GroupNet: Multiscale Hypergraph Neural Networks for Trajectory Prediction with Relational Reasoning

Chenxin Xu[1], Maosen Li[1], Zhenyang Ni[1], Ya Zhang[1,2], Siheng Chen[1,2*]

[1]Shanghai Jiao Tong University, [2]Shanghai AI Laboratory

{xcxwakaka,maosen_li,0107nzy,ya_zhang,sihengc}@sjtu.edu.cn

Abstract

Demystifying the interactions among multiple agents from their past trajectories is fundamental to precise and interpretable trajectory prediction. However, previous works only consider pair-wise interactions with limited relational reasoning. To promote more comprehensive interaction modeling for relational reasoning, we propose GroupNet, a multiscale hypergraph neural network, which is novel in terms of both interaction capturing and representation learning. From the aspect of interaction capturing, we propose a trainable multiscale hypergraph to capture both pair-wise and group-wise interactions at multiple group sizes. From the aspect of interaction representation learning, we propose a three-element format that can be learnt end-to-end and explicitly reason some relational factors including the interaction strength and category. We apply GroupNet into both CVAE-based prediction system and previous state-of-the-art prediction systems for predicting socially plausible trajectories with relational reasoning. To validate the ability of relational reasoning, we experiment with synthetic physics simulations to reflect the ability to capture group behaviors, reason interaction strength and interaction category. To validate the effectiveness of prediction, we conduct extensive experiments on three real-world trajectory prediction datasets, including NBA, SDD and ETH-UCY; and we show that with GroupNet, the CVAE-based prediction system outperforms state-of-the-art methods. We also show that adding GroupNet will further improve the performance of previous state-of-the-art prediction systems.

1. Introduction

Multi-agent trajectory prediction aims to predict future trajectories of agents conditioned on past movements. This task is not only critical to numerous real-world applications, such as autonomous driving and human behavior understand-

---

*Corresponding author.

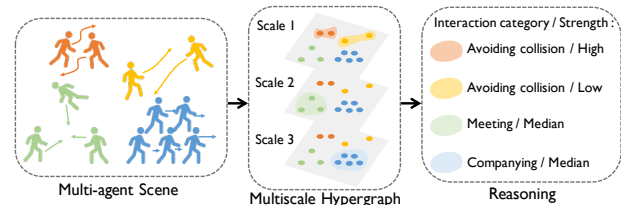Code is available at: https://github.com/MediaBrain-SJTU/GroupNet

Figure 1. GroupNet captures both pair-wise and group-wise interactions among multiple agents from their trajectories and infers the interaction category and strength for each interaction group.

ing, but also serves as a foundation to bridge the knowledge of the past and the action for the future. At least three factors would affect each agent's dynamics: ego momentum, instantaneous intent, and social influence from the other agents. The first factor has been well studied [36]; the second factor is unpredictable; and the third factor is an emerging research topic and the focus of this work. To demystify the social influence, we need to model and reason the interactions among agents based on their past spatio-temporal states, potentially leading to precise and interpretable trajectory prediction.

Many works have tried to model the interactions through social operations [1, 13, 43], graph-based modeling [14, 15, 18, 23] and attention mechanism [27, 33, 38]. Although these methods have greatly improved the empirical performances of trajectory prediction, they underestimate the significance of relational reasoning. Some previous works, such as NRI [17] and Evolvegraph [23], took a step forward towards explicit relational reasoning; however, they still have limitation from two aspects. First, only pair-wise interactions are modeled. In many scenarios, such as on a basketball court, a group of players execute a defensive strategy cooperatively; in the ocean, a school of fish coordinate their movements to evade predators, reflecting collective behaviors. While this kind of group-wise interaction is common, they have rarely been modeled. Second, only interaction categories are explicitly reasoned by the current methods, while the intensity level of the agent interaction cannot be reflected.

To further promote more comprehensive relational reasoning in trajectory prediction, this work puts efforts on two aspects: interaction capturing and interaction representation

learning. To capture more interactions, we propose a multi-scale hypergraph, which consists of a series of hypergraphs to model group-wise interactions with multiple group sizes. Instead of using handcrafted designs, we learn such a multiscale hypergraph topology in a data-driven manner. To learn the interaction embedding, we propose a three-element representation format: the neural interaction strength, the neural interaction category and the per-category function, which can reflect the interaction strength and category in an interactive group. Based on neural message passing over the multiscale hypergraph, we merge this three-element interaction embedding in the representation learning process.

Overall, we integrate these two designs and propose GroupNet, a multiscale hypergraph neural network, to capture social interactions for better trajectory prediction. The proposed GroupNet has two significant advantages: 1) *strong ability of relational reasoning*. Through extensive synthetic physics experiments, we validate that GroupNet is able to capture group behaviors, infer interaction strength and interaction category during trajectory prediction without any relation supervision; and 2) *strong ability of plug-and-play*. As a general and comprehensive social modeling module, GroupNet can be easily plugged in a prediction framework and improve its previous performance. For example, based on a straightforward CVAE-based framework, GroupNet significantly improves the prediction performance, leading to competitive or leading results on three real-world prediction benchmarks. Furthermore, for those current state-of-the-art prediction models, NMMP [14] on NBA dataset, PECNet [27] on SDD dataset and Trajectron++ [34] on ETH-UCY datasets, GroupNet brings significant benefit and lifts their results to even better performances on each dataset. The main contributions of this paper are summarized as follow:

• We propose GroupNet, a novel multiscale hypergraph neural network to model complex social influences for better trajectory prediction. GroupNet can capture both pair-wise and group-wise interactions at various group sizes and reflect interaction strength and interaction category.

• We validate the relational-reasoning ability of GroupNet via extensive synthetic simulations and show that GroupNet can capture group behaviors, interaction strength and interaction category in an unsupervised setting; see Section 6.2.

• With GroupNet, a simple CVAE-based framework achieves competitive performances on three real-world prediction benchmarks. We also show many state-of-the-art prediction systems can be significantly improved by replacing their social modules with GroupNet; see Section 6.3.

2. Related Works

Trajectory prediction. Traditional approaches use hand-crafted rules and energy potentials [2, 19, 28, 30, 40, 41]. For complex scenes, sequence-to-sequence models [1, 36] are leveraged to encode trajectories individually. Recent works model the interactions to predict more social-plausible trajectories. [3, 6–8, 46] use a spatial-centric mechanism to represent trajectories and consider the spatial interaction. [1, 13, 43] use a social mechanism to aggregate neighboring actors and broadcasts to each actor for neighboring information diffusion. Besides, attention mechanism [27, 33, 37, 38, 45] and transformer structure [11, 44] are used to capture agents' spatial and temporal dependencies. Graph-based methods [10, 14, 15, 18, 23, 24, 29, 42] are proposed to explicitly model the interaction between actors through non-grid structures. However, previous methods only focus on modeling the pair-wise interaction, ignoring the group behavior's influence on agents. In this work, we extend the graph-based mechanism from ordinary graphs to multiscale hypergraphs to capture group behaviors at different sizes, thereby modeling agent interactions more comprehensively.

