

Social NCE: Contrastive Learning of Socially-aware Motion Representations

Yuejiang Liu, Qi Yan, Alexandre Alahi
 École Polytechnique Fédérale de Lausanne (EPFL)
 {firstname.lastname}@epfl.ch

Abstract

Learning socially-aware motion representations is at the core of recent advances in multi-agent problems, such as human motion forecasting and robot navigation in crowds. Despite promising progress, existing representations learned with neural networks still struggle to generalize in closed-loop predictions (e.g., output colliding trajectories). This issue largely arises from the non-i.i.d. nature of sequential prediction in conjunction with ill-distributed training data. Intuitively, if the training data only comes from human behaviors in safe spaces, i.e., from “positive” examples, it is difficult for learning algorithms to capture the notion of “negative” examples like collisions. In this work, we aim to address this issue by explicitly modeling negative examples through self-supervision: (i) we introduce a social contrastive loss that regularizes the extracted motion representation by discerning the ground-truth positive events from synthetic negative ones; (ii) we construct informative negative samples based on our prior knowledge of rare but dangerous circumstances. Our method substantially reduces the collision rates of recent trajectory forecasting, behavioral cloning and reinforcement learning algorithms, outperforming state-of-the-art methods on several benchmarks. Our code is available at <https://github.com/vita-epfl/social-nce>.

1. Introduction

Humans have an instinctive ability to anticipate the future motions of other people while navigating in crowded spaces. This ability allows us to not only keep a comfortable distance from others but also identify potential dangers or discomforts ahead of time. However, building predictive models capable of doing so is challenging. Recent works have proposed a plethora of neural network-based models [6, 27, 41, 43, 47, 57, 84, 90, 96] to learn socially-aware motion representations for multi-agent problems such as trajectory forecasting [3, 33, 54, 85] and robot navigation [15, 18, 19]. Yet existing methods still output unacceptable solutions (e.g., collisions) from time to time in closed-loop predictions, which raises significant safety concerns for real-world deployment.

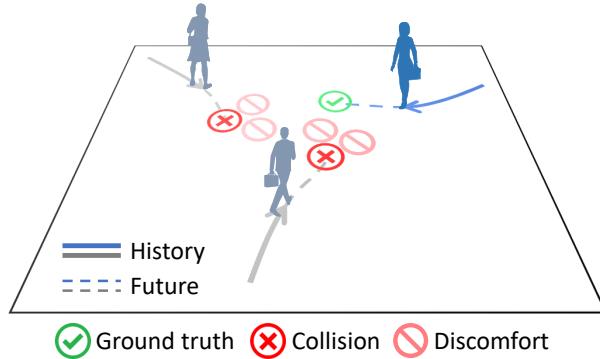


Figure 1: Illustration of social contrastive learning. Our method promotes the robustness of neural motion models by means of contrastive representation learning combined with negative data augmentation in the multi-agent context.

One key difficulty in learning robust motion representation stems from the common shortage of critical data. Very often, the training data is only collected from human behaviors in safe scenarios. The lack of near-accident examples poses a significant challenge to the discovery of social norms from data. As a consequence, prediction errors made by the learned model may accumulate over time, gradually create a discrepancy between the training and test state distributions, and eventually cause catastrophic errors [22, 25, 81].

Previous methods attempt to mitigate this issue through interactive data collections, such as expert queries [26, 53, 60, 82, 92] and additional experiments [12, 40, 48, 77, 97]. Unfortunately, these methods are not only costly and tedious but often impractical for forecasting problems, since human behaviors can hardly be intervened at scale for data collection. These shortcomings motivate us to explore an alternative approach: *given a fixed training dataset, can we learn a robust motion representation by exploiting our prior knowledge through self-supervision?*

To this end, we propose a social contrastive learning method built with negative data augmentation (Figure 1). Our main idea is to promote robust representations by *learning from the opposites* [74]. Intuitively, an effective way to elucidate the social norms that give rise to the “positive” examples is to explicitly portray the opposite “neg-

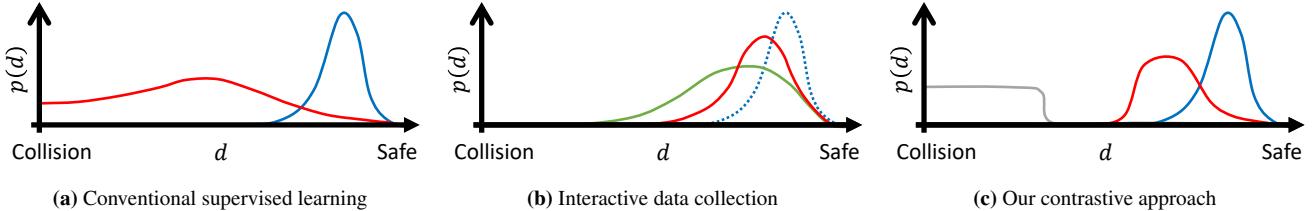


Figure 2: Illustration of different learning approaches to socially-aware sequential predictions. (a) The vanilla supervised learning approach often suffers from covariate shift between the training (blue) and test (red) data due to the dependence of the state distribution (e.g., separation distance d between agents) on the learned model [25, 81]. (b) Interactive data collection methods [26, 40, 53, 82] expand the training distribution to a wider range (green) through additional experiments, which are however expensive and even infeasible for forecasting problems. (c) Our social contrastive learning approach augments negative data based on prior knowledge, explicitly informing the learned model about unfavorable states (gray) for improved robustness.

tive” examples. We formulate this intuition into an auxiliary contrastive loss, named Social-NCE. It encourages the extracted motion representation to preserve sufficient information for distinguishing a positive event from a set of negative ones.

One crucial component of our method is the design of positive and negative events (states at a specific time step in the future). Existing contrastive methods in vision and language domains often rely on carefully designed data augmentation for positive pairs while using random samples to form negative pairs [16, 17, 32, 36, 65, 73, 98]. Despite its effectiveness for unsupervised pre-training, this common choice is not suitable for motion problems, since it does not bring any extra information than the main task about social norms. To more explicitly inject our prior knowledge, we introduce a social sampling strategy: construct the positive event from the ground-truth location of the primary agent and the negative events from the regions of other neighbors, given that one location cannot be occupied by multiple agents at the same time. As illustrated in Figure 2, our method can be viewed as a form of negative data augmentation [88]. It intentionally informs the model about low-density states through self-supervision, as opposed to laboriously collecting state-action pairs from dangerous scenarios.

We evaluate our method on three tasks: human trajectory forecasting, behavioral cloning, and reinforcement learning for robot navigation in crowds. Experimental results show that our proposed Social-NCE consistently reduces the collision rates and yields new state-of-the-art results on several benchmarks. Our method is model-agnostic and hence can be used as a generic component to promote the robustness of neural motion models.

2. Related Work

Socially-aware Motion Representations. Human motion in the social context has been traditionally studied based on relative distances and specific rules [4, 37, 64, 95].

While these hand-crafted models have been successfully applied to various tasks [23, 24, 30, 62, 99], they often struggle to capture the strong interactions among agents in complex scenes [83]. Some other methods attempt to learn the patterns of social behaviors from data. Yet early work often suffers from considerable performance drop in densely populated spaces due to limited modeling capacity [72, 94].

