$ and $\boldsymbol{e}$ are node and edge embeddings, respectively. The right side shows an illustration of the interaction network (decoder p_{θ_1}, prior p_{θ_2}, or encoder q_{ϕ_2}) that is parameterized by the discrete latent subspace: $\mathcal{E}_t^{(a)}$ picks from multiple node-to-edge MLPs, and $\mathcal{K}_t^{(a)}$ picks from multiple edge-to-node MLPs. Thus, implicit knowledge about fundamentally different interaction patterns is encoded in the parameters of the edge MLPs associated with the corresponding interaction class

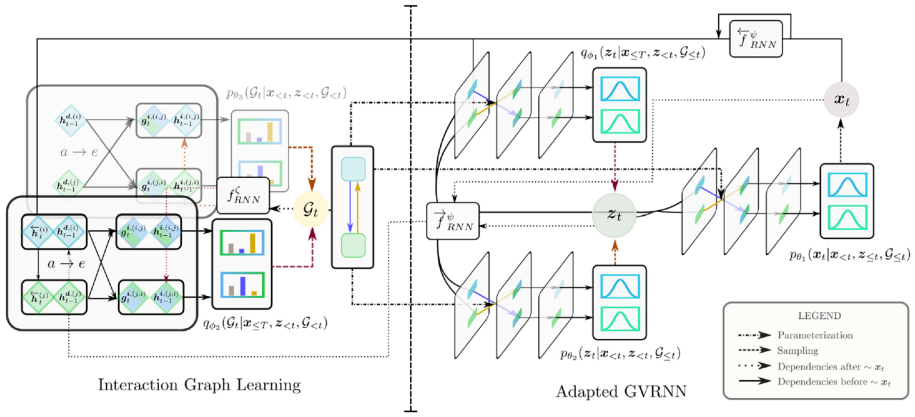

Fig. 2 Illustration of the computational dependencies at a timestep for a system with two interacting agents when using only one agent type. We only show the first layer of the overall two-layer graph network in the interaction part (left) to show the injection of the GRU states. Red connections denote the parts utilized only during training, while connections in orange color denote parts used during testing. Deterministic and stochastic variables are depicted by diamonds and circles, respectively

$$e \to v: \quad \boldsymbol{o}^{(i)} = f_v\Big(\sum_{j \in N(i)} \boldsymbol{e}^{(i,j)}\Big), \tag{13}$$

where $\boldsymbol{v}_i$ is the input representation of agent i, $\boldsymbol{o}_i$ is the updated agent embedding, $N(i)$ denotes the set of agents that interact with target agent i, $\boldsymbol{e}^{(i,j)}$ is the edge embedding for agent pair (i, j), and f_e and f_v are shallow networks.

The input to the encoder $f_{\text{enc}}^{\text{int}}(\boldsymbol{x}_{\leq T}, \boldsymbol{z}_{<t}, \mathcal{G}_{<t})$ is given via multiple GRU cells (Chung et al., 2014) defined over quantities related to the individual agents. As such, $f_{\text{enc}}^{\text{int}}$ is defined as

$$[\boldsymbol{v}_{\text{enc}}^{\text{int}}, \boldsymbol{e}_{\text{enc}}^{\text{int}}] = f_{\text{enc}}^{\text{int}}(\boldsymbol{x}_{\leq T}, \boldsymbol{z}_{<t}, \mathcal{G}_{<t}) = \text{GNN}_{\text{enc}}^{\text{int}}([\overrightarrow{\boldsymbol{h}}_{t-1}, \overleftarrow{\boldsymbol{h}}_t], \boldsymbol{h}_{t-1}^{\text{int}}; \phi_1),$$

