Quantifying Uncertainty in Motion Prediction with Variational Bayesian Mixture

Juanwu Lu*, Can Cui*, Yunsheng Ma, Aniket Bera, Ziran Wang

Purdue University, West Lafayette, USA

{juanwu, cancui, yunsheng, aniketbera, ziran}@purdue.edu

Abstract

Safety and robustness are crucial factors in developing trustworthy autonomous vehicles. One essential aspect of addressing these factors is to equip vehicles with the capability to predict future trajectories for all moving objects in the surroundings and quantify prediction uncertainties. In this paper, we propose the Sequential Neural Variational Agent (SeNeVA), a generative model that describes the distribution of future trajectories for a single moving object. Our approach can distinguish Out-of-Distribution data while quantifying uncertainty and achieving competitive performance compared to state-of-the-art methods on the Argoverse 2 and INTERACTION datasets. Specifically, a 0.446 meters minimum Final Displacement Error, a 0.203 meters minimum Average Displacement Error, and a 5.35% Miss Rate are achieved on the INTERACTION test set. Extensive qualitative and quantitative analysis is also provided to evaluate the proposed model. Our open-source code is available at https://github.com/PurdueDigitalTwin/seneva.

1. Introduction

Motion prediction is crucial for the safety and robustness of autonomous vehicles (AVs). The objective is to anticipate the potential future movements of the surrounding objects accurately. For the past few years, motion prediction has received emerging interests [19, 27, 28, 30, 31], and we have seen significant progress in prediction accuracy on several benchmarks [4, 40, 44, 45]. However, it remains a challenging task because the behaviors of traffic participants contain inherent multi-modal intentions and uncertainty. Therefore, instead of solely focusing on the prediction accuracy, it is also vital to identify modality and quantify the uncertainty about each predicted trajectory.

Existing methods account for the multi-modality primarily through generating multiple possible trajectories in parallel. Considering the procedure for generating the predic-

---
*Equal Contributions

tions, these methods mainly fall into two categories: sequential models and goal-based models. During inference, a sequential model directly forecasts a collection of possible future trajectories [9, 27]. Despite the prediction accuracy, most models fail to identify intentions and quantify uncertainty about predicted trajectories.

On the contrary, goal-based models generate a set of trajectory endpoint candidates, namely goals, and then complete the intermediate route connecting the object's current location to them [10, 12, 32]. They share a common assumption that these sampled goals are multi-modal and account for most uncertainty. Although one can empirically extrapolate the intention behind each predicted goal, these models often rely on an expressive latent variable space that arbitrarily represents the mixture of all modalities without a specific architecture design that accounts for individual modality. Meanwhile, most also ignore quantifying prediction uncertainties or require recursive sampling to approximate uncertainty in trajectories [1], sharing a limitation similar to the direct-regression models.

This paper addresses these limitations by explicitly modeling the multi-modal trajectory distributions. Specifically, we propose a novel Bayesian mixture model, Sequential Neural Variational Agent (SeNeVA), that treats each observed trajectory in the dataset as being drawn from one of the generating processes. Each process has its own neural network parameterizing the distribution, and all the processes share a common upstream feature encoder. To improve the expressiveness of each process, we introduce a set of latent variables and analytically approximate their posteriors using variational inference. Distribution parameters directly quantify the prediction uncertainties, while the index of the generating process helps identify intention categories. In addition, we train a separate assignment network as a proxy to estimate the posterior distribution of mixture coefficients conditioned solely on the traffic condition. It helps promote generalization ability across different traffic scenarios. At inference time, we select one of the mixture components based on the probability determined by the assignment network and then sample trajectories specific to that particular mode in the scenario. Our experiment re-
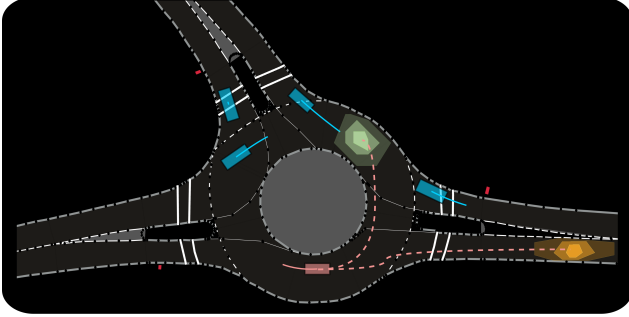
Figure 1. **Illustration of multi-modal trajectory distribution**. There are four surrounding vehicles (blue) and one target vehicle (red) in the scene. The target vehicle can either turn left within the roundabout (green) or turn right into the nearest exit (orange).

sults show that the proposed model achieves competitive prediction accuracy against state-of-the-art methods with extensive information on intention and uncertainty associated with the predictions. The main contributions presented in this paper are:

- We address the limited uncertainty quantification in existing motion prediction models and propose a novel Bayesian mixture method to model the multi-modal distribution of future trajectories conditioned on historical traffic conditions.
- A separate assignment network and an NMS sampling method are introduced to allow for generating a small set of representative trajectories.
- An end-to-end training procedure is developed using variational inference, where the efficiency and robustness of our model are demonstrated through extensive ablations.

2. Related Work

2.1. Generative Motion Prediction Model

Early works using deep neural networks for motion prediction commonly adopted recurrent networks [2, 34, 41] for time-series prediction. However, deterministic predictors have limited capability to capture multi-modal intentions. Recent advancements in generative models [11, 21, 39] have demonstrated their promising power in producing diverse and realistic predictions, enabling a shift from learning a deterministic prediction model to fitting the distribution of all possible future trajectories. Some existing works apply Generative Adversarial Networks (GANs) [11] to fit a pair of generator-discriminator for trajectory prediction [13, 23, 36, 47]. Despite their promising performance, GANs can be unstable during training [22] and lack interpretability of their generating processes. and lack interpretability in their generating processes. Other works adopt Gaussian Mixture Models (GMMs), leveraging the multi-modal property of mixture models [3, 17, 18, 26].