Relational reasoning. In multi-agent systems, only the trajectories of individual agents are available without knowledge of the underlying relations, thus it is challenging to explicitly infer agents' interaction and make relational reasoning. Some works attempt to infer the interaction between agents. NRI [17] infers a latent interaction graph via an autoencoder model. RAIN [22] also infers an interaction graph using reinforcement learning to select edges. Evolve-Graph [23] and dNRI [12] consider a evolve mechanism and learn a dynamic interaction graph. Our work also tries on relational reasoning but the main differences are: i) we capture more interactions including both group-wise interaction and pair-wise interaction, comparing to only inferring pair-wise relationship on previous methods; and ii) we infer both interaction category and interaction strength while previous works only consider the interaction category.

3. Problem Formulation

The task of trajectory prediction with relational reasoning is to predict future trajectories of multiple agents and infer interaction relationship given past trajectories. Mathematically, let $\mathbb{X}^- \in \mathbb{R}^{N \times T_p \times 2}$ and $\mathbb{X}^+ \in \mathbb{R}^{N \times T_f \times 2}$ be the past and future trajectories of all the N agents in the scene. Let $\mathbf{X}_i^- = \mathbb{X}_{i,:,:}^- = [\mathbf{x}_i^{-T_p+1}, \mathbf{x}_i^{-T_p+2}, \cdots, \mathbf{x}_i^0] \in \mathbb{R}^{T_p \times 2}$ and $\mathbf{X}_i^+ = \mathbb{X}_{i,:,:}^+ = [\mathbf{x}_i^1, \mathbf{x}_i^2, \cdots, \mathbf{x}_i^{T_f}] \in \mathbb{R}^{T_f \times 2}$ be the past and future trajectories of the ith agent, where $\mathbf{x}_i^t \in \mathbb{R}^2$ is the 2D coordinate at time t. Our goal is to learn a prediction model $g(\cdot)$, so that the predicted future trajectories $\widehat{\mathbb{X}}^+ = g(\mathbb{X}^-)$ are as close to the ground-truth future trajectories $\mathbb{X}^+$ as possible. When a prediction model $g(\cdot)$ is capable to infer social influences among the agents without any ground-truth supervision about interaction category or strength, this supplementary task is called *unsupervised relational reasoning*.

4. GroupNet

The core of GroupNet is to learn a multiscale hypergraph whose node is the agent and hyperedge is the interaction;
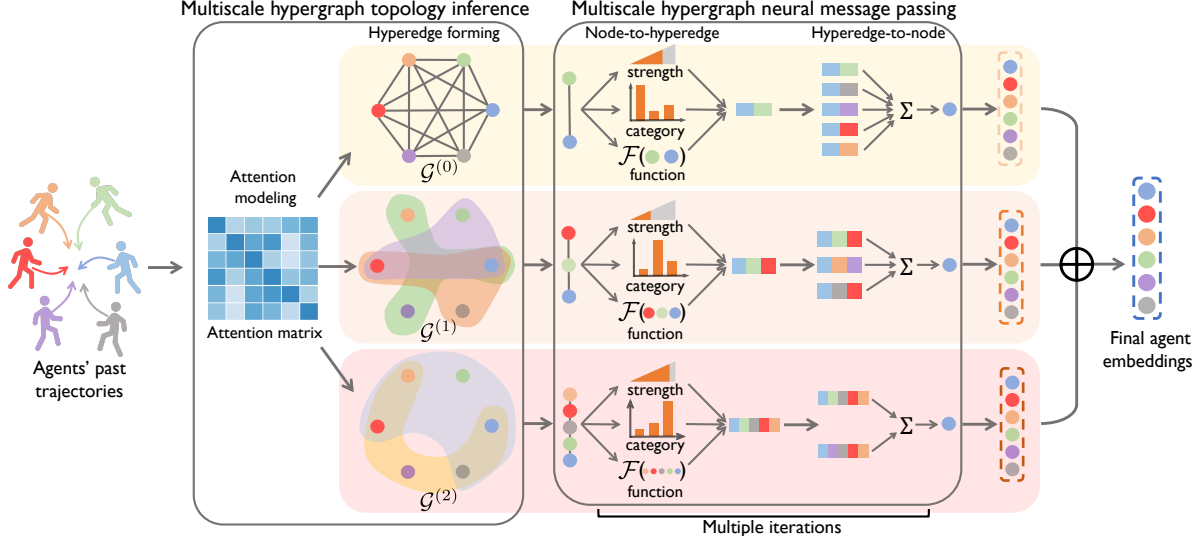
Figure 2. **The architecture of GroupNet.** GroupNet has two stages: multiscale hypergraph topology inference and multiscale hypergraph neural message passing, where we infer the hypergraph topologies and capture the agent patterns with their interaction, respectively. In multiscale hypergraph neural message passing, we model the interaction as three elements to reflect physical intuition and interpretation.

and then, leverage this multiscale hypergraph to learn agent and interaction embedding.

4.1. Multiscale hypergraph topology inference

To comprehensively model group-wise interactions at multiple scales, we consider inferring a multiscale hypergraph from agent dynamics to reflect interactions at various group sizes; see Figure 2. Mathematically, let $\mathcal{V} = \{v_1, v_2, \cdots, v_N\}$ be a set of agents and $\mathcal{G} = \{\mathcal{G}^{(0)}, \mathcal{G}^{(1)}, ...\mathcal{G}^{(S)}\}$ be a multiscale hypergraph showing the agent connections. At any scale s, $\mathcal{G}^{(s)} = (\mathcal{V}, \mathcal{E}^{(s)})$ has the hyperedge set $\mathcal{E}^{(s)} = \{e_1^{(s)}, e_2^{(s)}, \cdots, e_{M_s}^{(s)}\}$ to represent group-wise relations with M_s hyperedges, each of which links a number of agents to represent the common relations. A larger s indicates a larger scale of agent groups. Notably, $\mathcal{G}^{(0)} = (\mathcal{V}, \mathcal{E}^{(0)})$ is a specific hypergraph whose edges model the finest pair-wise agent connections. The topology of each $\mathcal{G}^{(s)}$ can be represented as an incidence matrix $\mathbf{H}^{(s)} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{E}^{(s)}|}$ where $\mathbf{H}_{i,j}^{(s)} = 1$ if the ith node is included in the jth hyperedge, otherwise $\mathbf{H}_{i,j}^{(s)} = 0$.

Affinity modeling. The trajectory interactions are in a stealth mode and nontrivial to capture since there are no explicitly predefined graph topology for relation description. To infer a multiscale hypergraph, we construct hyperedges by grouping agents that have highly correlated trajectories, whose correlations could be measured by mapping the trajectories as a high-dimensional feature vector. Concretely, the trajectory embedding of the ith agent is $\mathbf{q}_i = f_Q(\mathbf{X}_i^-) \in \mathbb{R}^d$, where $f_Q(\cdot)$ can be a MLP. We then compute an affinity matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ to reflect the correlation of any two agents. The (i, j)th element of $\mathbf{A}$ is:

$$\mathbf{A}_{i,j} = \mathbf{q}_i^\top \mathbf{q}_j / (\|\mathbf{q}_i\|_2 \|\mathbf{q}_j\|_2), \qquad (1)$$

which represents the relational weight of the ith agent and the jth agent reflecting the correlation between two agents. We normalize the embeddings to ensure the trajectory representations to carry unit energy for stable relation estimation.