More recently, a variety of neural networks have been explored for learning socially-aware motion representations [5, 49]. Some peculiar neural architecture designs, e.g., feature pooling [3, 27, 33], attention mechanism [15, 41, 84, 96], spatio-temporal graph [43, 47, 57], latent variable modeling [13, 20, 78], injecting prior knowledge [9, 50], have yielded promising results in crowded environments. However, the robustness of these methods remains a central concern. Our work is orthogonal to the design of neural motion models and focused on the learning approach towards robust motion representations.

Covariate Shift. The problem of covariate shift was observed back in [75] and has been a persistent challenge for sequential prediction problems [25, 81]. One practical solution is to actively query experts [53, 82, 92], which has been shown effective for behavioral cloning but hardly applicable to forecasting problems [49, 79, 83]. Inverse reinforcement learning methods [1, 12, 40, 48, 67, 97, 101] jointly learn a reward function and the corresponding optimal policy to account for the sequential nature. However, they typically require extensive explorations to solve a reinforcement learning (RL) subproblem [28].

Another line of work introduces additional loss terms penalizing the predictions that lead to undesirable events, such as collisions and off-road trajectories [10, 68]. However, these penalties are dependent on the predicted states during training and become utterly ineffective once the model fits the dataset well in late training stages.

Closely related to our work, [63, 100] propose to learn a robust value (or cost) function by making use of negative samples. Our method differs from theirs in two key aspects: [63, 100] change the task loss that directly affects

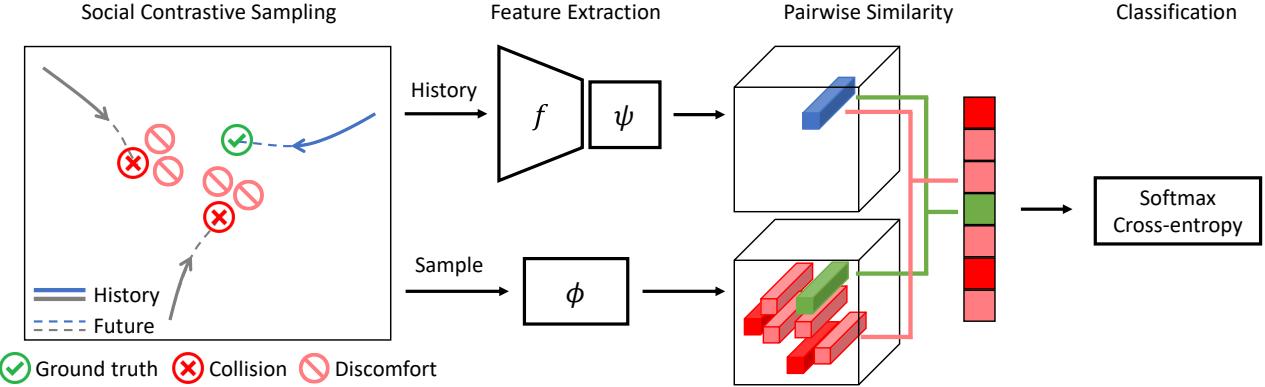


Figure 3: Social contrastive learning in the multi-agent context. Given a scenario that contains a primary agent of interest (blue) and multiple neighboring agents in the vicinity (gray), our Social-NCE loss encourages the extracted motion representation, in an embedding space, to be close to a positive future event and apart from some synthetic negative events that could have caused collisions or discomforts.

(and potentially biases) the model output, whereas our goal is to enhance the extracted motion representation without modifications in the main task; they draw negative samples randomly, in contrast, we design a more informed sampling strategy.

Contrastive Learning. Contrastive learning was proposed in [35] to learn an embedding space such that a simple similarity measure can approximate high-level semantic relations. This approach has recently achieved stunning results in a broad spectrum of areas, including computer vision [16, 36], natural language understanding [7, 61, 70], image synthesis [71] and robotics [87]. Some detailed design choices, such as positive and negative sampling, often play a critical role to the empirical success of contrastive methods [21, 45, 76, 80]. To the best of our knowledge, we are the first to adapt contrastive learning in the multi-agent motion context and explore the sampling method which is unique and critical to socially-aware motion representation learning.

3. Method

The robustness of neural motion models has been a long-standing issue [8, 38, 59, 93]. This issue is particularly concerning in sequential predictions such as behavior forecasting and autonomous navigation. Very often, the training data only contains “positive” examples collected from safe states without any dangerous “negative” occurrences. The severely imbalanced training distribution poses a significant challenge for learning algorithms to truly capture the underlying social norms and generalize to challenging scenarios.

In this section, we present a learning method that aims to tackle this challenge by means of contrastive representation learning combined with negative data augmentation. We will first briefly describe the basic idea of contrastive learning and then present a social contrastive loss. We will finally introduce a sampling strategy tailored for the multi-

agent context.

3.1. Contrastive Representation Learning

Representation learning typically consists in learning a parametric function (*i.e.*, encoder) that maps the raw data into a feature space to extract abstract and useful information for downstream tasks [11]. Recent contrastive learning methods often adopt the principle of noise contrastive estimation in an embedding space, namely the InfoNCE loss [29, 34, 69], to train an encoder:

$$\mathcal{L}_{\text{NCE}} = -\log \frac{\exp(\text{sim}(q, k^+)/\tau)}{\sum_{n=0}^N \exp(\text{sim}(q, k_n)/\tau)}, \quad (1)$$

where the encoded query q is brought close to one positive key $k_0 = k^+$ and pushed apart from N negative keys $\{k_1, \dots, k_N\}$, τ is a temperature hyperparameter, and $\text{sim}(u, v) = u^\top v / (\|u\| \|v\|)$ is the cosine similarity between two feature vectors. It has been shown that minimizing the InfoNCE loss is equivalent to maximize the lower bound on the mutual information between the raw input and the latent representation [69]. Moreover, the representations learned by this approach have provable performance guarantees on downstream tasks [7]. The empirical success of this approach often highly relies on the informativeness of positive and negative samples [16, 21, 44, 45, 76, 80, 89].

3.2. Social NCE

Consider a trajectory forecasting problem in crowded spaces as an example. Let $s_t^i = (x_t^i, y_t^i)$ denote the position of agent i at time t and $s_t = \{s_t^1, \dots, s_t^M\}$ denote the joint state of M agents in the scene. Given a sequence of history observations $\{s_1, \dots, s_t\}$, the task is to predict future trajectories of all agents $\{s_{t+1}, \dots, s_T\}$ until time T . Many recent forecasting models are designed as encoder-decoder neural networks, where the motion encoder $f(\cdot)$ first extracts a compact representation h_t^i with respect to agent i

and the decoder $g(\cdot)$ subsequently rolls out its future trajectory $\hat{s}_{t+1:T}^i$:

$$\begin{aligned} h_t^i &= f(s_{1:t}, i), \\ \hat{s}_{t+1:T}^i &= g(h_t^i). \end{aligned} \quad (2)$$

To model social interactions among agents, $f(\cdot)$ typically contains two sub-modules: a sequential module $f_S(\cdot)$ that encodes each individual sequence and an interaction module $f_I(\cdot)$ that shares information among agents, *e.g.*,

$$\begin{aligned} z_t^i &= f_S(h_{t-1}^i, s_t^i), \\ h_t^i &= f_I(z_t^i, i). \end{aligned} \quad (3)$$

where z_t^i is the latent representation of agent i given the observation of its own state at time t and $z_t = \{z_t^1, \dots, z_t^M\}$. A variety of architectures have been explored for each modules and validated on accuracy measures [3, 57, 58]. Nevertheless, their robustness remains an open issue. Several recent works [10, 49] have shown that trajectories predicted by existing models often output socially unacceptable solutions (*e.g.*, collisions), indicating a lack of common sense about social norms.