where $[\cdot, \cdot]$ denotes concatenation, $\overrightarrow{\boldsymbol{h}}_{t-1}$ is a set of forward RNN states $f_{\text{RNN}}(\boldsymbol{x}_t^{(a)}, \boldsymbol{z}_t^{(a)}, \overrightarrow{\boldsymbol{h}}_{t-1}^{(a)})$, $\overleftarrow{\boldsymbol{h}}_t$ is a set of backward RNN states $f_{\text{RNN}}(\boldsymbol{x}_t^{(a)}, \overleftarrow{\boldsymbol{h}}_{t+1}^{(a)})$, $\boldsymbol{h}_{t-1}^{\text{int}}$ is a set of hidden states $f_{\text{RNN}}(\mathcal{K}_{t-1}^{(a)}, \mathcal{E}_{t-1}^{(a,j)}, \boldsymbol{h}_{t-1}^{(a,j),\text{int}})$ encoding agent and interaction types of past timesteps in a

fully-connected graph structure, and $\boldsymbol{v}_{\text{enc}}^{\text{int}}, \boldsymbol{e}_{\text{enc}}^{\text{int}}$ are the node and edge embeddings of the last/output GNN layer, respectively, as shown in Figs. 1 and 2.

Finally, we model the distributions in Equations (10) and (11) as

$$q_{\phi_1}(\mathcal{G}_t|\boldsymbol{x}_{\leq T}, \boldsymbol{z}_{<t}, \mathcal{G}_{<t}) = \text{softmax}(\text{Linear}(f_{\text{enc}}^{\text{int}}(\boldsymbol{x}_{\leq T}, \boldsymbol{z}_{<t}, \mathcal{G}_{<t}; \boldsymbol{\phi}_1))),$$

see again Fig. 1. The prior distribution $p_{\theta_3}(\mathcal{G}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{<t}, \mathcal{G}_{<t})$ is derived analogously to the interaction encoder, omitting the summarized future $\overleftarrow{\boldsymbol{h}}_t$ as GNN input. While the probability values assigned to each interaction type can be directly construed as importance scores of interacting agents, attention-based aggregation mechanisms in graph networks tend to dilute attention to truly relevant information (Vemula et al., 2018; Shen et al., 2018). In contrast, we force the model to make binary decisions about potentially multiple social categories by realizing (differentiable) samples $\mathcal{G}_t \sim q_{\phi_1}, p_{\theta_3}$ using the Gumbel softmax trick (Maddison et al., 2017; Jang et al., 2017).

Generation Module

Once $\mathcal{G}_t$ is generated, the model needs to reason over the remaining components of the objective in Equation (5). In general, we can use any graph encoding strategy that uses the inferred latent subspace $\mathcal{G}_t$ to model the desired Gaussian distributions over $\boldsymbol{x}_t$ and $\boldsymbol{z}_t$. Thus, using interaction networks constitutes a reasonable choice for our purpose, as there is already empirical evidence of their effectiveness for modeling sports tracking data (Yeh et al., 2019; Dick et al., 2022).

For presentation purposes, we focus on an interactive system consisting of only two agents and formally define computations for an atomic graph $i \leftarrow j$ with agents $i, j \in \mathcal{A}$. To accommodate $\mathcal{G}_t$, we parameterize the node and edge functions inherent in each graph network. Assuming $1 \leq k \leq K$ different agent types and $1 \leq m \leq M$ distinct pairwise interaction labels, we introduce weight matrices ψ_k and γ_m to parameterize the corresponding node and edge MLPs f_v and f_e, respectively,

$$\boldsymbol{e}^{(i,j)} = f_e([\boldsymbol{v}^{(i)}, \boldsymbol{v}^{(j)}]; \psi_k) \qquad \text{for} \quad \mathcal{E}_t^{(i,j)} = k \tag{14}$$

$$\boldsymbol{o}^{(i)} = f_v(\boldsymbol{e}^{(i,j)}; \gamma_m) \qquad \text{for} \quad \mathcal{K}_t^{(i)} = m. \tag{15}$$

That is, updates in the generation module of our architecture are carried out with neural networks assigned by the node and edge types provided by the interaction module. As before, the node and edge input to each graph network is provided via GRU networks. The means and variances of the Gaussians over $\boldsymbol{z}_t$ and $\boldsymbol{x}_t$ are computed by MLPs that operate on the output node embeddings. See Fig. 1 for a visualization of the introduced computation strategy.