Hyperedge forming. Given the affinity matrix, we then form hyperedges at various scales. At the 0th scale, we consider pair-wise connections. Each node connects the nodes that have the largest affinity scores with it, leading to M_0 edges in the incident matrix $\mathbf{H}^{(0)}$. For the other scales, we consider group-wise connections. Intuitively, agents in a group should have high correlation with each other. We thus find the groups of nodes by looking for high-density submatrices in the affinity matrix $\mathbf{A}$. We first assign a sequence of increasing group sizes $\{K^{(s)}\}_{s=1}^S$; and then, for each node, we find a highly correlated group at any scale s, leading to $K^{(s)}$ groups/hyperedges in each scale. Mathematically, let the ith hyperedge $e_i^{(s)}$ at the sth scale be the one associated with the ith node v_i. This hyperedge is obtained by solving the following optimization problem of searching a $K^{(s)} \times K^{(s)}$ submatrix:

$$e_i^{(s)} = \arg\max_{\Omega \subseteq \mathcal{V}} \|\mathbf{A}_{\Omega,\Omega}\|_{1,1}\,,$$
$$\text{s.t. } |\Omega| = K^{(s)}, \; v_i \in \Omega, \; i = 1, \ldots, N, \qquad (2)$$

where the entrywise matrix norm $\|\cdot\|_{1,1}$ denotes the sum of the absolute values of all elements. The objective finds the most correlated agents and links them together to consider the group behaviors. The first constraint limits the group size and the second requires the participance of the ith node. In this way, we could form at least one hyperedge that belongs to each node at each scale. To solve the optimization (2),

when the total number of agents is small, we could adopt an enumeration algorithm to search for the optimum solution; otherwise, we employ a greedy algorithm approximation that first selects v_i and then adds new nodes of maximum affinity value with v_i at each move sequentially.

The incidence matrices for a multiscale hypergraph are thus $\{\mathbf{H}^{(0)} \in \mathbb{R}^{N \times NK^{(0)}}, \{\mathbf{H}^{(s)} \in \mathbb{R}^{N \times N}\}_{s=1}^{S}\}$, where $\mathbf{H}^{(0)}$ includes $NK^{(0)}$ edges to consider the pair-wise interactions and $\mathbf{H}^{(s)}(s \geq 1)$ includes N hyperedges to reflect the group-wise interactions within $K^{(s)}$ agents. Different from the common multiscale graph whose node numbers vary from scales [9, 25], our multiscale hypergraph has fixed node numbers, yet different hyperedge sizes on various scales. Note that all the edges and hyperedges are selected based on the same affinity matrix. This design brings two benefits: i) it is computationally efficient to search for high-order relationships from a single matrix; and ii) it makes the training of the affinity matrix more stable and informative through back-propagation.

Compared to [14, 17, 23], our topology inference method is novel from two aspects. First, our method actively infers a graph structure through learning; while many previous methods directly adopt a fixed fully-connected topology. Second, our method models both pair-wise and group-wise connections at multiple scales; while the previous methods only model the pair-wise connections at a single scale.

4.2. Multiscale hypergraph neural message passing

To learn the patterns of agents trajectories given the inferred multiscale hypergraph, we customize a multiscale hypergraph neural message passing method obtaining the embeddings of agents and interactions iteratively through node-to-hyperedge and hyperedge-to-node, see Figure 2. Specifically, we first initialize the agent embedding from trajectory of each agent; that is, at any scale, for the ith agent, v_i, its initial embedding is $\mathbf{v}_i = \mathbf{q}_i \in \mathbb{R}^d$ (see Sec. 4.1). At each scale, in the node-to-hyperedge phase, groups of agent embeddings are aggregated to get the interaction embeddings. In the hyperedge-to-node phase, each agent embedding is updated according to associated interaction embeddings. We execute the node-to-hyperedge and hyperedge-to-node for several iterations. We finally obtain the representation of each agent by fusing its embeddings across all the scales.

Node-to-hyperedge phase. To promote a representation for relational reasoning, we exhibit an interaction embedding with three elements: *neural interaction strength*, representing the intensity of the interaction, *neural interaction category*, reflecting agents interaction category, and *per-category function*, modeling how interaction process of this category works. At each scale of hypergraph, for the ith hyperedge, e_i, its interaction embedding is obtained through

$$\mathbf{e}_i = r_i \sum_{\ell=1}^{L} c_{i,\ell} \mathcal{F}_\ell \Big(\sum_{v_j \in e_i} \mathbf{v}_j \Big) \in \mathbb{R}^d, \qquad (3)$$

where r_i is its neural interaction strength. $c_{i,\ell}$, the ℓth element of a category vector $\mathbf{c}_i \in [0,1]^L$, denotes the probability of the ℓth neural interaction category within L possible categories. For each category, we assign a learnable category function $\mathcal{F}_\ell(\cdot)$ implemented by MLPs. Each of these three elements is trainable in an end-to-end framework.

To obtain the neural interaction strength r_i, and neural interaction category $\mathbf{c}_i$, we leverage a collective embedding that is a hidden state to reflect the overall information of agents in a group. For hypergraph at any scale, the collective embedding of e_i is obtained by the weighted sum of all the agent embedding associated with the hyperedge e_i; that is, $\mathbf{z}_i = \sum_{v_j \in e_i} w_j \mathbf{v}_j$, where $w_j = \mathcal{F}_{\mathrm{w}}\Big(\mathbf{v}_j, \sum_{v_m \in e_i} \mathbf{v}_m \Big)$, with $\mathcal{F}_{\mathrm{w}}(\cdot)$ implemented by an MLP. The weight w_j reflects the contribution of the jth node to the ith group. We then use the collective embedding to infer the neural interaction strength, r_i, and the neural interaction category, $\mathbf{c}_i$; that is,

$$r_i = \sigma\left(\mathcal{F}_{\mathrm{r}}(\mathbf{z}_i)\right), \quad \mathbf{c}_i = \mathrm{softmax}\left((\mathcal{F}_{\mathrm{c}}(\mathbf{z}_i) + \mathbf{g})/\tau\right),$$

where $\sigma(\cdot)$ is a sigmoid function to constrain the strength values, $\mathbf{g}$ is a vector whose elements are i.i.d. sampled from $\mathrm{Gumbel}(0,1)$ distribution and τ is the temperature controlling the smoothness of type distribution. We use the Gumbel softmax to make a continuous approximation of the discrete distribution following [26]. $\mathcal{F}_{\mathrm{r}}(\cdot)$ and $\mathcal{F}_{\mathrm{c}}(\cdot)$ are modeled by MLPs. The inference of the neural interaction strength, neural interaction category and per-category function considers the group behavior through the shared collective embedding.