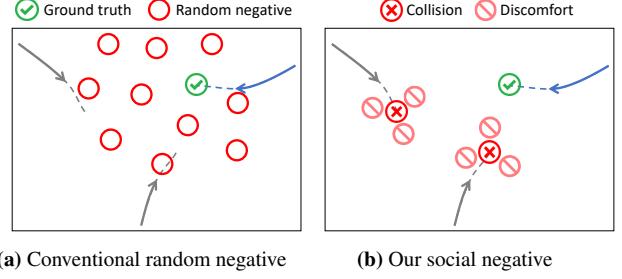
To tackle this challenge, we propose Social-NCE, a variant of InfoNCE adapted to socially-aware motion representations. As illustrated in Figure 3, we construct the encoded query and key vectors for the primary agent i at time t as follows:

- query: embedding of history observations $q = \psi(h_t^i)$, where $\psi(\cdot)$ is an MLP projection head.
- key: embedding of a future event $k = \phi(s_{t+\delta t}^i, \delta t)$, where $\phi(\cdot)$ is an event encoder modeled by an MLP, $s_{t+\delta t}^i$ is a sampled spatial location and $\delta t > 0$ is the sampling horizon.

By tuning $\delta t \in \Lambda$ in a range, *e.g.*, $\Lambda = \{1, \dots, 4\}$, we can take into account future events in the next few steps simultaneously. Nevertheless, when δt is a fixed value, $\phi(\cdot)$ can be simplified as a location encoder, *i.e.*, $\phi(s_{t+\delta t}^i)$.

In each frame, we draw one positive key and multiple negative keys based on future trajectories in the scene, which we will describe in the next Section 3.3. Following [16, 36, 98], we normalize the embedding vectors onto a unit sphere and train the parametric models $f(\cdot), \phi(\cdot), \psi(\cdot)$ jointly with the objective of mapping the positive pair of query and key to similar points, relative to the other negative pairs, in the embedding space:

$$\mathcal{L}_{\text{SocialNCE}} = -\log \frac{\exp(\psi(h_t^i) \cdot \phi(s_{t+\delta t}^{i,+}, \delta t) / \tau)}{\sum_{\delta t \in \Lambda} \sum_{n=0}^N \exp(\psi(h_t^i) \cdot \phi(s_{t+\delta t}^{i,n}, \delta t) / \tau)}. \quad (4)$$



(a) Conventional random negative (b) Our social negative

Figure 4: Different negative sampling methods in the multi-agent context. (a) The conventional random sampling method draws negative samples homogeneously scattered in the space, which does not provide much information about social norms. (b) Our social sampling strategy seeks more informative negative samples from the neighborhood of other agents in the future.

The full training objective is a weighted combination of the conventional task loss, *e.g.*, mean squared error (MSE) or negative log-likelihood (NLL) for trajectory forecasting, and the proposed social contrastive loss:

$$\mathcal{L}(f, g, \psi, \phi) = \mathcal{L}_{\text{task}}(f, g) + \lambda \mathcal{L}_{\text{SocialNCE}}(f, \psi, \phi), \quad (5)$$

where λ is a hyper-parameter controlling the emphasis on the Social-NCE loss.

3.3. Multi-agent Contrastive Sampling

One critical design choice of the proposed social contrastive learning lies in the sampling strategy. The recent successes in contrastive learning of visual representations are heavily tied to the use of a large set of negative samples uniformly drawn from the training dataset [16, 32, 36, 65, 73, 98]. Unfortunately, this common practice is not suitable for socially-aware motion representation learning. As the main predictive loss already encourages the model to replicate the socially good behaviors from training examples, adding another discrimination task between the correct solution and other randomly scattered negatives cannot provide much extra information about social norms. Worse yet, the random negative sampling, like Figure 4a, may contradict to the multimodal nature of the future trajectories and incorrectly penalize plausible solutions.

To effectively incorporate our domain knowledge of socially unfavorable events in the multi-agent context, we propose a social sampling strategy. As shown in Figure 4b, we draw a set of negative samples from the neighborhood of other agents in the future at time $t + \delta t$,

$$s_{t+\delta t}^{i,n-} = s_{t+\delta t}^j + \Delta s_p + \epsilon, \quad (6)$$

where $j \in \{1, 2, \dots, M\} \setminus i$ is the index of other agents, and $\Delta s_p = (\rho \cos \theta_p, \rho \sin \theta_p)$ is a local displacement to account for the social comfort area. For instance, ρ can be the minimum physical distance between two agents and $\theta_p =$

$0.25p\pi, p \in \{0, 1, \dots, 7\}$. Thus, a total of $N = 8(M - 1)$ negative samples are synthesized. We also add random perturbations to each sampled location $\epsilon \sim \mathcal{N}(0; c_\epsilon \mathbf{I})$, where c_ϵ is a small constant, *e.g.*, 0.05 [m], to prevent over-fitting. For positive sampling, we pick a location from the ground truth region of the primary agent i at time $t + \delta t$,

$$s_{t+\delta t}^{i,+} = s_{t+\delta t}^i + \epsilon. \quad (7)$$

The key intuition behind our method is that the conventional learning approaches only focus on replicating “positive” behaviors in normal states, without the need to understand the consequence of “negative” ones. In contrast, by using Social-NCE in tandem with our proposed sampling strategy, we actively enforce the extracted motion representation h_t^i to contain necessary information for identifying the scenarios that could have led to catastrophic outcomes. This subtle but essential difference enables the model to learn a significantly more robust representation from the fixed training dataset.

4. Experiment

We empirically validate the proposed Social-NCE on three different tasks: (i) human trajectory forecasting, (ii) imitation learning and (iii) reinforcement learning for robot navigation in multi-agent environments.

On each task, we compare the models obtained by three training methods:

- Vanilla: models trained without contrastive loss.
- Random: models trained with contrastive loss and the random negative sampling (Figure 4a).
- Social-NCE (ours): models trained with contrastive loss and the social negative sampling strategy (Figure 4b).

4.1. Implementation Details

In our experiments, we use two different 2-layer MLPs as the projection head $\psi(\cdot)$ and the event encoder $\phi(\cdot)$. We encode the history observations and future events into 8-dimensional embedding vectors. The distance hyper-parameter ρ is set as 0.2 [m] for trajectory forecasting tasks and 0.6 [m] for robot navigation according to the geometric size of agents in environments. By default, the sampling horizon δt is set up to 4 and the temperature τ is set as 0.1. All models are trained with the Adam optimizer [46].

4.2. Trajectory Forecasting

We first evaluate our method on the human trajectory forecasting task. We compare the performances of several forecasting models trained with and without the proposed Social-NCE on the ETH & UCY benchmark [55, 72] and

the Trajnet++ interaction-centric benchmark [49]. For a fair and direct comparison with prior work, we implement our learning method in the official public code of each model without any modifications on architectures.