Training & Testing Instead of absolute positions, our model predicts movements $\Delta\hat{\boldsymbol{x}}_t$ at each timestep. Consequently, we estimate agent locations via $\hat{\boldsymbol{x}}_t = \boldsymbol{x}_{t-1} + \Delta\hat{\boldsymbol{x}}_t$. For simplicity, $\boldsymbol{x}_t$ refers to both relative and absolute positions. For training, the model uses the entire $T = T_{obs} + T_{pred}$ timesteps from ground-truth sequences $\boldsymbol{x}_{\leq T}$. To enable gradient flow through stochastic operations, we use the reparameterization trick (Kingma & Welling, 2013; Rezende et al., 2014) and Gumbel-softmax (Maddison et al., 2017; Jang et al., 2017) for sampling from the encoders over the continuous $\boldsymbol{z}_t$ and the discrete $\mathcal{G}_t$ latent subspace, respectively. At test time, we divide the trajectories into an observation and prediction period, where the model only observes the first portion of the

ground-truth trajectory $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{T_{obs}}\}$ and predicts the remaining T_{pred} timesteps autoregressively: $\hat{\boldsymbol{x}}_{T_{obs}+i}, i \in \mathbb{Z}$. Latent variables are sampled from the prior distributions.

5 Experiments

In this section, we empirically evaluate our model *Detecting Important Agents* (DIA) on real-world data from soccer and basketball. The former contains trajectories of soccer players and the ball extracted from 16 elite soccer matches sampled at 25 frames per second provided by the German Football Association (DFB). We assemble a data set consisting of short game segments such that every center frame of the sequences corresponds to an on-ball event from the set $\mathcal{Y} = \{\text{pass, shot, other ball action, none}\}$. We choose a sequence length of 2 seconds and downsample the data to $10Hz$ so that we end with $T = 20$ for all sequences. This extraction process yields a total of roughly 34000 multi-agent segments. We randomly divide the collection into 70% training, 15% validation, and 15% test sets.

The basketball data has been released by STATS and consists of tracking data recorded from the 2016 NBA regular season.[2] Every game sequence has two-dimensional positions of 10 players and the ball sampled at 5 frames per second. The data is split into 60% training, 20% validation, and 20% test sets.

Baselines We compare our approach with several baselines, including *Weak-Sup* (Zhan et al., 2019), *GVRNN* (Yeh et al., 2019), *dNRI* (Graber & Schwing, 2020), *DAG-Net* (Monti et al., 2021), *Joint-β-cVAE* (Bhattacharyya et al., 2021) and *GRIN* (Li et al., 2021b). To enable direct comparison against this diverse set of methods, we benchmark in two experimental configurations.

The first configuration acknowledges that DAG-Net and Weak-Sup naturally excel at generating agent trajectories, as they integrate future locations via heuristic labels during prediction, thus potentially skewing a direct comparison with fully unsupervised generative methods, such as ours. To mitigate this bias, we measure quantitative results on the basketball data set over an extended prediction horizon, specifically we provide the model with an observation history of 10 timesteps and generate the next 40 timesteps as stochastic prediction ($T_{obs} = 10, T_{pred} = 40$). By contrast, for the baseline models GRIN and dNRI, long-term predictions are not as direct and we resort to compare the setting with $T_{obs} = 40$ and $T_{pred} = 10$ instead. Both models are evaluated on the soccer trajectories as well, where we use $T_{obs} = 10$ and $T_{pred} = 10$ for all models. GVRNN, Joint-β-cVAE, and our proposed framework are applicable across the full spectrum of prediction scenarios.

We measure performance in terms of Average Displacement Error (ADE) and Final Displacement Error (FDE). ADE refers to the l_2 error between the predicted locations and the ground truth averaged over the entire trajectory, while FDE is the l_2 error for the last predicted point (Alahi et al., 2014). Following common practice in existing literature, we report the minimum displacement over 20 generated samples.