Hyperedge-to-node phase. Given the interaction embeddings learnt from groups of agents, we then update each agent embedding by considering all the associated interactions. For any scale of hypergraph, let $\mathcal{E}_i = \{e_j | v_i \in e_j\}$ be the set of hyperedges associated with the ith node v_i. The embedding of the ith agent in the next iteration is updated as

$$\mathbf{v}_i \leftarrow f_{\mathrm{v}}\big(\big[\mathbf{v}_i, \sum_{e_j \in \mathcal{E}_i} \mathbf{e}_j\big]\big) \in \mathbb{R}^d,$$

where $f_{\mathrm{v}}(\cdot)$ is a trainable MLP; $[\cdot, \cdot]$ denotes the embedding concatenation of one node and the associated hyperedges to absorb the influence from interactions to an agent.

At any scale s, we repeat the node-to-hyperedge and hyperedge-to-node phases for several times and obtain the embedding of the ith node, $\mathbf{v}_i^{(s)}$. In this way, we compute the embeddings of the ith node, $\mathbf{v}_i^{(0)}, \mathbf{v}_i^{(1)} \ldots, \mathbf{v}_i^{(S)}$, across S scales in parallel. Note that the trainable layers in various scales are NOT shared. We finally concatenate the agent embedding across all the scales to obtain the comprehensive agents' embedding, formulated as $\mathbf{v}_i = [\mathbf{v}_i^{(0)}, \mathbf{v}_i^{(1)}, \cdots, \mathbf{v}_i^{(S)}] \in \mathbb{R}^{d(S+1)}$.

Compared to previous methods NRI [17], dNRI [12], and EvolveGraph [23], which achieve relational reasoning using neural message passing, the proposed hypergraph neural message passing is novel from three aspects. First, previ-
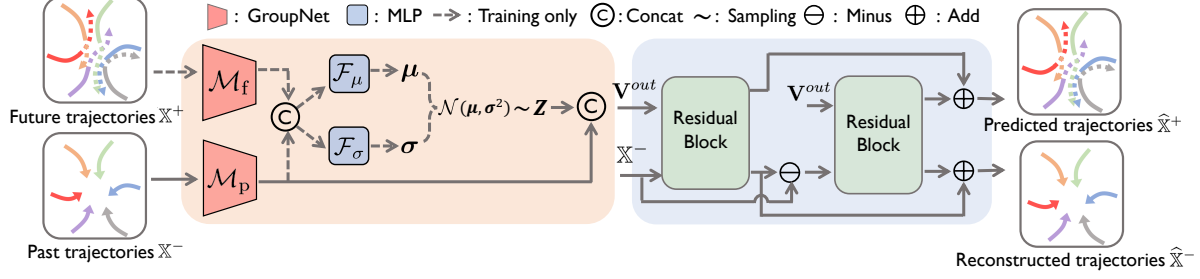
Figure 3. The architecture of the trajectory prediction system. The system consists of two processes: the encoding process (orange part) and the decoding process (blue part). Encoding process obtains the hidden representation for each agent. Decoding process aims to both predict the future trajectory and reconstruct the past trajectory given the hidden representation and past trajectory of each agent.

ous [12, 17, 23] adopt neural message passing on ordinary graphs; while we consider neural message passing on a series of hypergraphs, promoting more comprehensive information propagation and aggregation. Second, previous [12, 17, 23] only infer the interaction category; while our method can infer both interaction category and strength through a three-element representation format. Third, the interaction category inferred by previous [12, 17, 23] is the final output after message passing; while the interaction category and strength inferred by our method are the intermediate features that involve in neural message passing. Our design not only promotes relational reasoning in neural message passing, but also improves the prediction performance.

5. Prediction System with GroupNet

We apply GroupNet into a basic multi-agent trajectory prediction system based on the CVAE framework [35] to handle the stochasticity of each agent's behavior. Let $\log p(\mathbb{X}^+ \mid \mathbb{X}^-)$ be the log-likelihood of future trajectories $\mathbb{X}^+$ conditioned on the past trajectories $\mathbb{X}^-$. The corresponding evidence lower bound (ELBO) is:

$$\log p(\mathbb{X}^+ \mid \mathbb{X}^-) \geq \mathbb{E}_{q(\mathbf{Z}\mid\mathbb{X}^+,\mathbb{X}^-)} \log p(\mathbb{X}^+ \mid \mathbf{Z}, \mathbb{X}^-)$$
$$- \mathrm{KL}(q(\mathbf{Z} \mid \mathbb{X}^+, \mathbb{X}^-)\|p(\mathbf{Z} \mid \mathbb{X}^-)),$$

where $\mathbf{Z} \in \mathbb{R}^{N \times d_z}$ is the latent codes corresponding to all the agents; $p(\mathbf{Z}|\mathbb{X}^-)$ is the conditional prior of $\mathbf{Z}$, which is set to be a Gaussian distribution. Here we implement $q(\mathbf{Z}|\mathbb{X}^+, \mathbb{X}^-)$ by an encoding process for embedding learning, and implement $p(\mathbb{X}^+|\mathbf{Z}, \mathbb{X}^-)$ by a decoding process that predicts the future $\mathbb{X}^+$. The system is sketched in Figure 3.

Encoding process. The encoding process aims to generate the Gaussian parameters of the approximate posterior distribution. Mathematically, the Gaussian parameters generation of the approximate posterior distribution and the conditional prior distribution is given by:

$$\mathbf{V}^+ = \mathcal{M}_\mathrm{f}(\mathbb{X}^+), \quad \mathbf{V}^- = \mathcal{M}_\mathrm{p}(\mathbb{X}^-),$$
$$\boldsymbol{\mu}_q = \mathcal{F}_\mu([\mathbf{V}^+, \mathbf{V}^-]), \quad \boldsymbol{\sigma}_q = \mathcal{F}_\sigma([\mathbf{V}^+, \mathbf{V}^-]),$$

where $\mathcal{M}_\mathrm{f}(\cdot)$ and $\mathcal{M}_\mathrm{p}(\cdot)$ is aforementioned GroupNet to get future agent embedding $\mathbf{V}^+$ and past agent embedding $\mathbf{V}^-$. $\boldsymbol{\mu}_q, \boldsymbol{\sigma}_q$ are the mean and variation of the approximate posterior distribution. $\mathcal{F}_\mu(\cdot), \mathcal{F}_\sigma(\cdot)$ are MLPs. We sample latent

code of possible future trajectories $\mathbf{Z} \sim \mathcal{N}(\boldsymbol{\mu}_q, \mathrm{Diag}(\boldsymbol{\sigma}_q^2))$. In the testing phase, we sample $\mathbf{Z}$ from the prior distribution $\mathcal{N}(0, \lambda\mathbf{I})$, where λ is a hyperparameter. We concatenate the latent code $\mathbf{Z}$ with the embedding of agent past trajectories $\mathbf{V}^-$ as the output of encoding process: $\mathbf{V}^\mathrm{out} = [\mathbf{Z}, \mathbf{V}^-]$.