All forecasting models are trained and evaluated in the following setting: given 8 time steps (3.2 seconds) of observations as input, predict future trajectories for 12 time steps (4.8 seconds) for all human agents in the scene. Similar to previous work [33, 49, 100], we measure the performance of each model with two metrics:

- Final displacement error (FDE): the Euclidean distance between the predicted output and the ground truth at the last time step.
- Collision rate (COL): the percentage of test cases where the predicted trajectories of different agents run into collisions.

More experimental details are outlined in the Appendix A.1.

4.2.1 ETH & UCY Benchmark

We start the evaluation of our method on the ETH and UCY datasets, which contain 5 subsets of real-world pedestrian trajectories widely used in previous work [3, 33, 66, 85]. We consider the following two state-of-the-art multi-modal forecasting models as baselines:

- Social-STGCNN [66]: a spatio-temporal convolutional network with graph representations of pedestrian trajectories.
- Trajectron++ [85]: a VAE-based recurrent model using element-wise sum to aggregate interactions.

Table 1 shows the experimental results in terms of Top-20 FDE and collision rate. Compared with the conventional predictive learning, our Social-NCE reduces the collision rate by 37.0% on average for the Social-STGCNN and 45.7% for the Trajectron++, while remaining competitive with respect to the final displacement error. These results suggest the benefits of our method for boosting the robustness of modern trajectory forecasting models without hurting the accuracy and diversity of endpoint predictions.

4.2.2 Trajnet++ Benchmark

We further validate our method on the Trajnet++ [49], an emerging open challenge with an emphasis on interaction modeling. Our evaluation is performed on the following interacting sub-categories:

- Avoidance: the sub-category where the primary pedestrian avoids others coming from the opposite direction.
- Group: the sub-category where the primary pedestrian keeps a close and approximately constant distance with one or more moving neighbors.

Dataset	Social-STGCNN [66]					Trajectron++ [85]				
	Vanilla		Ours		Gain	Vanilla		Ours		Gain
	FDE ↓	COL ↓	FDE ↓	COL ↓	COL ↑	FDE ↓	COL ↓	FDE ↓	COL ↓	COL ↑
ETH	1.223	1.33	1.224	0.61	54.1%	0.810	1.16	0.791	0.00	100.0%
Hotel	0.687	3.82	0.678	3.35	12.3%	0.184	0.84	0.177	0.38	54.6%
Univ	0.912	9.11	0.879	6.44	29.3%	0.450	3.38	0.435	3.08	8.9%
Zara1	0.525	2.27	0.515	1.02	55.1%	0.320	0.46	0.330	0.18	61.5%
Zara2	0.480	6.86	0.482	3.37	50.9%	0.250	1.03	0.255	0.99	3.3%
Average	0.765	4.70	0.756	2.96	37.0%	0.403	1.37	0.398	0.93	45.7%

Table 1: Comparison between Social-NCE and the vanilla predictive learning for two recent multi-modal trajectory forecasting models, Social-STGCNN [66] and Trajectron++ [85], on the ETH [72] / UCY [55] datasets. Our method reduces the collision rates of these state-of-the-art models by more than 37%, while being on par with the vanilla training counterparts in terms of the top-20 final displacement error. Note that we run the official public code to obtain the baseline results, which are subject to mild differences (< 2% on average) from the corresponding papers.

Category	Social-LSTM [3]					Directional-LSTM [49]				
	Vanilla		Ours		Gain	Vanilla		Ours		Gain
	FDE ↓	COL ↓	FDE ↓	COL ↓	COL ↑	FDE ↓	COL ↓	FDE ↓	COL ↓	COL ↑
Avoidance	1.23	13.12	1.23	10.83	17.5%	1.33	12.92	1.33	10.42	19.4%
Group	0.97	7.44	0.97	4.65	37.5%	1.05	6.51	1.05	5.89	9.5%
Overall	1.14	6.44	1.14	5.31	17.6%	1.22	5.49	1.22	4.59	16.4%

Table 2: Comparison between Social-NCE and the vanilla predictive learning for two top-ranking uni-modal forecasting models on different interacting categories of the Trajnet++ benchmark [49]. Our method reduces the collision rate of the previously most robust model by a clear margin. Note that we obtain all results from the benchmark reports, where FDEs are rounded to two decimal places.

- Overall: all the scenes with a presence of social interactions, including the ones that are hard to categorize.

These interacting categories exclude both *linear* trajectories that often dilute the evaluations of interaction modeling and *non-linear yet non-agent-interacting* trajectories that are strongly affected by the scene context or other unobservable variables, and are therefore particularly suitable for assessing the social awareness of forecasting models.

We use two top-ranking models on the Trajnet++ benchmark as baselines:

- Social-LSTM [3]: a LSTM-based model with an interaction module over hidden states of nearby agents.
- Directional-LSTM [49]: a LSTM-based model with an interaction module sharing velocities of nearby agents.

Table 2 shows that, similar to the results on the ETH & UCY benchmark, our social contrastive method yields lower collision rates than the vanilla predictive learning by a clear margin (9.5%-37.5%) across all considered sub-categories. In particular, the Directional-LSTM trained with our method clearly outperforms its counterpart, which was previously the most robust model on the public benchmark. It is worth noting that, again, our method does not have a significant impact on prediction accuracy. We conjecture

that this is because our method tends to adjust the output trajectories locally instead of changing their global patterns, as reflected by the qualitative examples in the Appendix C.

4.3. Imitation Learning

Next, we examine the effectiveness of Social-NCE applied to imitation learning for robot navigation in dense crowds [15, 18, 19]. We use an open-sourced simulator of crowd navigation [15], where the task for the robot is to navigate through 5 simulated pedestrians and arrive at the target destination with time efficiency. In each time step, the robot receives the observable states of other agents and outputs an action. We follow the evaluation protocol in [15], which quantifies the performance of a policy using three metrics: navigation time, collision rate, and the accumulated reward as follows:

$$r(s_t, a_t) = \begin{cases} -0.25 & \text{if } d_m^t < 0 \\ -0.1 + d_m^t/2 & \text{else if } d_m^t < 0.2 \\ 1 & \text{else if goal is reached} \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

where d_m^t is the minimum separation distance between the robot and the humans during the time interval $[t - \Delta t, t]$.

Method	Reward \uparrow	Time (s) \downarrow	Collision (%) \downarrow
Vanilla	0.28 ± 0.01	10.31 ± 0.07	11.11 ± 1.45
Random	0.24 ± 0.02	10.32 ± 0.12	18.60 ± 4.69
Ours	0.32 ± 0.01	10.33 ± 0.07	3.40 ± 1.36

Table 3: Quantitative results of imitation learning with different methods on a 5k demonstration dataset. Higher is better for reward, and lower is better for the other metrics. Compared with the vanilla baseline, our method brings down the collision rate by approximately 69%.

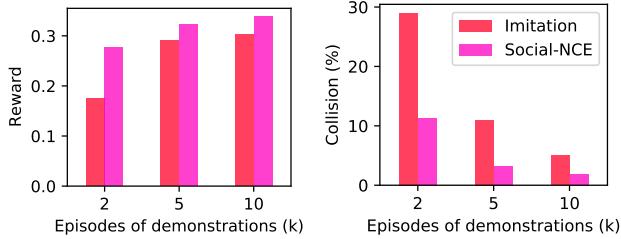


Figure 5: Social-NCE for imitation learning with different amounts of demonstrations. The conventional behavioral cloning method suffers from a significant performance drop in the low data regime, whereas our method is able to retain much better results thanks to the additional information absorbed from the social contrastive task.