5.1 Quantitative evaluation

Baseline Comparisons In the first set of experiments, we focus on trajectory forecasting and benchmark our model against recent generative methods. Our empirical findings are

[2] https://github.com/linouk23/NBA-Player-Movements

Table 1 Results for basketball and soccer

		Joint-β-cVAE	GVRNN	Weak-Sup	DAG-Net	DIA
basketball	min ADE	10.64	9.73	9.47	8.98	**8.29**
$T_{obs} = 10, T_{pred} = 40$	min FDE	14.47	15.80	16.98	14.08	**12.05**
		Joint-β-cVAE	GVRNN	dNRI	GRIN	DIA
basketball	min ADE	4.03	2.60	2.77	3.00	**2.20**
$T_{obs} = 40, T_{pred} = 10$	min FDE	6.56	5.66	5.52	6.12	**4.51**
soccer	min ADE		7.48	7.60	7.88	**7.02**
$T_{obs} = 10, T_{pred} = 10$	min FDE		10.72	10.88	10.54	**9.68**

Table 2 Importance of latent structure versus latent capacity

	minADE	minFDE	# Parameters
GVRNN (best)	9.73	15.80	355588
GVRNN (x2 width)	9.91	16.12	733700
GVRNN (x3 width)	9.94	16.20	1647364
DIA (no structure)	10.27	15.17	400836
DIA	8.29	12.05	421768

summarized in Table 1, where the top rows for basketball contain long-term predictions and the bottom rows contain short-term predictions. Throughout all experiments, our DIA emerges as the best generative tool across all tested tasks improving the best (unsupervised) competitor performance by at least 17.3% minADE and 25.5% minFDE for basketball. Remarkably, despite their inherent advantages, DIA also outperforms recent supervised generative methods (Weak-Sup and DAG-Net) by at least 8.3% minADE and 16.8% minFDE. Furthermore, we observe that methods learning global latent variables (GRIN and Joint-β-cVAE) perform significantly worse than their peers. These methods, however, were designed to address prediction tasks in urban environments rather than sports where interaction patterns are known to be more dynamic and heterogeneous (Makansi et al., 2022). On the experiment with soccer trajectories our model also outperforms all baseline models.

Capacity vs Structure We now focus on the trade-off between our proposed interaction graph and latent capacity. To do so, we increase latent, hidden, and RNN dimensions of the GVRNN and compare the altered baseline to a simplified version of DIA where we discarded the graph component so that it basically resembles a deep GVRNN.

Table 2 shows the results. Unsurprisingly, increasing the GVRNN's latent capacity in encoder, decoder, and prior degenerates performance. GVRNNs represent interactions by a fully-connected graph, so that adding more capacity does not add to the receptive field and cannot be translated into increased expressivity. In the end, the GVRNNs become overparameterized and more difficult to train. By contrast, the results for DIA highlight the importance of the interaction graph for the problem at hand, since the only difference between the two variants is the proposed interaction graph.

To further quantify the benefit of augmenting the latent space with an explicit causal graph $\mathcal{G}_t$, we compare performance metrics of the GVRNN and another modified version

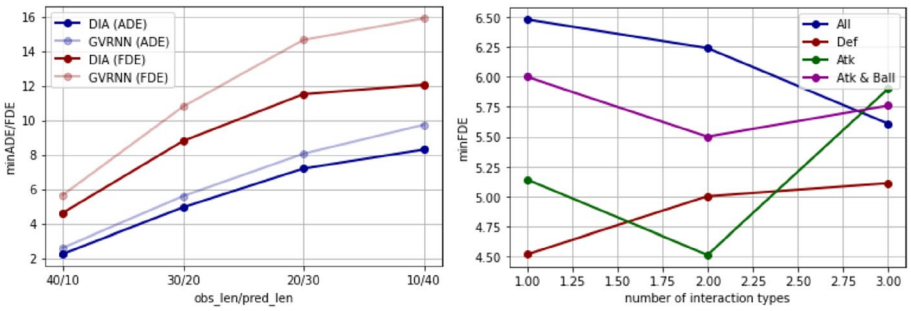

Fig. 3 *Left:* Comparison between GVRNN and an adapted version of DIA for different observation and prediction horizons. *Right:* Results for different agent subgroups where we fix $\mathcal{K}_t$ and vary the dimensionality of the interaction space $\mathcal{E}_t$