Meanwhile, existing works explore using variational autoencoders (VAEs) for motion prediction [18, 37, 38]. However, these works often directly use distribution means as predictions and neglect quantifying uncertainties in the predictions. Our approach addresses these limitations with a variational Bayes mixture model and investigates the prediction uncertainty quantified by its parameters.

2.2. Uncertainty Quantification

Uncertainty quantification has emerged as an area of interest since learning-based methods can be unreliable when data are out of the distribution of training samples. In autonomous driving, existing works have explored applying uncertainty quantification in object detection [6, 8, 14] to improve perception robustness. Meanwhile, a few works investigate uncertainty quantification in trajectory prediction tasks. Djuric et al. [7] account for the inherent uncertainty of motions in traffic and use CNN in short-term motion prediction. Gaussian Process regression is also an alternative method in motion prediction tasks [42] as it can quantify uncertainty. Wang et al. [43] apply a Bayesian-entropy method considering uncertainty for predicting accurate trajectories and accidents. A comprehensive review of existing works can be found in [1]. Nevertheless, uncertainty quantification in motion prediction remains underexplored.

3. Problem Statement

This paper aims to address the problem of predicting the future trajectories of a single traffic participant in the scene and quantifying prediction uncertainties. The model derives predictions conditioned on the observation history. Suppose the observation horizon is H, and the prediction horizon is T. Given the history motion states $\boldsymbol{s}_{\mathrm{h}}^{(i)} = \left[s_1^{(i)}, s_2^{(i)}, \ldots, s_H^{(i)} \right]$ of a single agent i, the objective is to predict its future motion states $\boldsymbol{s}_{\mathrm{f}}^{(i)} = \left[s_{H+1}^{(i)}, s_{H+2}^{(i)}, \ldots, s_{H+T}^{(i)} \right]$, conditioned on the observation history of its surroundings $\boldsymbol{m}_{\mathrm{h}} = [m_1, m_2, \ldots, m_H]$. To enable uncertainty quantification, instead of learning a model that outputs deterministic predictions, we build our model to estimate the conditional probability $p(\boldsymbol{s}_{\mathrm{f}}^{(i)} | \boldsymbol{m}_{\mathrm{h}}, \boldsymbol{s}_{\mathrm{h}}^{(i)})$.

However, the conditional distribution in real-world cases can be multi-modal. The modes can be spatially disjoint in some circumstances since they reflect mutually exclusive intentions. The example in Figure 1 illustrates such a situation where a vehicle in the roundabout can either keep cruising within the roundabout or head towards the nearest exit. Existing models that generate predictions and uncertainties using features from a unified latent space can derive inaccurate trajectory predictions between any pair of spatially disjoint modes. Therefore, we decompose the condi-
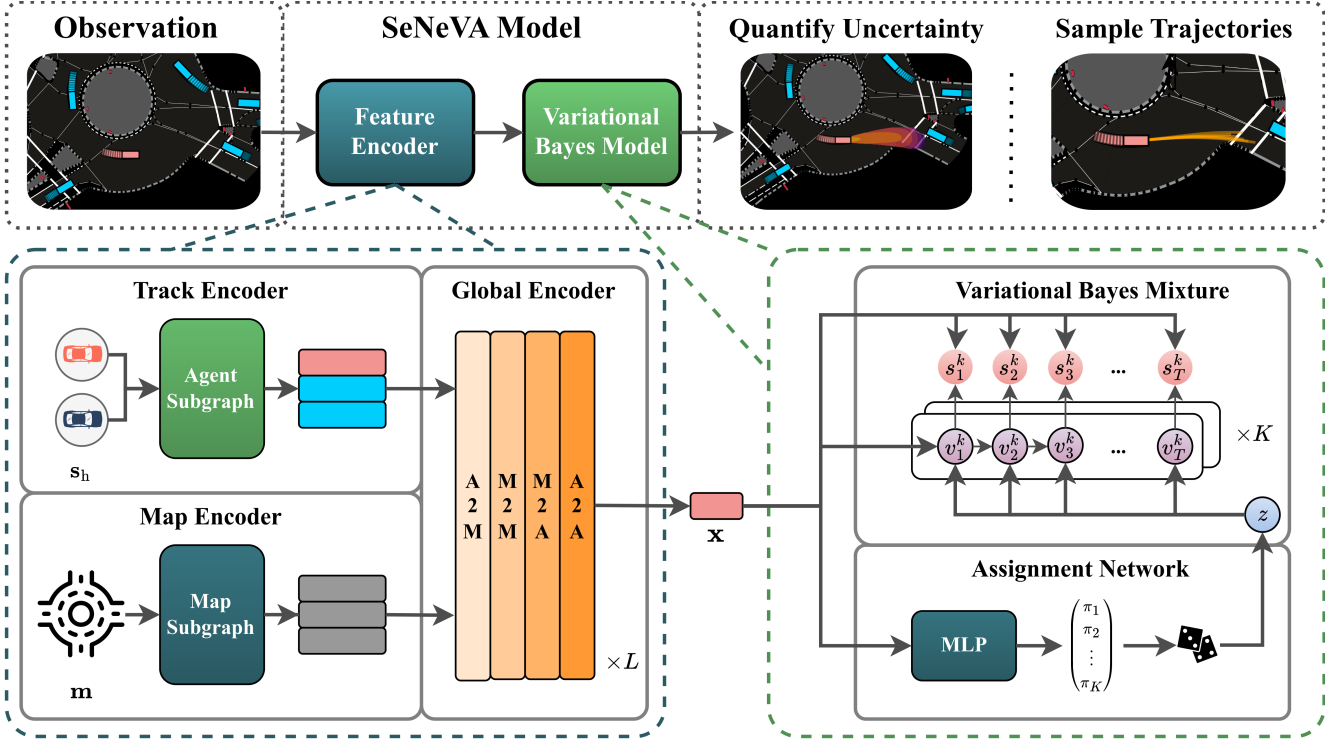
Figure 2. **Architecture of the proposed SeNeVA model**. The track and map encoders encode HD map and agent history trajectories. A global encoder module with a cascade of multi-head attention layers passes messages between map and agents to compute the context feature $\boldsymbol{x}$ from the perspective of the target agent. A variational Bayes Model of K components estimates the distribution $p(\boldsymbol{s}_f|\boldsymbol{x})$ of trajectories conditioned on the context feature x. Additionally, we have an assignment network to estimate the distribution of mixture coefficients $p(z|\boldsymbol{x})$ conditioned on the context feature. The estimated distributions quantify the uncertainty of all possible future trajectories and enable the sampling of representative ones.