4.3.1 Behavioral Cloning

For imitation learning, we collect a demonstration dataset that consists of 5000 simulation episodes using the pre-trained SARNL policy in [15] as an expert. We train an imitator on the collected data for 200 epochs and evaluate the average performance of the last 10 models saved every 5 training epochs.

As shown in Table 3, the imitation learning algorithm trained with random negative sampling fails to outperform the vanilla baseline. In fact, it even worsens the learned policy. In contrast, our method (with weight $\lambda = 0.1$) leads to consistently higher reward and lower collision rate. Specifically, our method reduces the collision rate by approximately 69% compared with the vanilla baseline and attains an average reward of 0.323, which is highly close to the result of the demonstrator in [15].

4.3.2 Low-data Regime

The performance of the standard behavioral cloning approach often degrades substantially when provided with limited demonstrations. We examine the potential of the proposed Social-NCE for data-efficient imitation learning by comparing policies trained on datasets of different sizes. As shown in Figure 5, with decreasing amounts of demonstrations, the performance of the vanilla method drops sharply. Notably, the baseline model trained on 2k episodes of demonstrations causes collisions in 29% of test cases. In

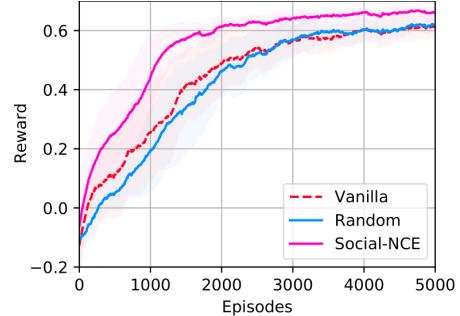


Figure 6: Learning curves of Rainbow DQN [39] with different methods for crowd navigation. Results are averaged across 8 random seeds. The shaded area spans one standard deviation. In contrast to the vanilla and random negative counterparts, the agent with Social-NCE is significantly more sample efficient and achieves higher final reward.

Method	Reward w.r.t. fraction of dataset			
	100%	50%	25%	10%
Vanilla	80.1%	75.2%	53.2%	14.4%
Random	81.3%	71.5%	51.7%	7.0%
Ours	91.6%	84.6%	79.2%	69.0%

Table 4: Offline RL normalized scores attained by the vanilla rainbow and Social-NCE agents (higher is better). Normalized score is calculated as: $100 \times (\text{agent score} - \text{random play score}) / (\text{optimal agent score} - \text{random play score})$. Our Social-NCE consistently facilitates the recovery of a near-optimal policy and is particularly advantageous in the low-data regime.

contrast, our method succeeds in retaining a much higher reward and safety in the low-data regime. For instance, the collision rate of our method with 2k demonstrations is comparable to the baseline with 5k demonstrations. Similarly, our method using 5k training data obtains a higher reward than the counterpart using 10k training data. This result suggests that the learner can absorb a considerable amount of useful information from the designed negative data augmentation, greatly alleviating the information shortage in small training sets.

4.4 Reinforcement Learning

Finally, we evaluate the proposed Social-NCE for reinforcement learning (RL) algorithms on the crowd navigation task. We adopt the Rainbow DQN [39], a state-of-the-art model-free RL method, as baseline and follow the architecture of value-based SARNL policy [15] to build the encoder $f(\cdot)$. To effectively apply Social-NCE to the Rainbow agent, we add a linear layer after the interaction and pooling modules. We also replace the planning module in SARNL by dueling and categorical layers, as in the standard Rainbow agent [39]. In order to isolate the impact of Social-NCE, we use the dense reward function proposed in [86], which

Horizon	Vanilla	1	2	3	4	1-4
Reward \uparrow	0.283 ± 0.008	0.281 ± 0.019	0.296 ± 0.009	0.311 ± 0.009	0.307 ± 0.012	0.323 ± 0.005
Time (s) \downarrow	10.306 ± 0.068	10.345 ± 0.065	10.281 ± 0.141	10.322 ± 0.134	10.348 ± 0.107	10.334 ± 0.072
Collision (%) \downarrow	11.11 ± 1.45	11.24 ± 3.46	9.13 ± 2.02	5.83 ± 1.62	6.09 ± 2.26	3.40 ± 1.36

Table 5: Social-NCE for imitation learning with different sampling horizons. Higher is better for reward, and lower is better for the other metrics. Taking multiple time steps (1-4) into account simultaneously yields better results than a fixed horizon of one time step.

eliminates the necessity of imitation pre-training in [15].

4.4.1 Off-policy Reinforcement Learning

We first validate our Social-NCE method in the standard off-policy setting, where an RL agent learns from the replay buffer data gathered over the learning process. We set the temperature as $\tau = 0.2$ and weight as $\lambda = 1.0$ for the Social-NCE loss. Figure 6 shows the experimental results of each method averaged over 8 random seeds.

The vanilla Rainbow agent reaches a reward value of 0.6 using more than 4000 episodes. In comparison, the agent equipped with our method demonstrates a much higher sample efficiency. It attains the same level of reward in less than 2000 episodes and quickly obtains a collision-free policy thanks to the prior knowledge from the social contrastive task. Additionally, our method also offers a slight improvement in the final performance. On the contrary, the random negative sampling is not able to provide any significant performance gain, similar to the experimental results above.

4.4.2 Offline Reinforcement Learning

Lastly, we explore the potential of our method in the offline RL setting, in which an agent learns from a static dataset of logged experiences without additional interactions with the environment [56]. The offline RL setting has attracted rapidly growing attentions due to its tremendous promise for making good use of immerse experience datasets. Nevertheless, most deep reinforcement learning algorithms are highly vulnerable to the distribution mismatch between the policy being trained and the ones used for data collection [2, 31, 42, 51, 52].

To verify our method in the offline RL setting, we collect a dataset using the vanilla rainbow agent in the following process: (i) 10k episodes are collected during online RL training from scratch, (ii) multiple free explorations are carried out using online RL checkpoint models trained after K episodes, where $K \in \{500, 1000, 3000, 5000\}$. Each free exploration contains 5k episodes, and the full dataset is made up of 30k episodes of trials in total. We train an offline Rainbow policy on $N\%$ of the experiences randomly-sampled from the dataset, where $N \in \{10, 25, 50, 100\}$. We use temperature $\tau = 0.2$ and weight $\lambda = 0.1$.

Table 4 reports the average performance of different offline methods across 10 random seeds. Due to the aforementioned distributional shift, no offline methods attain reward scores as high as the online RL algorithm. Nevertheless, our Social-NCE substantially narrows the performance gap and consistently delivers better results than the vanilla rainbow. Notably, the offline policy with our method achieves comparable performance to the best of vanilla baseline using only 25% of the collected data.