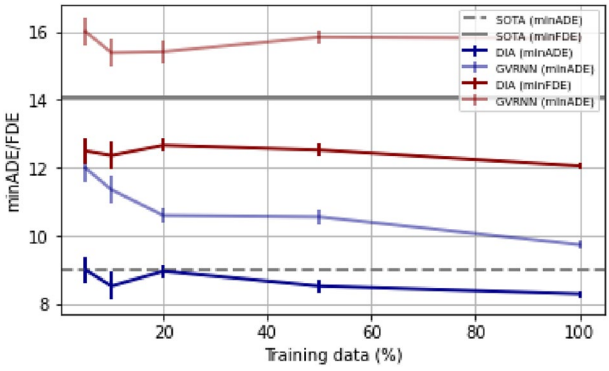

Fig. 4 Results for varying sample sizes

of DIA at varying observation and prediction lengths. The DIA architecture is altered such that the second part of the training procedure is identical to a GVRNN. In this way, the resulting performance differences capture only influences originating from the proposed latent graph. The left part of Fig. 3 shows the results. The modified DIA yields substantially lower displacement errors over the GVRNN. These differences increase with task complexity given by longer prediction horizons. The results thus highlight that modeling relations via attention-based aggregation strategies as in GVRNNs is insufficient to capture decisive social signals in non-trivial multi-agent scenarios.

The right part of Fig. 3 shows predictive results for different agent subgroups for varying representational capacities of the latent interaction graph $\mathcal{E}_t \in \mathcal{G}_t$. Recall that every interaction class has its own parameters so that increasing the dimensionality of the interaction space may lead to overfitting, despite higher model expressivity. We conjecture that the best results are realized by sufficiently small dimensionalities that still allow for capturing the underlying dynamics. Accordingly, this experimental setup sheds light on whether the proposed interaction mechanism $\mathcal{E}_t$ behaves according to our theoretical considerations. For example, in modeling defensive players, we would expect reactive patterns based on the behavior of attacking players, resulting in little intra-group dependencies. Indeed, the results in Fig. 3 indicate that our model inherits the expected social behaviors, as the best

parameter configuration is in line with the structural complexity of the different player subgroups.

Sample Size We now vary the amount of training data to study whether the additional latent graph in DIA leverages to data efficient training processes. We randomly sample 5%, 10%, 20%, and 50% of all available basketball data and then train both, DIA and GVRNN, on the same subsets. Figure 4 shows average minADE and minFDE, error bars indicate standard errors. Grey lines indicate the best reported results for this task on that data for comparison (SOTA). Remarkably, our model significantly outperforms all GVRNNs in both metrics using only 5% of the data. Our DIA discovers the underlying patterns much faster than GVRNNs, whose performance improves significantly with more available data. Our DIA manages to govern its attention to relevant aspects of even little data because interaction graph accurately captures the latent ground-truth factors that arise from social interactions.

Convergence

To determine the degree to which latent variables z_t capture useful information for predicting future agent trajectories, we monitor the KL divergence between the variational posterior and prior. Posterior collapse is a frequent issue in existing sequential latent variable models, where the model converges to regions of the loss surface that contain bad local minima (or saddle points) at $KL = 0$, leading to z_t essentially becoming Gaussian noise and the model behaving like an unconditional RNN. We display the values of $\mathcal{L}_{DIA}$ and $\mathcal{KL}[q_{\phi_1}(z_t|\boldsymbol{x}_{\leq T}, z_{<t}, \mathcal{G}_{\leq t}) \parallel p_{\theta_2}(z_t|\boldsymbol{x}_{<t}, z_{<t}, \mathcal{G}_{\leq t})]$ in Fig. 5, obtained from a training run on the soccer data. We observe a gradual increase in KL loss values that is accompanied by a steady decrease in training loss. These results suggest an effective exploitation of the continuous subspace z_t to generate future agent movements without the necessity of using common optimization strategies such as cost annealing (Bowman et al., 2016; Sønderby et al., 2016).