tional distribution as a mixture of K disjoint components to promote accurate quantification of the uncertainties:

$$p(\boldsymbol{s}_{\mathrm{f}}^{(i)}|\boldsymbol{m}_{\mathrm{h}},\boldsymbol{s}_{\mathrm{h}}^{(i)}) = \sum_{k=1}^{K} p(\boldsymbol{s}_{\mathrm{f}}^{(i)}|\boldsymbol{m}_{\mathrm{h}},\boldsymbol{s}_{\mathrm{h}}^{(i)},z=k)p(z=k),$$
(1)

where z is an indicator variable for possible futures. We will drop the superscripts i in the following sections to simplify our notation. Following this formulation, we further make the below assumptions:

- The ground-truth trajectory in each training data is a sample from one of the mixture components reflecting its associated intention.
- The residuals concerning the trajectory predictions follow a Multivariate normal distribution in space.
- The displacements in space between consecutive time steps form a temporally correlated time series.

Our proposed model reflects the formulation and the assumptions with three key designs. First, we model each component trajectory distribution as a Multivariate Gaussian; that is, we quantify the uncertainty of each prediction

as the residual. Then, the ground-truth trajectory follows a Bayes Mixture of individual Gaussian distributions with only one of the components activated for each observation. Finally, we introduce an extra latent variable to capture the temporal correlations of the time series.

Compared to other existing methods, our method directly models the entire distribution of all possible futures, providing rich information for downstream tasks such as risk and safety analysis. We incorporate variational inference to train our model. During model inference, the trained probabilistic model supports direct sampling on the mixture distribution using selection methods such as Non-Maximum Suppression [15, 35] for applications requiring only a small set of representative trajectories.

4. Sequential Neural Variational Agent

This section presents the Sequential Neural Variational Agent (SeNeVA) model for single-agent motion prediction (see Figure 2). Our model learns to use encoded features of the traffic environment (Section 4.1) to parameterize a spatial distribution of plausible trajectories as a mixture of Multivariate Gaussian distributions (Section 4.2). In addi-

tion, we implement an assignment network to estimate the mixture weights (Section 4.3), which aims to avoid repeated sampling of latent variables and improve the generalization performance in unseen cases. Finally, we introduce the training objective (Section 4.4) and how we sample from the distribution (Section 4.5).

4.1. Feature Encoding

We represent the history of the surrounding environment and agent motions using a vectorized representation. Specifically, the history motion states of agent i are represented by a vector $s_h^{(i)} \in \mathbb{R}^{H \times 5}$ consisting of the locations, heading, and velocities at each time step. We consider the surrounding to be a static HD map represented by a collection of p polylines $m_h = \{l_1, \ldots, l_p\}$, where each polyline is a set of vectors $l_p \in \mathbb{R}^{N_p \times 4}$, each denoted by the coordinates of its head and tail. All coordinates are projected into a target-centric frame.

We encode the map and agent history using two separate VectorNet subgraphs [9], resulting in a polyline feature p_i for each agent and each polyline on the map. To model high-level interaction, we follow LaneGCN [27] and model four types of global interactions, including agent-to-map-polyline (A2M), map-polyline-to-map-polyline (M2M), map-polyline-to-agent (M2A), and agent-to-agent (A2A) interaction. We use four individual multi-head attention (MHA) layers at each global interaction level to model each type of interaction individually and use a cascade of L-level MHA layers to encode global interactions. The implementation of the encoder module is provided in the supplementary material.

As a result, the output x from the encoder module is a latent representation of the traffic condition from the perspective of the target agent. The consecutive probabilistic model learns the conditional distribution $p(s_f|x, z)$ as the equivalence for $p(s_f|m_h, s_h, z)$ in equation 1.

4.2. Variational Bayes Mixture

Instead of directly predicting a sequence of future locations, our model predicts the displacements between consecutive time steps for stability. We model these displacements as a time series. To capture the temporal dependencies, we introduce a latent variable $v = [v_{H+1}, v_{H+2}, \ldots, v_{H+T}]$ and factorize the conditional distribution as $p(s_f|x, z) = \int_{v,x} p(s_f|v, x) \cdot p(v|x, z)dv$, where

$$p(s_f|v, x) = \prod_{t=1}^{T} p(s_{H+t}|v_{H+t}, x), \quad (2)$$

$$p(v|x, z) = p(v_{H+1}|x, z) \prod_{t=1}^{T-1} p(v_{H+t+1}|v_{H+t}, x). \quad (3)$$

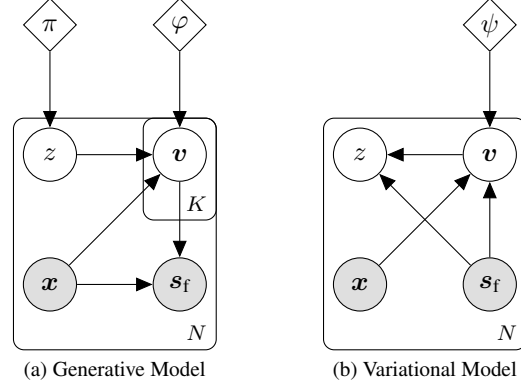(a) Generative Model   (b) Variational Model

Figure 3. The graphical representation of the (a) generative model and the (b) variational family. Shaded and unshaded nodes are the observed and latent random variables. Diamond nodes are the model parameters.