4.5. Ablation: Event Encoder

To validate the benefits of event encoder $\phi(s_{t+\delta t}^i, \delta t)$, we compare the performance of Social-NCE for imitation learning with different sampling horizons. When the sampling horizon is a fixed value, we use a simplified location encoder $\phi(s_{t+\delta t}^i)$ that only takes the location of a sample as input. Table 5 reports the results obtained by using contrastive samples either at a single fixed horizon in a range from 1 to 4 or across all four steps simultaneously. Among the single-step choices, $\delta t = 3$ yields significant performance gains on both reward and collision metrics. On the contrary, $\delta t = 1$ does not provide any improvements in comparison to the baseline due to its short-sightedness. When taking all four steps into account, our method attains the best result, suggesting the importance of establishing social contrastive tasks at multiple horizons.

5. Conclusion

In this work, we present a contrastive method for learning socially-aware motion representations. The proposed Social-NCE loss, combined with our sampling strategy, significantly boosts the performance of recent human trajectory forecasting and crowd navigation algorithms. Our result suggests that learning from the opposite by means of negative data augmentations can be a promising alternative to the traditional interactive data collection for robust sequential predictions.

Acknowledgments

This work is supported by the Swiss National Science Foundation under the Grant 200021-L92326. We thank Parth Kothari, Yifan Sun, Taylor Mordan, Mohammadhosseini Bahari, Lorenzo Bertoni and Sven Kreiss for valuable feedback on early drafts.

References

[1] Pieter Abbeel and Andrew Y. Ng. Apprenticeship Learning via Inverse Reinforcement Learning. In *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04. ACM, 2004.

[2] Rishabh Agarwal, Dale Schuurmans, and Mohammad Norouzi. An Optimistic Perspective on Offline Reinforcement Learning. In *International Conference on Machine Learning*, pages 104–114. PMLR, Nov. 2020. ISSN: 2640-3498.

[3] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social LSTM: Human Trajectory Prediction in Crowded Spaces. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 961–971, June 2016. ISSN: 1063-6919.

[4] Alexandre Alahi, Vignesh Ramanathan, and Li Fei-Fei. Socially-Aware Large-Scale Crowd Forecasting. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2211–2218, June 2014. ISSN: 1063-6919.

[5] Alexandre Alahi, Vignesh Ramanathan, Kratarth Goel, Alexandre Robicquet, Amir A Sadeghian, Li Fei-Fei, and Silvio Savarese. Learning to predict human behavior in crowded scenes. In *Group and Crowd Behavior for Computer Vision*, pages 183–207. Elsevier, 2017.

[6] Javad Amirian, Jean-Bernard Hayet, and Julien Pettré. Social Ways: Learning Multi-Modal Distributions of Pedestrian Trajectories With GANs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[7] Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A Theoretical Analysis of Contrastive Unsupervised Representation Learning. *arXiv:1902.09229 [cs, stat]*, Feb. 2019. arXiv: 1902.09229.

[8] Samaneh Azadi, Catherine Olsson, Trevor Darrell, Ian Goodfellow, and Augustus Odena. Discriminator Rejection Sampling. In *International Conference on Learning Representations*, Sept. 2018.

[9] Mohammadhossein Bahari, Nejjar Ismail, and Alexandre Alahi. Injecting knowledge in data-driven vehicle trajectory predictors. *Transportation Research Part C*, 2021.

[10] Mayank Bansal, Alex Krizhevsky, and Abhijit Ogale. ChauffeurNet: Learning to Drive by Imitating the Best and Synthesizing the Worst. *arXiv:1812.03079 [cs]*, Dec. 2018. arXiv: 1812.03079.

[11] Y. Bengio, A. Courville, and P. Vincent. Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, Aug. 2013. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[12] Kianté Brantley, Wen Sun, and Mikael Henaff. Disagreement-Regularized Imitation Learning. In *International Conference on Learning Representations*, Sept. 2019.

[13] Yuning Chai, Benjamin Sapp, Mayank Bansal, and Dragomir Anguelov. MultiPath: Multiple Probabilistic Anchor Trajectory Hypotheses for Behavior Prediction. In *Conference on Robot Learning*, pages 86–99. PMLR, May 2020.

[14] T. Chavdarova, P. Baqué, S. Bouquet, A. Maksai, C. Jose, T. Bagautdinov, L. Lettry, P. Fua, L. Van Gool, and F. Fleuret. WILDTRACK: A Multi-camera HD Dataset for Dense Unscripted Pedestrian Detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5030–5039, June 2018. ISSN: 2575-7075.

[15] Changan Chen, Yuejiang Liu, Sven Kreiss, and Alexandre Alahi. Crowd-Robot Interaction: Crowd-Aware Robot Navigation With Attention-Based Deep Reinforcement Learning. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6015–6022, May 2019. ISSN: 2577-087X.

[16] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. *arXiv:2002.05709 [cs, stat]*, Mar. 2020. arXiv: 2002.05709.

[17] Xinlei Chen and Kaiming He. Exploring Simple Siamese Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021.

[18] Yu Fan Chen, Michael Everett, Miao Liu, and Jonathan P. How. Socially aware motion planning with deep reinforcement learning. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1343–1350, Sept. 2017. ISSN: 2153-0866.

[19] Yu Fan Chen, Miao Liu, Michael Everett, and Jonathan P. How. Decentralized Non-communicating Multiagent Collision Avoidance with Deep Reinforcement Learning. *arXiv:1609.07845 [cs]*, Sept. 2016. arXiv: 1609.07845.

[20] Chiho Choi, Srikanth Malla, Abhishek Patil, and Joon Hee Choi. DROGON: A Trajectory Prediction Model based on Intention-Conditioned Behavior Reasoning. *arXiv:1908.00024 [cs]*, Nov. 2020. arXiv: 1908.00024.

[21] Ching-Yao Chuang, Joshua Robinson, Lin Yen-Chen, Antonio Torralba, and Stefanie Jegelka. Debiased Contrastive Learning. *arXiv:2007.00224 [cs, stat]*, July 2020. arXiv: 2007.00224.

[22] Felipe Codevilla, Eder Santana, Antonio M. Lopez, and Adrien Gaidon. Exploring the Limitations of Behavior Cloning for Autonomous Driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9329–9338, 2019.

[23] Pasquale Coscia, Francesco Castaldo, Francesco AN Palmieri, Alexandre Alahi, Silvio Savarese, and Lamberto Ballan. Long-term path prediction in urban scenarios using circular distributions. *Journal on Image and Vision Computing (JIVC)*, 2018.

[24] Pasquale Coscia, Francesco Castaldo, Francesco AN Palmieri, Lamberto Ballan, Alexandre Alahi, and Silvio Savarese. Point-based path prediction from polar histograms. In *2016 19th International Conference on Information Fusion (FUSION)*, pages 1961–1967. IEEE, 2016.

[25] Hal Daumé, John Langford, and Daniel Marcu. Search-based structured prediction. *Machine Learning*, 75(3):297–325, June 2009.

[26] Pim de Haan, Dinesh Jayaraman, and Sergey Levine.

Causal Confusion in Imitation Learning. *arXiv:1905.11979 [cs, stat]*, Nov. 2019. arXiv: 1905.11979.

[27] Nachiket Deo and Mohan M. Trivedi. Convolutional Social Pooling for Vehicle Trajectory Prediction. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1549–15498, June 2018. ISSN: 2160-7516.

[28] Gabriel Dulac-Arnold, Daniel Mankowitz, and Todd Hester. Challenges of Real-World Reinforcement Learning. *arXiv:1904.12901 [cs, stat]*, Apr. 2019. arXiv: 1904.12901.