Decoding process. We apply a residual decoder similar with [5] which takes both the output embedding of the encoding process and past trajectories as input. The residual decoder aims to both predict the future trajectories and reconstruct the past trajectories, which avoids input information loss. The residual decoder consists of two decoding blocks with the same structure. Mathematically, the calculation of the two blocks $\mathcal{F}_{Block1}(\cdot), \mathcal{F}_{Block2}(\cdot)$ are:

$$\widehat{\mathbb{X}}_1^+, \widehat{\mathbb{X}}_1^- = \mathcal{F}_{Block1}(\mathbf{V}^{out}, \mathbb{X}^-),$$
$$\widehat{\mathbb{X}}_2^+, \widehat{\mathbb{X}}_2^- = \mathcal{F}_{Block2}(\mathbf{V}^{out}, \mathbb{X}^- - \widehat{\mathbb{X}}_1^-),$$

where $\widehat{\mathbb{X}}_1^+, \widehat{\mathbb{X}}_2^+$ and $\widehat{\mathbb{X}}_1^-, \widehat{\mathbb{X}}_2^-$ are predicted future trajectories and reconstructed past trajectories of two blocks. Each block consists of a GRU encoder to encode the sequence and two MLPs as the output header. The final prediction of future trajectories $\widehat{\mathbb{X}}^+$ and reconstruction of past trajectories $\widehat{\mathbb{X}}^-$ is the sum of two decoding blocks:

$$\widehat{\mathbb{X}}^+ = \widehat{\mathbb{X}}_1^+ + \widehat{\mathbb{X}}_2^+, \quad \widehat{\mathbb{X}}^- = \widehat{\mathbb{X}}_1^- + \widehat{\mathbb{X}}_2^-.$$

Loss function. To train the prediction system, we minimize an overall loss function formed by an evidence lower bound loss, a past trajectory reconstruction loss and a variety loss in Social-GAN [13] to optimize the "best" prediction:

$$\mathcal{L}_\mathrm{elbo} = \alpha\|\widehat{\mathbb{X}}^+ - \mathbb{X}^+\|_2^2 + \beta\,\mathrm{KL}(\mathcal{N}(\boldsymbol{\mu}_q, \mathrm{Diag}(\boldsymbol{\sigma}_q^2))\|\mathcal{N}(0, \lambda\mathbf{I})),$$
$$\mathcal{L}_\mathrm{rec} = \gamma\|\widehat{\mathbb{X}}^- - \mathbb{X}^-\|_2^2, \mathcal{L}_\mathrm{variety} = \min_k \|\widehat{\mathbb{X}}^{+(k)} - \mathbb{X}^+\|_2^2,$$
$$\mathcal{L} = \mathcal{L}_\mathrm{elbo} + \mathcal{L}_\mathrm{rec} + \mathcal{L}_\mathrm{variety},$$

where α, β and γ are hyperparameters to control the weights, $\|\cdot\|_2$ denotes the ℓ_2 norm and $\mathrm{KL}(\cdot\|\cdot)$ denotes the KL divergence. In the variety loss, we predict multiple times and $\widehat{\mathbb{X}}^{+(k)}$ denotes kth prediction results.

6. Experiments and Analysis

6.1. Experimental setup

Datasets. We evaluate our method on both physical simulation datasets and three real-world public datasets: the NBA
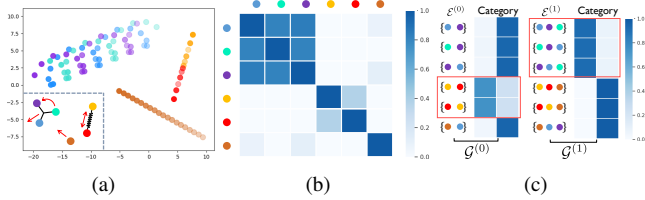
Figure 4. Visualization of learnt group behavior. (a) The particle trajectories containing a three-group with a light bar, a two-group with a spring and an individual particle. (b) The heatmap of the learnt affinity matrix via affinity modeling. (c) The multiscale hypergraph topology via hyperedge forming and the interaction category vector of each hyperedge. The red box represents the three-group and two-group we inferred.

Figure 5. Trajectory samples under three interaction categories.

Table 1. Comparison of accuracy (mean $\pm$ std in %) of on interaction category recognition.

	Corr.(path)	Corr.(LSTM)	NRI	**Ours**	Supervised
2-types	78.3$\pm$0.0	75.3$\pm$2.5	89.7$\pm$3.5	**99.3$\pm$0.3**	99.5$\pm$0.2
3-types	–	–	64.9$\pm$2.3	**90.3$\pm$0.9**	98.7$\pm$0.7

SportVU Dataset (NBA), the ETH-UCY Dataset [21, 31], and the Stanford Drone Dataset (SDD) [32]. For the physical simulation datasets, since the ground truth of the interaction relationship can be accessed, we evaluate the ability of relational reasoning of our GroupNet. For the real-world datasets we evaluate the effectiveness of our proposed GroupNet. More details of datasets are introduced in Appendix.

Metrics. Following [1, 20, 23], we use $\min ADE_K$ and $\min FDE_K$ as evaluation metrics. $\min ADE_K$ is the minimum among K average distances of the K predicted trajectories to the ground-truths in terms of the whole trajectories; $\min FDE_K$ shows the minimum distance among K predicted endpoints to the ground-truth endpoints.

Implementation details. In our model, all the MLPs have 3 layers with the ReLU activation function. The latent code dimension d_z is 32. In the loss function, we set α, β, γ to 1.0 and the variance of the prior distribution $\lambda = 1.0$. We consider all other agent at the 0th scale. For NBA dataset, we set the scale to 2,5,11 players. For ETH dataset, we set the group scales to the half and the full of agents number and for SDD dataset to full of agent number in less than 100 pixels distance. To train the model, we apply Adam optimizer [16] with learning rate 10^{-4} and decay every 10 epochs. See more details in Appendix.

6.2. Validation on relational reasoning

Ability to capture group behaviors. We consider six particles forming two scales of groups: the Y-shape light bar and the spring; see Figure 4 (a) for an example. During the prediction process, GroupNet aims to find the group belongingness. Figure 4 (b) shows the learnt affinity matrix in (1), where a darker color denotes higher affinity, which is revealed between the particles connected by the light bar or spring. Figure 4 (c) illustrates the inferred (hyper)edges and their interaction categories (the category number is set to be 2). We see that i) the edge of spring and the hyperedge of light bar are inferred on our multiscale hypergraph $\{\mathcal{G}^{(0)}, \mathcal{G}^{(1)}\}$. ii) the spring and light bar have different interaction categories from other hyperedges, indicating that we effectively capture these two group interactions.
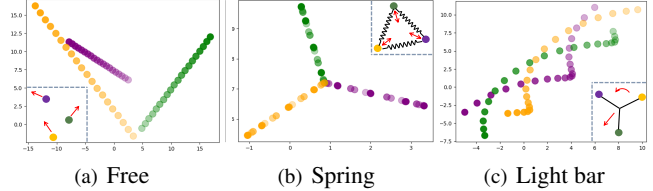
Ability to reason interaction category. We consider three particles forming one group with three possible interaction categories: free, spring and light bar. Figure 5 gives examples of each interaction type. We compare the neural interaction category in (3) with the ground-truth category to achieve interaction recognition and report the recognition accuracy. This category recognition is unsupervised because the model only accesses the trajectories of particles without any prior information about the three types of interaction. We compare our method with three baselines [17] which consider only pair-wise interactions and an upper bound. To adapt them to group-wise actions, we use a majority vote for all pair-wise edges. The upper bound 'Supervised' represents using the ground-truth category to train the neural interaction category. Table 1 reports the recognition accuracy on 2-type (only free and light bar type) and 3-type recognition tasks, which are averaged over 5 independent runs. '–' represents not applicable. We see that i) our method significantly outperforms three baselines, as we model the group interaction while NRI only considers pair-wise interaction; and ii) our method achieves similar performance to the upper bound, reflecting that our method is capable to reason the interaction category in an unsupervised manner.