As illustrated by Figure 3, we parameterize the generative process as a conditional variational model given by:

$$p_{\theta,\varphi}(s_f, v, z|x) = p_\theta(s_f|v, x) \cdot p_\varphi(v|x, z) \cdot p(z), \quad (4)$$

where we consider the latent variables to follow the below processes:

$$z \sim \text{Categorical}(\pi), \quad (5a)$$

$$v_{H+1}|x, z \sim \prod_{k=1}^{K} \mathcal{N}\left(\mu(x; \varphi_{0,k}), \text{diag}(\sigma^2(x; \varphi_{0,k}))\right)^{z_k}, \quad (5b)$$

$$v_{t+1}|v_t, x, z \sim \prod_{k=1}^{K} \mathcal{N}\left(\mu(x; \varphi_{r,k}), \text{diag}(\sigma^2(x; \varphi_{r,k}))\right)^{z_k}, \quad (5c)$$

$$s_t|v_t, x \sim \mathcal{N}\left(\mu([v_t, x]; \theta), \Sigma([v_t, x]; \theta)\right). \quad (5d)$$

The $\mu(\cdot; \theta), \Sigma(\cdot; \theta), \mu(\cdot; \varphi_*)$, and $\Sigma(\cdot; \varphi_*)$ above are outputs from neural networks with parameters θ and φ_*. The $[\cdot, \cdot]$ denotes concatenation operation. The diag$[\cdot]$ is the operation to create a diagonal matrix with the given values. In practice, θ and $\varphi_{0,k}$ are parameters of a Multi-layer Perceptron (MLP), and $\varphi_{r,k}$ are parameters of a Long Short-Term Memory (LSTM) [33]. We use a non-informative, uniform prior $\pi_k = 1/K, k = 1, \ldots, K$ for the random variable z since we assume the dataset covers all possible intentions equally. One of the key advantages of our formulation is that the use of LSTM cells and the MLP in equation 5d promotes parameter efficiency and enables the handling of variable-length trajectory predictions

However, fitting the generative model by directly maximizing the log-likelihood $\log p(\boldsymbol{s}_{\mathrm{h}}|\boldsymbol{x})$ is intractable. We approach this problem using variational inference. Specifically, we use the mean-field variational family to approximate the true posterior, given as:

$$q(\boldsymbol{v}, z|\boldsymbol{s}_{\mathrm{f}}, \boldsymbol{x}) = q_{\psi}(\boldsymbol{v}|\boldsymbol{s}_{\mathrm{f}}, \boldsymbol{x}) \cdot q_{\varphi}(z|\boldsymbol{v}, \boldsymbol{x}), \qquad (6)$$

Similar to the generative model, we factorize the sequential dependency of $q_{\psi}(\boldsymbol{v}|\boldsymbol{x}, \boldsymbol{s}_{\mathrm{f}})$ and parameterize it using a combination of MLP and LSTM. The factorization structure of $q_{\psi}(\boldsymbol{v}|\boldsymbol{x}, \boldsymbol{s}_{\mathrm{f}})$ resembles the one in equation 3, given as:

$$q_{\psi}(\boldsymbol{v}|\boldsymbol{x}, \boldsymbol{s}_f) = q(v_{H+1}|\boldsymbol{x}, \boldsymbol{s}_{\mathrm{f}}) \prod_{t=1}^{T-1} q(v_{H+t+1}|v_{H+t}, \boldsymbol{x}, \boldsymbol{s}_{\mathrm{f}})$$
$$(7)$$

Meanwhile, we reparameterize the posterior for the random variable z by:

$$q_{\varphi}(z = j|\boldsymbol{v}, \boldsymbol{x}) = \frac{p(z = j)p_{\varphi}(\boldsymbol{v}|\boldsymbol{x}, z = j)}{\sum\limits_{k=1}^{K} p(z = k)p_{\varphi}(\boldsymbol{v}|\boldsymbol{x}, z = k)} \qquad (8)$$

This avoids an additional $q(z)$ network while allowing gradients to backpropagate onto the parameter set φ. The derivation is in the supplementary. Nevertheless, the trade-off is that we need further Monte-Carlo sampling during inference to estimate the z-posterior. The following section introduces our solution to this issue with a proxy network to directly output $p(z|x)$.

4.3. Assignment Network

Sampling from the latent space to estimate the posterior assignment $q(z|\boldsymbol{v}^{(i)}, \boldsymbol{x})$ can be computationally intensive. We introduce an assignment network that approximates $p(z|x)$ conditioned solely on the input feature x to address this. The network is parameterized as an MLP that outputs the log-likelihood of mixture components:

$$\log \hat{\pi} = \mathrm{MLP}(\boldsymbol{x}). \qquad (9)$$

The estimated $\hat{\pi}$ can be used as the mixture coefficient for uncertainty quantification during inference. For evaluations based on a small set of trajectories, the assignment network can help identify the components most likely to be sampled instead of uniformly sampling from all the mixture components. This approach can significantly reduce computational complexity while maintaining the quality of the generated trajectories.

4.4. Model Training

We train our variational Bayes model to maximize the Evidence Lower Bound (ELBO). Using the factorizations given

by equation 4 and equation 6, the ELBO objective can be written as:

$$\begin{aligned}
\mathcal{L}_{\mathrm{ELBO}} = {} & \mathbb{E}_{q_{\psi}(\boldsymbol{v}|\boldsymbol{s}_{\mathrm{f}}, \boldsymbol{x})} \log p_{\theta}(\boldsymbol{s}_{\mathrm{f}}|\boldsymbol{v}, \boldsymbol{x}) \\
& - \mathbb{E}_{q_{\varphi}(\boldsymbol{z}|\boldsymbol{v}, \boldsymbol{x})} D_{\mathrm{KL}}\left(q_{\psi}(\boldsymbol{v}|\boldsymbol{s}_{\mathrm{f}}, \boldsymbol{x})\|p_{\varphi}(\boldsymbol{v}|\boldsymbol{x}, z)\right) \quad (10) \\
& - \mathbb{E}_{q_{\psi}(\boldsymbol{v}|\boldsymbol{s}_{\mathrm{f}}, \boldsymbol{x})} D_{\mathrm{KL}}\left(q_{\varphi}(z|\boldsymbol{v}, \boldsymbol{x})\|p(z)\right),
\end{aligned}$$