[29] Chris Dyer. Notes on Noise Contrastive Estimation and Negative Sampling. *arXiv:1410.8251 [cs]*, Oct. 2014. arXiv: 1410.8251.

[30] Gonzalo Ferrer, Anais Garrell, and Alberto Sanfeliu. Robot companion: A social-force based approach with human awareness-navigation in crowded environments. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1688–1694, Nov. 2013. ISSN: 2153-0866.

[31] Scott Fujimoto, David Meger, and Doina Precup. Off-Policy Deep Reinforcement Learning without Exploration. In *International Conference on Machine Learning*, pages 2052–2062. PMLR, May 2019. ISSN: 2640-3498.

[32] Yoav Goldberg and Omer Levy. word2vec Explained: deriving Mikolov et al.’s negative-sampling word-embedding method. *arXiv:1402.3722 [cs, stat]*, Feb. 2014. arXiv: 1402.3722.

[33] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi. Social GAN: Socially Acceptable Trajectories with Generative Adversarial Networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2255–2264, June 2018. ISSN: 2575-7075.

[34] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, Mar. 2010. ISSN: 1938-7228.

[35] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality Reduction by Learning an Invariant Mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR’06)*, volume 2, pages 1735–1742, New York, NY, USA, 2006. IEEE.

[36] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. *arXiv:1911.05722 [cs]*, Mar. 2020. arXiv: 1911.05722.

[37] Dirk Helbing and Peter Molnar. Social Force Model for Pedestrian Dynamics. *Physics Review E*, May 1998.

[38] Dan Hendrycks and Thomas Dietterich. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. In *International Conference on Learning Representations*, Sept. 2018.

[39] Matteo Hessel, Joseph Modayil, Hado van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining Improvements in Deep Reinforcement Learning. *arXiv:1710.02298 [cs]*, Oct. 2017. arXiv: 1710.02298.

[40] Jonathan Ho and Stefano Ermon. Generative Adversarial Imitation Learning. *arXiv:1606.03476 [cs]*, June 2016.

[41] 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.

[42] Riashat Islam, Komal K. Teru, Deepak Sharma, and Joelle Pineau. Off-Policy Policy Gradient Algorithms by Constraining the State Distribution Shift. *arXiv:1911.06970 [cs, stat]*, Dec. 2019. arXiv: 1911.06970.

[43] Boris Ivanovic and Marco Pavone. The Trajectron: Probabilistic Multi-Agent Trajectory Modeling With Dynamic Spatiotemporal Graphs. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2375–2384, Oct. 2019. ISSN: 2380-7504.

[44] Allan Jabri, Andrew Owens, and Alexei Efros. Space-Time Correspondence as a Contrastive Random Walk. *Advances in Neural Information Processing Systems*, 33, 2020.

[45] Yannis Kalantidis, Mert Bulent Sarayildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard Negative Mixing for Contrastive Learning. *arXiv:2010.01028 [cs]*, Oct. 2020. arXiv: 2010.01028.

[46] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, Dec. 2014. arXiv: 1412.6980.

[47] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, Hamid Rezatofighi, and Silvio Savarese. Social-BiGAT: Multimodal Trajectory Forecasting using Bicycle-GAN and Graph Attention Networks. *Advances in Neural Information Processing Systems*, 32:137–146, 2019.

[48] Ilya Kostrikov, Kumar Krishna Agrawal, Debidatta Dwibedi, Sergey Levine, and Jonathan Tompson. Discriminator-Actor-Critic: Addressing Sample Inefficiency and Reward Bias in Adversarial Imitation Learning. *arXiv:1809.02925 [cs, stat]*, Oct. 2018. arXiv: 1809.02925.

[49] Parth Kothari, Sven Kreiss, and Alexandre Alahi. Human trajectory forecasting: A deep learning perspective. *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[50] Parth Kothari, Brian Siffringer, and Alexandre Alahi. Interpretable social anchors for human trajectory forecasting in crowds. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[51] Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. Stabilizing Off-Policy Q-Learning via Bootstrapping Error Reduction. *Advances in Neural Information Processing Systems*, 32:11784–11794, 2019.

[52] Sascha Lange, Thomas Gabel, and Martin Riedmiller. Batch Reinforcement Learning. In Marco Wiering and Martijn van Otterlo, editors, *Reinforcement Learning: State-of-the-Art*, Adaptation, Learning, and Optimization, pages 45–73. Springer, Berlin, Heidelberg, 2012.

[53] Michael Laskey, Jonathan Lee, Roy Fox, Anca Dragan, and Ken Goldberg. DART: Noise Injection for Robust Imitation Learning. In *Conference on Robot Learning*, pages 143–156. PMLR, Oct. 2017. ISSN: 2640-3498.

[54] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B. Choy, Philip H. S. Torr, and Manmohan Chandraker. DE-

SIRE: Distant Future Prediction in Dynamic Scenes with Interacting Agents. *arXiv:1704.04394 [cs]*, Apr. 2017. arXiv: 1704.04394.

[55] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by Example. *Computer Graphics Forum*, 26(3):655–664, 2007.

[56] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems. *arXiv:2005.01643 [cs, stat]*, May 2020. arXiv: 2005.01643.

[57] Jiachen Li, Fan Yang, Masayoshi Tomizuka, and Chiho Choi. EvolveGraph: Multi-Agent Trajectory Prediction with Dynamic Relational Reasoning. *arXiv:2003.13924 [cs]*, Oct. 2020. arXiv: 2003.13924.

[58] Lingyun Luke Li, Bin Yang, Ming Liang, Wenyuan Zeng, Mengye Ren, Sean Segal, and Raquel Urtasun. End-to-end Contextual Perception and Prediction with Interaction Transformer. *arXiv:2008.05927 [cs]*, Aug. 2020. arXiv: 2008.05927.

[59] Yuejiang Liu, Parth Kothari, and Alexandre Alahi. Collaborative Sampling in Generative Adversarial Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):4948–4956, Apr. 2020. Number: 04.

[60] Yuejiang Liu, An Xu, and Zichong Chen. Map-based Deep Imitation Learning for Obstacle Avoidance. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8644–8649, Oct. 2018. ISSN: 2153-0866.

[61] Lajanugen Logeswaran and Honglak Lee. An efficient framework for learning sentence representations. *arXiv:1803.02893 [cs]*, Mar. 2018. arXiv: 1803.02893.

[62] Matthias Luber, Johannes A. Stork, Gian Diego Tipaldi, and Kai O Arras. People tracking with human motion predictions from social forces. In *2010 IEEE International Conference on Robotics and Automation*, pages 464–469, May 2010. ISSN: 1050-4729.

[63] Yuping Luo, Huazhe Xu, and Tengyu Ma. Learning Self-Correctable Policies and Value Functions from Demonstrations with Negative Sampling. *arXiv:1907.05634 [cs, stat]*, Oct. 2019. arXiv: 1907.05634.

[64] 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, June 2009. ISSN: 1063-6919.

[65] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc., 2013.

[66] A. Mohamed, K. Qian, M. Elhoseiny, and C. Claudel. Social-STGCNN: A Social Spatio-Temporal Graph Convolutional Neural Network for Human Trajectory Prediction. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14412–14420, June 2020. ISSN: 2575-7075.