Ability to reason interaction strength. We consider the movement of two charged particles which interact via Coulomb forces. Figure 6 shows two examples of particle trajectories. A stronger repulsion the moving particle receives leading to a higher interaction strength. During the prediction we aim to validate whether the neural interaction strength in (3) can reflect the amount of charge without any direct supervision. Figure 6 (c) shows the relations between the learnt interaction strength (x-axis) and charge on the fixed particle (y-axis). We see that the neural interaction strength has a proportional relationship with the amount of charge, reflecting our model is capable to implicitly capture the interaction strength in an unsupervised manner.

6.3. Validation on effectiveness

NBA dataset. We predict the future 10 timestamps (4.0s) based on the historical 5 timestamps (2.0s). Table

Table 2. minADE$_{20}$ / minFDE$_{20}$ (meters) of trajectory prediction (NBA dataset). ∗ denotes that NMMP is the previous best model on this dataset. **With GroupNet, NMMP works even much better. Simple CVAE with GroupNet achieves the current best result.**

Time	Social -LSTM [1]	Social -GAN [13]	Social- STGCNN [29]	STGAT [15]	NRI [17]	STAR [44]	PECNet [27]	NMMP* [14]	GroupNet +NMMP	CVAE	GroupNet +CVAE
1.0s	0.45/0.67	0.46/0.65	0.36/0.50	0.38/0.55	0.45/0.64	0.43/0.65	0.51/0.76	0.38/0.54	0.38/0.54	0.37/0.52	**0.34/0.48**
2.0s	0.88/1.53	0.85/1.36	0.75/0.99	0.73/1.18	0.84/1.44	0.77/1.28	0.96/1.69	0.70/1.11	0.69/1.08	0.67/1.06	**0.62/0.95**
3.0s	1.33/2.38	1.24/1.98	1.15/1.79	1.07/1.74	1.24/2.18	1.00/1.55	1.41/2.52	1.01/1.61	0.98/1.47	0.96/1.51	**0.87/1.31**
4.0s	1.79/3.16	1.62/2.51	1.59/2.37	1.41/2.22	1.62/2.84	1.26/2.04	1.83/3.41	1.33/2.05	1.25/1.80	1.25/1.96	**1.13/1.69**

Table 3. minADE$_{20}$ / minFDE$_{20}$ (pixels) of trajectory prediction (SDD dataset). ∗ denotes that PECNet is the previous best model on this dataset. **With GroupNet, PECNet works even better. Simple CVAE with GroupNet achieves the current best ADE result.**

Time	Social- LSTM [1]	Social -GAN [13]	SOPHIE [33]	Trajectron++ [34]	NMMP [14]	EvolveGraph [23]	CF-VAE [4]	PECNet* [27]	GroupNet +PECNet	CVAE	GroupNet +CVAE
4.8s	31.19/56.97	27.23/41.44	16.27/29.38	19.30/32.70	14.67/26.72	13.90/22.90	12.60/22.30	9.96/15.88	9.65/**15.34**	10.22/18.18	**9.31**/16.11

Table 4. minADE$_{20}$ / minFDE$_{20}$ (meters) of trajectory prediction (ETH-UCY dataset). ∗ denotes that Trajectron++ is the previous best model on this dataset. **With GroupNet, Trajectron++ works even better. Simple CVAE with GroupNet achieves comparable results.**

Subset	Social- LSTM [1]	Social- Attention [38]	Social- GAN [13]	SOPHIE [33]	STGAT [15]	NMMP [14]	STAR [44]	PECNet [27]	Trajectron++* [34]	GroupNet +Trajectron++	CVAE	GroupNet +CVAE
ETH	1.09/2.35	1.39/2.39	0.87/1.62	0.70/1.43	0.65/1.12	0.61/1.08	**0.36/0.65**	0.54/0.87	0.39/0.83	0.38/0.74	0.47/0.80	0.46/0.73
HOTEL	0.79/1.76	2.51/2.91	0.67/1.37	0.76/1.67	0.35/0.66	0.33/0.63	0.17/0.36	0.18/0.24	0.12/0.21	**0.11/0.20**	0.17/0.31	0.15/0.25
UNIV	0.67/1.40	1.25/2.54	0.76/1.52	0.54/1.24	0.52/1.10	0.52/1.11	0.31/0.62	0.35/0.60	0.20/0.44	**0.19/0.40**	0.28/0.52	0.26/0.49
ZARA1	0.47/1.00	1.01/2.17	0.35/0.68	0.30/0.63	0.34/0.69	0.32/0.66	0.26/0.55	0.22/0.39	0.15/0.33	**0.14/0.32**	0.25/0.49	0.21/0.39
ZARA2	0.56/1.17	0.88/1.75	0.42/0.84	0.38/0.78	0.29/0.60	0.43/0.85	0.22/0.46	0.17/0.30	**0.11/0.25**	**0.11/0.25**	0.21/0.40	0.17/0.33
AVG	0.72/1.54	1.41/2.35	0.61/1.21	0.54/1.15	0.43/0.83	0.41/0.82	0.26/0.53	0.29/0.48	**0.19**/0.41	**0.19/0.38**	0.28/0.50	0.25/0.44

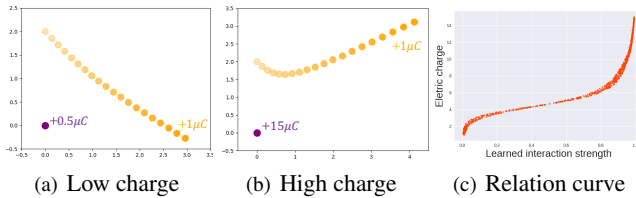(a) Low charge (b) High charge (c) Relation curve

Figure 6. Particles' trajectories and the curve of neural interaction strength with particle's electric charge.

2 shows the comparison with nine state-of-the-art methods. We see that i) GroupNet with CVAE framework achieves a significant improvement over the state-of-the-art methods: the minADE$_{20}$ and minFDE$_{20}$ at 4.0s is reduced by 15.0%/17.6% comparing to the best baseline method (NMMP). The improvement over previous methods increases with timestamps, since the proposed multi-scale hypergraph can capture more comprehensive interaction patterns and ii) We add GroupNet into previous the state-of-the-art method and replace its encoder (Group-Net+NMMP). Our GroupNet improve its performance by reducing minADE$_{20}$/minFDE$_{20}$ at 4.0s by 6.0%/12.2%.

SDD dataset. Table 3 compares the proposed method with nine state-of-the-art methods. We see that i) GroupNet with CVAE framework achieves state-of-the-art performance and reduces the minADE$_{20}$ by 6.5% comparing to PECNet. ii) We add GroupNet into the previous best framework and replace its Social Pooling encoder (GroupNet+PECNet). Our GroupNet is capable to improve its performance by reducing minADE$_{20}$/minFDE$_{20}$ by 3.1%/3.4%.