where $D_{\mathrm{KL}}(\cdot\|\cdot)$ denotes the Kullback-Leibler divergence [24]. We apply Monte-Carlo sampling to estimate the expectations in the ELBO. Since we reparameterize the z-posterior in equation 8, we can directly compute the second term in the ELBO using the Monte-Carlo samples. For each sample j, we define $w_{jk} = q_{\varphi}(z = k|\boldsymbol{v}^{(j)}, \boldsymbol{x})$ and $d_{jk} = D_{\mathrm{KL}}\left(q_{\psi}(\boldsymbol{v}^{(j)}|\boldsymbol{s}_{\mathrm{f}}, \boldsymbol{x})\|p_{\varphi}(\boldsymbol{v}^{(j)}|\boldsymbol{x}, z = k)\right)$, then

$$\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{v}, \boldsymbol{x})} D_{\mathrm{KL}}\left(q(\boldsymbol{v}|\boldsymbol{s}_{\mathrm{f}}, \boldsymbol{x})\|p(\boldsymbol{v}|\boldsymbol{x}, z)\right) \approx \frac{1}{N_{\mathrm{mc}}} \sum_{j=1}^{N_{\mathrm{mc}}} \sum_{k=1}^{K} w_{jk} \cdot d_{jk},$$
$$(11)$$

where N_{mc} is the number of Monte-Carlo samples. A detailed derivation of the ELBO is provided in the supplementary material.

To train the assignment network, we first marginalize the ground-truth assignment weights of the mixture components given the motion states $\boldsymbol{s}_f$ by:

$$p(\boldsymbol{s}_{\mathrm{f}}|\boldsymbol{x}, z) = \int_{\boldsymbol{v} \sim q(\boldsymbol{v}|\boldsymbol{s}_f, \boldsymbol{x})} p(\boldsymbol{s}_f|\boldsymbol{v}, \boldsymbol{x})p(\boldsymbol{v}|\boldsymbol{x}, z)d\boldsymbol{v}. \quad (12)$$

We approximate the marginalization by applying Monte-Carlo sampling on the posterior distribution $q(\boldsymbol{v}|\boldsymbol{s}_f, \boldsymbol{x})$. Since we are using a non-informative uniform prior distribution for z, we can obtain the target assignment weights by applying Bayes' rule:

$$p(z = j|\boldsymbol{x}) = \frac{p(\boldsymbol{s}_f|\boldsymbol{x}, z = j)p(z = j)}{p(\boldsymbol{s}_{\mathrm{f}}|\boldsymbol{x})} = \frac{p(\boldsymbol{s}_f|\boldsymbol{x}, z = j)}{\sum\limits_{k=1}^{K} p(\boldsymbol{s}_{\mathrm{f}}|\boldsymbol{x}, z = k)}.$$
$$(13)$$

We train the assignment network to minimize the focal loss [29] given by:

$$\mathcal{L}_{\hat{\pi}} = -\sum_{k=1}^{K} (1 - \hat{\pi}_k)^{\gamma} \cdot p(z = k|\boldsymbol{x}) \log \hat{\pi}_k, \qquad (14)$$

where γ is a tunable focusing hyperparameter balancing well-classified and misclassified components. Overall, we train the SeNeVA model to minimize the weighted sum of two losses:

$$\mathcal{L} = -\mathcal{L}_{\mathrm{ELBO}} + \alpha \mathcal{L}_{\hat{\pi}}. \qquad (15)$$

Algorithm 1 Destination Sampling

Require: List of candidates $\{y_{H+T}\}$, Candidate buffer radius r, Intersection-over-Union (IoU) Threshold γ, Number of trajectories to sample M
1: Evaluate probabilities $p(y_{H+T}|v_{\leq T}, \boldsymbol{x}, z)$
2: Sort candidates by probabilities in a descending order
3: Initialize an empty list Q
4: **while** size$(Q) < M$ **do**
5:     Take the most probable candidate y_{H+T}^*
6:     Add y_{H+T}^* to Q
7:     Create circle c centered at y_T^* of radius r
8:     **for** each other candidate y'_{H+T} **do**
9:         Create circle c' centered at y'_{H+T} of radius r
10:        **if** IoU$(c, c') > \gamma$ **then**
11:            Remove y'_{H+T} from the list of candidates
12:        **end if**
13:    **end for**
14:    Remove y_{H+T}^* from the list of candidates
15: **end while**
16: **return** Q

4.5. Trajectory Sampling

The output from the SeNeVA model is the distribution of all possible trajectories in the future. However, many existing motion prediction challenges and applications require only a small set of the most probable predictions for evaluation. To promote the leverage of the distribution information, we propose a method to sample from the generative model with Non-Maximum Suppression (NMS). Denote $y_t = \sum_{t'=1}^{t} s_t$ as the location of the target agent at time t. Since s_t are Gaussian random variables, the sum of them also follows a Gaussian distribution:

$$y_t|v_{\leq t}, \boldsymbol{x}, z \sim \mathcal{N}\left(\sum_{t'=1}^{t} \mu([v_t, \boldsymbol{x}]; \theta), \sum_{t'=1}^{t} \Sigma([v_t, \boldsymbol{x}]; \theta)\right). \tag{16}$$

During sampling, we obtain the mixture weights approximation $\hat{\pi}$ from the assignment network to help determine which component we should sample from. Since the final location y_{H+T} holds the most uncertainty, quantified by its covariance matrix as a summation of displacement covariance over all previous time steps, we propose first sampling y_{H+T} using NMS, as described in Algorithm 1. For each sampled y_{H+T}, we generate the intermediate path from the target agent's current location y_H to the sampled final destination y_{H+T} by assuming uniform uncertainty over time. This assumption helps guarantee the smoothness of the sampled trajectories. Details are provided in the supplementary.

5. Experiments

5.1. Experiment Setup

Datasets The evaluation is conducted on two benchmark datasets: the INTERACTION [45] and the Argoverse 2 [44] dataset. The INTERACTION dataset consists of data collected from 18 different locations globally. The goal is to predict the 3-second future conditioned on a 1-second history observation. There are about 35% Out-of-Distribution (OOD) data in the test dataset (i.e., with unseen map and traffic conditions). This is an interesting feature we leverage to analyze the uncertainty quantification performance of our proposed SeNeVA model. The Argoverse 2 dataset contains 250,000 scenarios collected from 6 different cities. The task is to predict a future 6-second trajectory based on a 5-second history observation.