5.2 Qualitative evaluation

To provide more insights in the learned sequential graphs $\mathcal{G}_{\leq T}$ of DIA, we visually depict a multi-agent segment from the test set and estimated probability values within two interaction categories (and one "no interaction" category) in Fig. 6. The upper right image

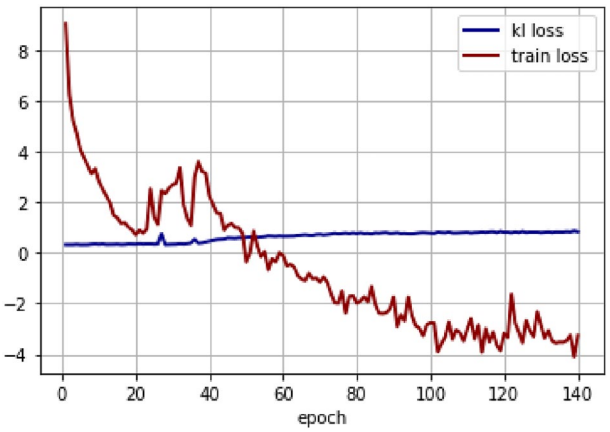

Fig. 5 The full training loss $\mathcal{L}_{DIA}$ and KL loss between prior and approximate posterior on $z_{\leq T}$ during a training run on soccer data

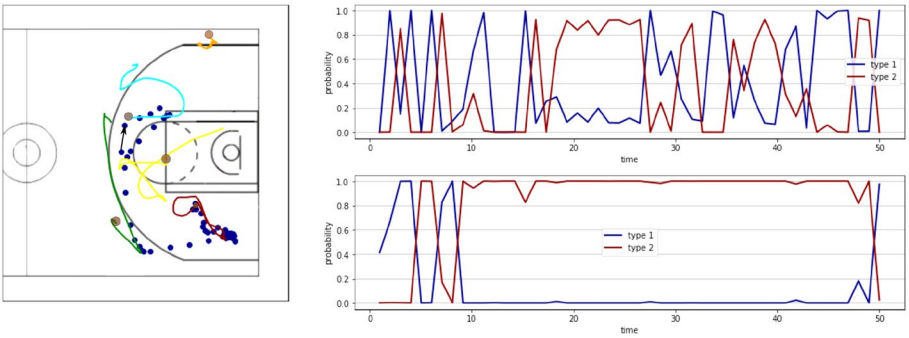

Fig. 6 *Left:* Trajectories of the offensive players and the ball (blue dots) from a data point in the test set of the basketball data. *Top right:* Influence $\mathcal{E}^{(b,r)}$ of the red player r on the ball b. *Bottom right:* Influence of the ball on the red player

shows the influencing factors of the red player on the ball. This interaction structure is mainly characterized by frequent alternations between the two interaction types with scarce instants of low aggregated influence. Since the interaction values are determined to accurately reflect the input data, the fluctuating pattern illustrates highly dynamic social structures in our tested setting.

6 Conclusion

We presented a hierarchical variational autencoder for modeling the joint distribution of agent trajectories. In the proposed data generation process, a discrete latent graph is sampled to capture social structures at a given timestep, which informs subsequent computations including sampling of a continuous latent variable for all remaining factors. We demonstrated that the emerging architecture performs better in predicting trajectories compared to existing strategies.

Acknowledgements We would like to thank the German Football Association (DFB) for sharing the data. We also thank Yannick Rudolph for discussions on the technical contribution and empirical setups as well as for proofreading the manuscript at different stages.

Funding Open Access funding enabled and organized by Projekt DEAL.