[67] Andrew Y. Ng and Stuart Russell. Algorithms for Inverse Reinforcement Learning. In *in Proc. 17th International Conf. on Machine Learning*, pages 663–670. Morgan Kaufmann, 2000.

[68] M. Niedoba, Henggang Cui, K. Luo, Darshan Hegde, Fang-Chieh Chou, and Nemanja Djuric. Improving Movement Prediction of Traffic Actors using Off-road Loss and Bias Mitigation, 2019.

[69] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation Learning with Contrastive Predictive Coding. *arXiv:1807.03748 [cs, stat]*, Jan. 2019. arXiv: 1807.03748.

[70] Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. Unsupervised Learning of Sentence Embeddings Using Compositional n-Gram Features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 528–540, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[71] Taesung Park, Alexei A. Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive Learning for Unpaired Image-to-Image Translation. *arXiv:2007.15651 [cs]*, July 2020. arXiv: 2007.15651.

[72] Stefano Pellegrini, Andreas Ess, and Luc Van Gool. Improving Data Association by Joint Modeling of Pedestrian Trajectories and Groupings. In *Computer Vision – ECCV 2010*, Lecture Notes in Computer Science, pages 452–465, Berlin, Heidelberg, 2010. Springer.

[73] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[74] Plato. *Phaedo*. Harvard University Press ; W. Heinemann, 1967.

[75] Dean A. Pomerleau. ALVINN: An Autonomous Land Vehicle in a Neural Network. In D. S. Touretzky, editor, *Advances in Neural Information Processing Systems*, pages 305–313, 1989.

[76] Senthil Purushwalkam and Abhinav Gupta. Demystifying Contrastive Self-Supervised Learning: Invariances, Augmentations and Dataset Biases. *arXiv:2007.13916 [cs]*, July 2020. arXiv: 2007.13916.

[77] Siddharth Reddy, Anca D. Dragan, and Sergey Levine. SQIL: Imitation Learning via Reinforcement Learning with Sparse Rewards. In *International Conference on Learning Representations*, Sept. 2019.

[78] Nicholas Rhinehart, Rowan Mcallister, Kris Kitani, and Sergey Levine. PRECOG: PREdiction Conditioned on Goals in Visual Multi-Agent Settings. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2821–2830, Oct. 2019. ISSN: 2380-7504.

[79] Daniela Ridel, Eike Rehder, Martin Lauer, Christoph Stiller, and Denis Wolf. A Literature Review on the Prediction of Pedestrian Behavior in Urban Scenarios. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3105–3112, Nov. 2018. ISSN: 2153-0017.

[80] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive Learning with Hard Negative

Samples. *arXiv:2010.04592 [cs, stat]*, Oct. 2020. arXiv: 2010.04592.

[81] Stephane Ross and Drew Bagnell. Efficient Reductions for Imitation Learning. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 661–668. JMLR Workshop and Conference Proceedings, Mar. 2010. ISSN: 1938-7228.

[82] Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, June 2011. ISSN: 1938-7228.

[83] Andrey Rudenko, Luigi Palmieri, Michael Herman, Kris M Kitani, Dariu M Gavrila, and Kai O Arras. Human motion trajectory prediction: a survey. *The International Journal of Robotics Research*, 39(8):895–935, July 2020. Publisher: SAGE Publications Ltd STM.

[84] 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.

[85] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-Feasible Trajectory Forecasting with Heterogeneous Data. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, Lecture Notes in Computer Science, pages 683–700, Cham, 2020. Springer International Publishing.

[86] Samaneh Hosseini Semnani, Hugh Liu, Michael Everett, Anton de Ruiter, and Jonathan P. How. Multi-Agent Motion Planning for Dense and Dynamic Environments via Deep Reinforcement Learning. *IEEE Robotics and Automation Letters*, 5(2):3221–3226, Apr. 2020. Conference Name: IEEE Robotics and Automation Letters.

[87] Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, and Sergey Levine. Time-Contrastive Networks: Self-Supervised Learning from Video. *arXiv:1704.06888 [cs]*, Mar. 2018. arXiv: 1704.06888.

[88] Abhishek Sinha, Kumar Ayush, Jiaming Song, Burak Uzkent, Hongxia Jin, and Stefano Ermon. Negative Data Augmentation. In *International Conference on Learning Representations*, Sept. 2020.

[89] Jiaming Song and Stefano Ermon. Multi-label Contrastive Predictive Coding. *arXiv:2007.09852 [cs, stat]*, July 2020. arXiv: 2007.09852.

[90] Jianhua Sun, Qinhong Jiang, and Cewu Lu. Recursive Social Behavior Graph for Trajectory Prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 660–669, 2020.

[91] L. Sun, Z. Yan, S. M. Mellado, M. Hanheide, and T. Duckett. 3DOF Pedestrian Trajectory Prediction Learned from Long-Term Autonomous Mobile Robot Deployment Data. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5942–5948, May 2018. ISSN: 2577-087X.

[92] Wen Sun, Arun Venkatraman, Geoffrey J. Gordon, Byron Boots, and J. Andrew Bagnell. Deeply AggreVaTeD: Differentiable Imitation Learning for Sequential Prediction. In *International Conference on Machine Learning*, pages 3309–3318. PMLR, July 2017. ISSN: 2640-3498.

[93] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv:1312.6199 [cs]*, Dec. 2013. arXiv: 1312.6199.

[94] Peter Trautman and Andreas Krause. Unfreezing the robot: Navigation in dense, interacting crowds. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 797–803, Oct. 2010. ISSN: 2153-0866.

[95] Jur van den Berg, Stephen J. Guy, Ming Lin, and Dinesh Manocha. Reciprocal n-Body Collision Avoidance. In Cédric Pradalier, Roland Siegwart, and Gerhard Hirzinger, editors, *Robotics Research*, Springer Tracts in Advanced Robotics, pages 3–19, Berlin, Heidelberg, 2011. Springer.

[96] Anirudh Vemula, Katharina Muelling, and Jean Oh. Social Attention: Modeling Attention in Human Crowds. *arXiv:1710.04689 [cs]*, Oct. 2017. arXiv: 1710.04689.

[97] Ruohan Wang, Carlo Ciliberto, Pierluigi Amadori, and Yiannis Demiris. Random Expert Distillation: Imitation Learning via Expert Policy Support Estimation. *arXiv:1905.06750 [cs, stat]*, June 2019. arXiv: 1905.06750.

[98] Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised Feature Learning via Non-parametric Instance Discrimination. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, June 2018. ISSN: 2575-7075.

[99] Francesco Zanlungo, Tetsushi Ikeda, and Takayuki Kanda. Social force model with explicit collision prediction. *EPL (Europhysics Letters)*, 93(6):68005, Mar. 2011. Publisher: IOP Publishing.

[100] Wenyuan Zeng, Shenlong Wang, Renjie Liao, Yun Chen, Bin Yang, and Raquel Urtasun. DSDNet: Deep Structured self-Driving Network. *arXiv:2008.06041 [cs]*, Aug. 2020. arXiv: 2008.06041.

[101] Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *Proceedings of the 23rd national conference on Artificial intelligence - Volume 3*, AAAI’08, pages 1433–1438, Chicago, Illinois, July 2008. AAAI Press.