Metrics For uncertainty quantification, we evaluate the OOD identification capability of SeNeVA by comparing the predicted distribution entropy under in-distribution and OOD cases. For predicting a small set of representative trajectories, we use standard evaluation metrics, including minimum Average Displacement Error (minADE$_k$), minimum Final Displacement Error (minFDE$_k$), and Miss Rate (MR$_k$). We present further details about the metrics in the supplementary material.

5.2. Uncertainty Quantification

5.2.1 Quantitative Analysis

The output trajectories from the SeNeVA model follow a bivariate Gaussian distribution, allowing the total uncertainty about the prediction to be directly measured by its entropy (calculation details in supplementary). We expect the SeNeVA model to have lower entropy for in-distribution cases and higher entropy for out-of-distribution (OOD) ones.

We compute the distribution entropy for the 22,498 cases of the INTERACTION test dataset, where 7,898 data cases are from unseen locations. Figure 4 shows that the predicted total uncertainty in OOD cases is generally higher than in in-distribution data, with a 1.5% increase in highway merging cases and 0.6% increase in roundabout cases. This demonstrates SeNeVA's ability to distinguish OOD data during prediction and assign higher uncertainty to these cases. However, the OOD entropy in intersection cases is 0.5% lower than in-distribution cases, possibly due to the more complex traffic conditions in intersections, including cyclists and pedestrians.

We further compare our model with the Deep Ensembles method [25] for uncertainty quantification, using VectorNet as the base ensembled model for fair comparison. Table 2 shows that SeNeVA can better distinguish OOD data, with

Table 1. Comparison of the proposed SeNeVA with various state-of-the-art methods on the two datasets. The top-performing method for each setting is highlighted in **bold**. The second-top-performing method is highlighted with an underscore (_).

| Dataset	Split	Method	# Param.	minFDE$_6$ ($\downarrow$)	minADE$_6$ ($\downarrow$)	MR ($\downarrow$)
INTERACTION	test	DenseTNT [12]	-	0.795	0.424	0.060
		Multi-Branch SS-ASP [19]	-	0.539	<u>0.178</u>	0.115
		MultiModalTransformer [16]	6.3M	0.551	0.213	**0.051**
		HDGT [20]	15.3M	<u>0.478</u>	**0.168**	0.056
		SeNeVA (ours)	**1.3M**	**0.446**	0.203	<u>0.053</u>
INTERACTION	val	DESIRE [26]	-	0.880	0.320	-
		MultiPath [3]	-	0.990	0.300	-
		TNT [46]	-	<u>0.670</u>	<u>0.210</u>	-
		SeNeVA (ours)	**1.3M**	**0.431**	**0.197**	**0.079**
Argoverse 2	val	DenseTNT [12]	-	1.620	0.960	0.233
		Forecast-MAE [5]	1.9M	<u>1.409</u>	<u>0.901</u>	<u>0.178</u>
		SeNeVA (ours)	**1.3M**	**1.319**	**0.713**	**0.175**
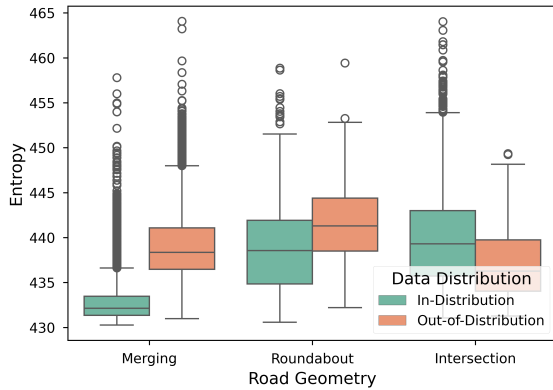
Figure 4. **Predicted uncertainty in different road geometry and data distributions.** The predicted uncertainty in OOD cases is generally higher than in in-distribution cases.

Table 2. Predictive uncertainty measured by total entropy in in-distribution (ID) and out-of-distribution (OOD) cases.

Model	Geometry	Entropy	
		ID	OOD
VectorNet Ensembles	Merging	90.98 (5.26)	87.69 (8.97)
	Roundabout	95.59 (7.63)	104.62 (8.39)
	Intersection	92.73 (6.81)	76.37 (5.35)
SeNeVA (Ours)	Merging	422.64 (2.64)	429.28 (4.32)
	Roundabout	428.12 (4.63)	431.17 (4.47)
	Intersection	429.60 (5.63)	427.13 (4.33)

an average inference time of 51ms compared to $1,590$ms for the Deep Ensembles model, indicating better efficiency.

5.2.2 Qualitative Analysis

In Figure 5, we visualize the quantified uncertainty in an in-distribution and an OOD case side-by-side. The results show that the SeNeVA model can predict a distribution that conforms well to the road geometry even in both in-distribution and OOD cases. The likelihood of a location being visited in the future gradually decreases with respect to the distance, indicating an increased uncertainty in prediction.

5.3. Motion Prediction

We comprehensively evaluate our SeNeVA model on the INTERACTION and Argoverse 2 dataset motion prediction tasks, comparing its performance with several state-of-the-art motion prediction algorithms. Our results are presented in Table 1, demonstrating that the SeNeVA model consistently outperforms other models in the literature. On the INTERACTION dataset, we report our results on the online INTERACTION test set. Our SeNeVA model has demonstrated a remarkable enhancement in minFDE$_6$, outperforming other baseline models by at least 6.8%. Regarding minADE$_6$, our SeNeVA model closely approaches the state-of-the-art performance, achieving nearly identical results. Regarding MR$_6$, SeNeVA ranks as the second-best performing algorithm, with only a marginal 0.21% difference from the top-performing result. On the INTERACTION val set, all of the metrics outperform the other baselines, with our model achieving a remarkable 35.7% improvement over the second-best results in minFDE$_6$ and a substantial 7.0% lead in minADE$_6$. Similarly, on the Argoverse 2 validation set, SeNeVA consistently demonstrates competitive results. It achieves a significant 6.8% lead in minFDE$_6$ and a remarkable 26.0% improvement in