References

Alahi, A., Ramanathan, V., & Fei-Fei, L. (2014). Socially-aware large-scale crowd forecasting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 2203–2210.

Battaglia, P. W., Hamrick, J. B., Bapst, V., et al. (2018). Relational inductive biases, deep learning, and graph networks. arXiv preprint arXiv:1806.01261.

Bayer, J., & Osendorfer, C. (2014). Learning stochastic recurrent networks. arXiv preprint arXiv:1411.7610.

Bhattacharyya, A., Reino, D. O., Fritz, M., et al. (2021). Euro-pvi: Pedestrian vehicle interactions in dense urban centers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6408–6417.

Bowman, S. R., Vilnis, L., Vinyals, O., et al. (2016). Generating sentences from a continuous space. In: 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Association for Computational Linguistics (ACL), pp 10–21.

Brefeld, U., Lasek, J., & Mair, S. (2019). Probabilistic movement models and zones of control. *Machine Learning, 108*(1), 127–147.

Casas, S., Gulino, C., Suo, S., et al. (2020). Implicit latent variable model for scene-consistent motion forecasting. In: European Conference on Computer Vision, pp 624–641.

Choi, C., Malla, S., Patil, A., et al. (2021). Drogon: A trajectory prediction model based on intention-conditioned behavior reasoning. In: Conference on Robot Learning, PMLR, pp 49–63.

Chung, J., Gulcehre, C., Cho, K., et al. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.

Chung, J., Kastner, K., Dinh, L., et al. (2015). A recurrent latent variable model for sequential data. Advances in neural information processing systems 28.

Dick, U., Link, D., & Brefeld, U. (2022). Who can receive the pass? a computational model for quantifying availability in soccer. *Data Mining and Knowledge Discovery, 36*(3), 987–1014.

Fan, S., Li, X., & Li, F. (2021). Intention-driven trajectory prediction for autonomous driving. In: 2021 IEEE Intelligent Vehicles Symposium (IV), IEEE, pp 107–113.

Fraccaro, M., Sønderby, S. K., Paquet, U., et al. (2016). Sequential neural models with stochastic layers. Advances in neural information processing systems 29.

Gilmer, J., Schoenholz, S. S., Riley, P. F., et al. (2017). Neural message passing for quantum chemistry. In: International conference on machine learning, PMLR, pp 1263–1272.

Girase, H., Gang, H., Malla, S., et al. (2021). Loki: Long term and key intentions for trajectory prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 9803–9812.

Girgis, R., Golemo, F., Codevilla, F., et al. (2021). Latent variable sequential set transformers for joint multi-agent motion prediction. In: International Conference on Learning Representations.

Goyal, A., Sordoni, A., Côté, M. A., et al. (2017). Z-forcing: Training stochastic recurrent networks. arXiv preprint arXiv:1711.05411.

Graber, C., & Schwing, A. (2020). Dynamic neural relational inference for forecasting trajectories. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp 1018–1019.

Jang, E., Gu, S., & Poole, B. (2017). Categorical reparametrization with gumble-softmax. In: International Conference on Learning Representations (ICLR 2017), OpenReview. net.

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

Kipf, T., Fetaya, E., Wang, K. C., et al. (2018). Neural relational inference for interacting systems. In: International Conference on Machine Learning, PMLR, pp 2688–2697.

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.

Le, H. M., Yue, Y., Carr, P., et al. (2017). Coordinated multi-agent imitation learning. In: International Conference on Machine Learning, PMLR, pp 1995–2003.

Li, J., Yang, F., Ma, H., et al. (2021a). Rain: Reinforced hybrid attention inference network for motion forecasting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 16096–16106.

Li, L., Yao, J., Wenliang, L., et al. (2021). Grin: Generative relation and intention network for multi-agent trajectory prediction. *Advances in Neural Information Processing Systems, 34*, 27107–27118.

Liu, G., Schulte, O., Poupart, P., et al. (2020). Learning agent representations for ice hockey. *Advances in Neural Information Processing Systems, 33*, 18704–18715.