ETH-UCY dataset. Table 4 compares the proposed method with nine state-of-the-art methods and the 'AVG' means the averaged result over 5 subsets. We see that i)

GroupNet with CVAE framework outperforms most of previous methods and achieves closing performance with the state-of-the-art method (Trajectron++); and ii) We add GroupNet into previous the state-of-the-art method and replace its encoder (GroupNet+Trajectron++). Our GroupNet is capable to improve its performance by reducing minFDE$_{20}$ by 7.3% and reaches a new state-of-the-art performance.

Discussion. We see that previously, NMMP, PECNet and Trajctron++ are the best models on NBA, SDD and ETH-UCY datasets, respectively, and no single method achieves the best performances across all three datasets. *Here we analyze that this phenomenon may result from the inherence of the datasets.* In SDD, there are relatively smooth trajectories and simple interactions among agents. Hence, once a specific destination is given, the predictions tend to be straightforward. This enables PECNet that emphasize accurate destinations prediction to achieve good results. Opposite to SDD, the NBA dataset entails much more complex agent interactions. Methods that pay more attention to interaction modeling would suit NBA setting well. For example, NMMP and STAR achieve leading performances on NBA. Lastly, ETH-UCY contains five subsets, each of which varies from each other. This matches Trajectron++ that incorporates multi-modal decoders to promote generalization.

Under this circumstance, our GroupNet shows commendable consistency: across all the datasets and methods, *once GroupNet is plugged in to a prediction system, the performance can be significantly and consistently lifted up.* This reflects the robustness and generalization of our method.

Qualitative results. Figure 7 compares the predicted trajectories of NMMP, GroupNet with the CVAE framework (Ours) and ground-truth (GT) trajectories on NBA dataset.

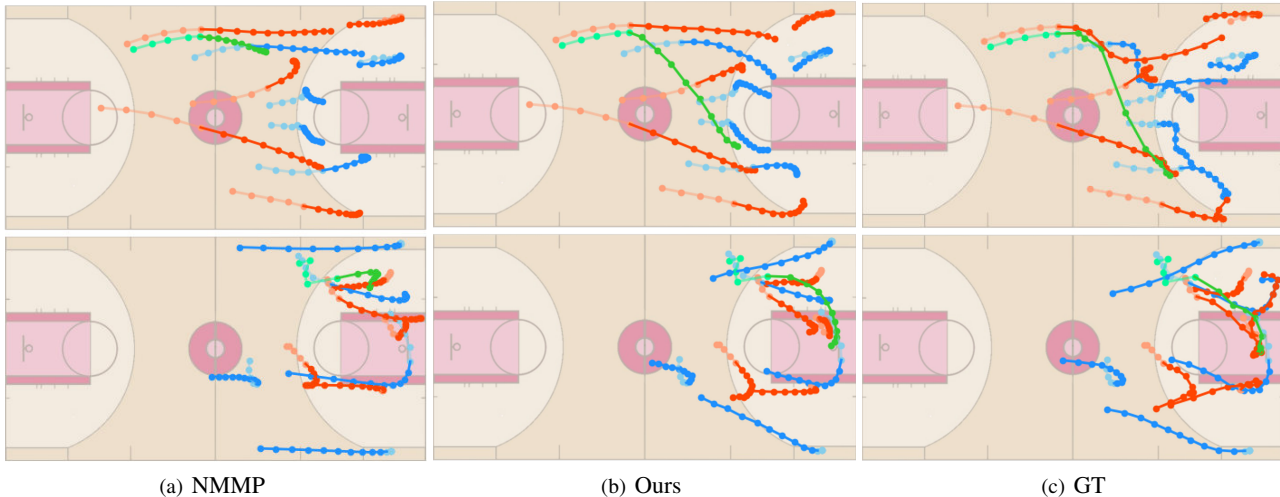|		
(a) NMMP	(b) Ours	(c) GT

Figure 7. Qualitative results on the NBA dataset. We plot the best trajectory among 20 predictions for the state-of-the-art method (NMMP), GroupNet with the CVAE framework (Ours) and ground truth (GT). The red/blue color represents players of two teams and the green color represents the basketball. Light color represents the past trajectory.

Table 5. Ablation studies of different feature extraction modules on the NBA dataset. We report minADE$_{20}$ / minFDE$_{20}$ (meters).

Module	Prediction time			
$\mathcal{M}_p$ & $\mathcal{M}_f$	1.0s	2.0s	3.0s	4.0s
MLP	0.37/0.52	0.67/1.06	0.96/1.51	1.25/1.96
NMMP	0.35/0.49	0.63/0.97	0.90/1.36	1.17/1.76
ATT	0.35/0.50	0.63/0.97	0.90/1.35	1.16/1.74
GroupNet	**0.34/0.48**	**0.62/0.95**	**0.87/1.31**	**1.13/1.69**

Table 6. minADE$_{20}$ / minFDE$_{20}$ (meters) of different group scales on NBA dataset.

	Prediction time			
Scale	1.0s	2.0s	3.0s	4.0s
1	0.37/0.52	0.67/1.05	0.97/1.50	1.27/1.95
1,2	0.35/0.49	0.63/0.97	0.89/1.35	1.16/1.75
1,2,5	0.35/0.49	0.62/0.95	0.88/1.32	1.14/1.71
1,2,5,11	0.34/0.48	0.62/0.95	0.87/1.31	1.13/1.69
1,2,3,5,11	0.34/0.48	0.62/0.94	0.87/1.31	1.13/1.70

We see that i) our method produces more precise predictions than previous state-of-the art method NMMP; and ii) for the scenes of fierce confrontation between two teams (the second row), our method has a larger improvement. This is because the proposed method captures more comprehensive interactions among players in complicated scenes.

6.4. Ablation studies

Effects of GroupNet in the system. We perform extensive ablation studies on the NBA dataset to investigate the contribution of the key technical component GroupNet;see Table 5. We apply three other feature extraction modules. The 'MLP' means that we use a multi-layer perceptron. The 'ATT' amd 'NMMP' means that using the non-local attention mechanism in [39] and using the graph neural message passing in [14] which models the pair-wise interactions between two agents. We see that the proposed GroupNet leads to su-

perior performance comparing to the multi-layer perceptron, the attention mechanism and the graph-based mechanism.

Effects of multiple scales. Table 6 presents the effect of multiple scales in the GroupNet on the CVAE framework on the NBA dataset. We see that i) The performance increases at first when choosing more scales. ii) The performance becomes stable when having sufficient scales.

7. Conclusion

This paper proposes GroupNet, which promotes a more comprehensive interaction modeling from two aspects: i) designing a multiscale hypergraph to capture both pairwise and group-wise interactions; and ii) designing a three-element representation format of interaction embedding, including neural interaction strength, category and per-category function. We apply the proposed GroupNet as a key component to a CVAE-based prediction system and previous systems. Experiments on synthetic physics simulations and real-world datasets reveal the ability of relational reasoning and the effectiveness of GoupNet.

Limitation and future work. In this work, the agent number is relatively small and the time window is relatively short due to the attribute of datasets. In the future, we will extend the network into more complex interacting systems and explore the long-term prediction task containing a time-varying number of agents with an adaptive GNN.