(a) In-Distribution Case (DR_USA_Roundabout_SR)   (b) Out-of-Distribution Case (DR_Roundabout_RW)
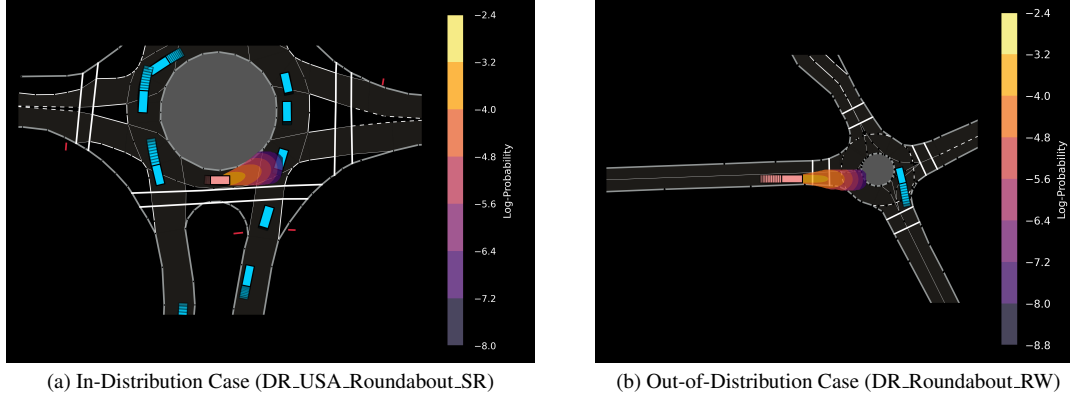
Figure 5. **Example visualization of quantified uncertainty.** We visualize the predicted uncertainty in an in-distribution test case (**left**) and an OOD test case (**right**). The heatmap reflects the log-likelihood of a location on the map being visited in the future.

$minADE_6$. Additionally, our model's consistently superior performance across various datasets highlights its reliability in handling diverse different scenarios, enhancing safety and efficiency. Our model's lightweight design with just 1.3 million parameters, shows its efficiency in computational resources and deployment.

5.4. Ablation Study

We conduct ablation studies to assess the influence of different modules. These experiments are carried out using the validation split of the INTERACTION dataset.

Number of Mixtures We compare prediction performance varying number of mixtures $N_{components}$ in our model. As shown in Table 3, we discover that the optimal choice for the number of mixtures is 6. We argue that a lower $N_{components}$ can limit the expressiveness for multi-modality, while a higher value can lead to a risk of excessive complexity that prevents effective learning. In particular, it affects training the assignment network since the similarities among mixture components increase with increasing mixtures, making it hard for the assignment network to distinguish different distributions.

Assignment Network and NMS Sampling Method In Table 4, we analyze the effect of assignment network (AN) and the sampling method (NMS). The results show that the sampling guided by the assignment network can improve the overall performance. Applying NMS sampling can reduce the MR at the cost of increasing minADE.

6. Conclusion

This paper introduces SeNeVA, a novel variational Bayes model for uncertainty quantification in motion prediction. An assignment network and an NMS-based trajectory sampling are introduced to support use cases requiring

Table 3. Ablation study investigating the influence of the number of mixture components $N_{components}$ in the SeNeVA model.

$N_{components}$	$minFDE_6$	$minADE_6$	MR
4	0.5362	0.4418	0.1065
6	0.4306	0.1967	0.0790
16	0.5352	0.2030	0.1280
32	0.5504	0.2115	0.1316

Table 4. The performance of SeNeVA on the validation set with and without the inclusion of the assignment network (AN) and NMS sampling method (NMS).

Module			$minFDE_6$	$minADE_6$	MR
Means	AN	NMS			
✓			0.5135	0.2281	0.083
✓	✓		0.4306	0.1967	0.079
	✓	✓	0.4265	0.2186	0.073

only representative trajectories. Experiments demonstrate SeNeVA's ability to distinguish in-distribution and OOD data by quantifying uncertainty while performing comparative motion prediction to state-of-the-art methods.

Limitations. SeNeVA requires target-centric inputs and only predicts the distribution of future trajectories for a single agent at a time. Predictions in large-scale traffic scenarios may require parallel inference of multiple models, which can be computationally expensive. Therefore, predicting the joint distribution of multiple agents can be an important future topic to explore.

References

[1] Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U. Rajendra Acharya, Vladimir Makarenkov, and Saeid Nahavandi. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76: 243–297, 2021. 1, 2

[2] 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. 2

[3] Yuning Chai, Benjamin Sapp, Mayank Bansal, and Dragomir Anguelov. Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction. In *Proceedings of the Conference on Robot Learning*, pages 86–99. PMLR, 2020. 2, 7

[4] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, et al. Argoverse: 3d tracking and forecasting with rich maps. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8748–8757, 2019. 1

[5] Jie Cheng, Xiaodong Mei, and Ming Liu. Forecast-mae: Self-supervised pre-training for motion forecasting with masked autoencoders. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8679–8689, 2023. 7

[6] Hsu-kuang Chiu, Jie Li, Rareş Ambruş, and Jeannette Bohg. Probabilistic 3d multi-modal, multi-object tracking for autonomous driving. In *2021 IEEE international conference on robotics and automation (ICRA)*, pages 14227–14233. IEEE, 2021. 2

[7] Nemanja Djuric, Vladan Radosavljevic, Henggang Cui, Thi Nguyen, Fang-Chieh Chou, Tsung-Han Lin, NITIN SINGH, and Jeff Schneider. Uncertainty-aware short-term motion prediction of traffic actors for autonomous driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2020. 2

[8] Di Feng, Ali Harakeh, Steven L Waslander, and Klaus Dietmayer. A review and comparative study on probabilistic object detection in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):9961–9980, 2021. 2

[9] 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 (CVPR)*, 2020. 1, 4

[10] Thomas Gilles, Stefano Sabatini, Dzmitry Tsishkou, Bogdan Stanciulescu, and Fabien Moutarde. GOHOME: Graph-Oriented Heatmap Output for future Motion Estimation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 9107–9114, Philadelphia, PA, USA, 2022. IEEE. 1