Löwe, S., Madras, D., Zemel, R., et al. (2022). Amortized causal discovery: Learning to infer causal graphs from time-series data. In: Conference on Causal Learning and Reasoning, PMLR, pp 509–525.

Maddison, C., Mnih, A., & Teh, Y. (2017). The concrete distribution: A continuous relaxation of discrete random variables. In: Proceedings of the international conference on learning Representations, International Conference on Learning Representations.
Makansi, O., von Kügelgen, J., Locatello, F., et al. (2022). You mostly walk alone: Analyzing feature attribution in trajectory prediction. In: International Conference on Learning Representations (ICLR).
Mangalam, K., Girase, H., Agarwal, S., et al. (2020). It is not the journey but the destination: Endpoint conditioned trajectory prediction. In: European conference on computer vision, Springer, pp 759–776.
Mangalam, K., An, Y., Girase, H., et al. (2021). From goals, waypoints & paths to long term human trajectory forecasting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 15233–15242.
Mohamed, A., Qian, K., Elhoseiny, M., et al. (2020). Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 14424–14432.
Monti, A., Bertugli, A., Calderara, S., et al. (2021). Dag-net: Double attentive graph neural network for trajectory forecasting. In: 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, pp 2551–2558.
Omidshafiei, S., Hennes, D., Garnelo, M., et al. (2022). Multiagent off-screen behavior prediction in football. *Scientific reports, 12*(1), 1–13.
Passos, P., & Davids, K. (2015). Learning design to facilitate interactive behaviours in team sports. RICYDE Revista internacional de ciencias del deporte 39(11).
Radke, D., & Orchard, A. (2023). Presenting multiagent challenges in team sports analytics. Tech. Rep. arXiv:2303.13660.
Rezende, D. J., Mohamed, S., & Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In: International conference on machine learning, PMLR, pp 1278–1286.
Rudenko, A., Palmieri, L., Herman, M., et al. (2020). Human motion trajectory prediction: A survey. *The International Journal of Robotics Research, 39*(8), 895–935.
Salzmann, T., Ivanovic, B., Chakravarty, P., et al. (2020). Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In: European Conference on Computer Vision, Springer, pp 683–700.
Shen, T., Zhou, T., Long, G., et al. (2018). Reinforced self-attention network: A hybrid of hard and soft attention for sequence modeling. In: IJCAI International Joint Conference on Artificial Intelligence.
Sohn, K., Lee, H., & Yan, X. (2015). Learning structured output representation using deep conditional generative models. Advances in neural information processing systems 28.
Sønderby, C. K., Raiko, T., Maaløe, L., et al. (2016). Ladder variational autoencoders. Advances in neural information processing systems 29.
Sun, C., Karlsson, P., Wu, J., et al. (2019). Stochastic prediction of multi-agent interactions from partial observations. arXiv preprint arXiv:1902.09641.
Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention is all you need. Advances in neural information processing systems 30.
Veličković, P., Cucurull, G., Casanova, A., et al. (2017). Graph attention networks. arXiv preprint arXiv:1710.10903.
Vemula, A., Muelling, K., & Oh, J. (2018). Social attention: Modeling attention in human crowds. In: 2018 IEEE international Conference on Robotics and Automation (ICRA), IEEE, pp 4601–4607.
Yeh, R. A., Schwing, A. G., Huang, J., et al. (2019). Diverse generation for multi-agent sports games. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 4610–4619.
Yuan, Y., Weng, X., Ou, Y., et al. (2021). Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 9813–9823.
Yue, Y., Lucey, P., Carr, P., et al. (2014). Learning fine-grained spatial models for dynamic sports play prediction. In: 2014 IEEE international conference on data mining, IEEE, pp 670–679.
Zhan, E., Zheng, S., Yue, Y., et al. (2019). Generating multi-agent trajectories using programmatic weak supervision. In: International Conference on Learning Representations.
Zhao, H., Gao, J., Lan, T., et al. (2021). Tnt: Target-driven trajectory prediction. In: Conference on Robot Learning, PMLR, pp 895–904.