Acknowledgements

References

[1] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016. 1, 2, 6, 7

[2] Gianluca Antonini, Michel Bierlaire, and Mats Weber. Discrete choice models of pedestrian walking behavior. *Transportation Research Part B: Methodological*, 40(8):667–687, 2006. 2

[3] Mayank Bansal, Alex Krizhevsky, and Abhijit Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *arXiv preprint arXiv:1812.03079*, 2018. 2

[4] Apratim Bhattacharyya, Michael Hanselmann, Mario Fritz, Bernt Schiele, and Christoph-Nikolas Straehle. Conditional flow variational autoencoders for structured sequence prediction. *arXiv preprint arXiv:1908.09008*, 2019. 7

[5] Defu Cao, Yujing Wang, Juanyong Duan, Ce Zhang, Xia Zhu, Conguri Huang, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, et al. Spectral temporal graph neural network for multivariate time-series forecasting. *arXiv preprint arXiv:2103.07719*, 2021. 5

[6] Sergio Casas, Wenjie Luo, and Raquel Urtasun. Intentnet: Learning to predict intention from raw sensor data. In *Conference on Robot Learning*, pages 947–956. PMLR, 2018. 2

[7] Fang-Chieh Chou, Tsung-Han Lin, Henggang Cui, Vladan Radosavljevic, Thi Nguyen, Tzu-Kuo Huang, Matthew Niedoba, Jeff Schneider, and Nemanja Djuric. Predicting motion of vulnerable road users using high-definition maps and efficient convnets. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1655–1662. IEEE, 2018. 2

[8] Nachiket Deo and Mohan M Trivedi. Convolutional social pooling for vehicle trajectory prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1468–1476, 2018. 2

[9] Hongyang Gao and Shuiwang Ji. Graph u-nets. In *international conference on machine learning*, pages 2083–2092. PMLR, 2019. 4

[10] Jiyang Gao, Chen Sun, Hang Zhao, Yi Shen, Dragomir Anguelov, Congcong Li, and Cordelia Schmid. Vectornet: Encoding hd maps and agent dynamics from vectorized representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11525–11533, 2020. 2

[11] Francesco Giuliari, Irtiza Hasan, Marco Cristani, and Fabio Galasso. Transformer networks for trajectory forecasting. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 10335–10342. IEEE, 2021. 2

[12] Colin Graber and Alexander G Schwing. Dynamic neural relational inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8513–8522, 2020. 2, 4, 5

[13] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the*

IEEE Conference on Computer Vision and Pattern Recognition*, pages 2255–2264, 2018. 1, 2, 5, 7

[14] Yue Hu, Siheng Chen, Ya Zhang, and Xiao Gu. Collaborative motion prediction via neural motion message passing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6319–6328, 2020. 1, 2, 4, 7, 8

[15] Yingfan Huang, Huikun Bi, Zhaoxin Li, Tianlu Mao, and Zhaoqi Wang. Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6272–6281, 2019. 1, 2, 7

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[17] Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *International Conference on Machine Learning*, pages 2688–2697. PMLR, 2018. 1, 2, 4, 5, 6, 7

[18] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, S Hamid Rezatofighi, and Silvio Savarese. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. *arXiv preprint arXiv:1907.03395*, 2019. 1, 2

[19] Jae-Gil Lee, Jiawei Han, and Kyu-Young Whang. Trajectory clustering: a partition-and-group framework. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pages 593–604, 2007. 2

[20] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B Choy, Philip HS Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 336–345, 2017. 6

[21] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. In *Computer graphics forum*, volume 26, pages 655–664. Wiley Online Library, 2007. 6

[22] Jiachen Li, Fan Yang, Hengbo Ma, Srikanth Malla, Masayoshi Tomizuka, and Chiho Choi. Rain: Reinforced hybrid attention inference network for motion forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16096–16106, 2021. 2

[23] Jiachen Li, Fan Yang, Masayoshi Tomizuka, and Chiho Choi. Evolvegraph: Multi-agent trajectory prediction with dynamic relational reasoning. *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2020. 1, 2, 4, 5, 6, 7

[24] Maosen Li, Siheng Chen, Yanning Shen, Genjia Liu, Ivor W Tsang, and Ya Zhang. Online multi-agent forecasting with interpretable collaborative graph neural network. *arXiv preprint arXiv:2107.00894*, 2021. 2

[25] Maosen Li, Siheng Chen, Yangheng Zhao, Ya Zhang, Yanfeng Wang, and Qi Tian. Dynamic multiscale graph neural networks for 3d skeleton based human motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 214–223, 2020. 4

[26] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016. 4

[27] Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *European Conference on Computer Vision*, pages 759–776. Springer, 2020. 1, 2, 7

[28] Ramin Mehran, Alexis Oyama, and Mubarak Shah. Abnormal crowd behavior detection using social force model. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 935–942. IEEE, 2009. 2

[29] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. 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*, pages 14424–14432, 2020. 2, 7

[30] Brendan Morris and Mohan Trivedi. Learning trajectory patterns by clustering: Experimental studies and comparative evaluation. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 312–319. IEEE, 2009. 2

[31] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision*, pages 261–268. IEEE, 2009. 6

[32] Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *European conference on computer vision*, pages 549–565. Springer, 2016. 6

[33] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1349–1358, 2019. 1, 2, 7

[34] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Multi-agent generative trajectory forecasting with heterogeneous data for control. 2020. 2, 7

[35] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28:3483–3491, 2015. 5

[36] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *arXiv preprint arXiv:1409.3215*, 2014. 1, 2

[37] Bohan Tang, Yiqi Zhong, Ulrich Neumann, Gang Wang, Ya Zhang, and Siheng Chen. Collaborative uncertainty in multi-agent trajectory forecasting. *Advances in Neural Information Processing Systems*, 34, 2021. 2

[38] Anirudh Vemula, Katharina Muelling, and Jean Oh. Social attention: Modeling attention in human crowds. In *2018 IEEE international Conference on Robotics and Automation (ICRA)*, pages 4601–4607. IEEE, 2018. 1, 2, 7

[39] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018. 8

[40] Xiaogang Wang, Keng Teck Ma, Gee-Wah Ng, and W Eric L Grimson. Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models. *International journal of computer vision*, 95(3):287–312, 2011. 2

[41] Xiaogang Wang, Xiaoxu Ma, and W Eric L Grimson. Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. *IEEE Transactions on pattern analysis and machine intelligence*, 31(3):539–555, 2008. 2

[42] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 753–763, 2020. 2

[43] Yanyu Xu, Zhixin Piao, and Shenghua Gao. Encoding crowd interaction with deep neural network for pedestrian trajectory prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5275–5284, 2018. 1, 2

[44] Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *European Conference on Computer Vision*, pages 507–523. Springer, 2020. 2, 7

[45] Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris M Kitani. Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9813–9823, 2021. 2

[46] Tianyang Zhao, Yifei Xu, Mathew Monfort, Wongun Choi, Chris Baker, Yibiao Zhao, Yizhou Wang, and Ying Nian Wu. Multi-agent tensor fusion for contextual trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12126–12134, 2019. 2