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 2

[12] Junru Gu, Chen Sun, and Hang Zhao. Densetnt: End-to-end trajectory prediction from dense goal sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15303–15312, 2021. 1, 7

[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. 2

[14] David Hall, Feras Dayoub, John Skinner, Haoyang Zhang, Dimity Miller, Peter Corke, Gustavo Carneiro, Anelia Angelova, and Niko Sünderhauf. Probabilistic object detection: Definition and evaluation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1031–1040, 2020. 2

[15] Jan Hosang, Rodrigo Benenson, and Bernt Schiele. Learning non-maximum suppression. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4507–4515, 2017. 3

[16] Zhiyu Huang, Xiaoyu Mo, and Chen Lv. Multi-modal Motion Prediction with Transformer-based Neural Network for Autonomous Driving. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2605–2611, Philadelphia, PA, USA, 2022. IEEE. 7

[17] Boris Ivanovic and Marco Pavone. The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2375–2384, 2019. 2

[18] Boris Ivanovic, Edward Schmerling, Karen Leung, and Marco Pavone. Generative modeling of multimodal multi-human behavior. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3088–3095. IEEE, 2018. 2

[19] Faris Janjoš, Max Keller, Maxim Dolgov, and J. Marius Zöllner. Bridging the Gap Between Multi-Step and One-Shot Trajectory Prediction via Self-Supervision. In *2023 IEEE Intelligent Vehicles Symposium (IV)*, pages 1–8, Anchorage, AK, USA, 2023. IEEE. 1, 7

[20] Xiaosong Jia, Penghao Wu, Li Chen, Hongyang Li, Yu Liu, and Junchi Yan. HDGT: Heterogeneous Driving Graph Transformer for Multi-Agent Trajectory Prediction via Scene Encoding, 2022. arXiv:2205.09753 [cs]. 7

[21] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. 2

[22] Naveen Kodali, Jacob Abernethy, James Hays, and Zsolt Kira. On convergence and stability of gans, 2017. 2

[23] 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, 2019. 2

[24] S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79 – 86, 1951. 5

[25] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 6

[26] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B. Choy, Philip H. S. Torr, and Manmohan Chandraker. DESIRE: Distant Future Prediction in Dynamic Scenes with Interacting Agents. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2165–2174, Honolulu, HI, 2017. IEEE. 2, 7

[27] Ming Liang, Bin Yang, Rui Hu, Yun Chen, Renjie Liao, Song Feng, and Raquel Urtasun. Learning lane graph representations for motion forecasting. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 541–556. Springer, 2020. 1, 4

[28] Xishun Liao, Ziran Wang, Xuanpeng Zhao, Zhouqiao Zhao, Kyungtae Han, Prashant Tiwari, Matthew J Barth, and Guoyuan Wu. Online prediction of lane change with a hierarchical learning-based approach. In *2022 IEEE International Conference on Robotics and Automation (ICRA)*, 2022. 1

[29] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 5

[30] Yicheng Liu, Jinghuai Zhang, Liangji Fang, Qinhong Jiang, and Bolei Zhou. Multimodal Motion Prediction with Stacked Transformers. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7573–7582, Nashville, TN, USA, 2021. IEEE. 1

[31] Yongkang Liu, Ziran Wang, Kyungtae Han, Zhenyu Shou, Prashant Tiwari, and John Hansen. Vision-cloud data fusion for adas: A lane change prediction case study. *IEEE Transactions on Intelligent Vehicles*, 7(2):210–220, 2022. 1

[32] Juanwu Lu, Wei Zhan, Masayoshi Tomizuka, and Yeping Hu. Towards generalizable and interpretable motion prediction: A deep variational bayes approach, 2024. 1

[33] Long Short-Term Memory. Long short-term memory. *Neural computation*, 9(8):1735–1780, 2010. 4

[34] Jeremy Morton, Tim A Wheeler, and Mykel J Kochenderfer. Analysis of recurrent neural networks for probabilistic modeling of driver behavior. *IEEE Transactions on Intelligent Transportation Systems*, 18(5):1289–1298, 2016. 2

[35] Alexander Neubeck and Luc Van Gool. Efficient non-maximum suppression. In *18th international conference on pattern recognition (ICPR'06)*, pages 850–855. IEEE, 2006. 3

[36] 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. 2

[37] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16*, pages 683–700. Springer, 2020. 2

[38] Edward Schmerling, Karen Leung, Wolf Vollprecht, and Marco Pavone. Multimodal probabilistic model-based planning for human-robot interaction. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3399–3406. IEEE, 2018. 2

[39] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28, 2015. 2

[40] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020. 1

[41] 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. 2

[42] Jack M. Wang, David J. Fleet, and Aaron Hertzmann. Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):283–298, 2008. 2

[43] Yuhao Wang, Yutian Pang, Oliver Chen, Hari N. Iyer, Parikshit Dutta, P.K. Menon, and Yongming Liu. Uncertainty quantification and reduction in aircraft trajectory prediction using bayesian-entropy information fusion. *Reliability Engineering & System Safety*, 212:107650, 2021. 2

[44] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting, 2023. 1, 6

[45] Wei Zhan, Liting Sun, Di Wang, Haojie Shi, Aubrey Clausse, Maximilian Naumann, Julius Kummerle, Hendrik Konigshof, Christoph Stiller, Arnaud de La Fortelle, and Masayoshi Tomizuka. Interaction dataset: An international, adversarial and cooperative motion dataset in interactive driving scenarios with semantic maps, 2019. 1, 6

[46] Hang Zhao, Jiyang Gao, Tian Lan, Chen Sun, Ben Sapp, Balakrishnan Varadarajan, Yue Shen, Yi Shen, Yuning Chai, Cordelia Schmid, Congcong Li, and Dragomir Anguelov. Tnt: Target-driven trajectory prediction. In *Proceedings of the 2020 Conference on Robot Learning*, pages 895–904. PMLR, 2021. 7

[47] 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