

An Ensemble Learning Framework for Vehicle Trajectory Prediction in Interactive Scenarios

Li, Zirui; Lin, Yunlong ; Cheng, Gong; Wang, Xinwei; Liu, Qi; Gong, Jianwei; Lu, Chao

DOI

[10.1109/IV51971.2022.9827070](https://doi.org/10.1109/IV51971.2022.9827070)

Publication date

2022

Document Version

Accepted author manuscript

Published in

Proceedings of the 2022 IEEE Intelligent Vehicles Symposium, IV 2022

Citation (APA)

Li, Z., Lin, Y., Cheng, G., Wang, X., Liu, Q., Gong, J., & Lu, C. (2022). An Ensemble Learning Framework for Vehicle Trajectory Prediction in Interactive Scenarios. In *Proceedings of the 2022 IEEE Intelligent Vehicles Symposium, IV 2022* (pp. 51-57). Article 9827070 IEEE.
<https://doi.org/10.1109/IV51971.2022.9827070>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

An Ensemble Learning Framework for Vehicle Trajectory Prediction in Interactive Scenarios*

Zirui Li^{1,2,3}, Yunlong Lin^{1,2}, Cheng Gong², *Student Member IEEE*,
Xinwei Wang³, Qi Liu², Jianwei Gong², *Member IEEE*, Chao Lu², *Member IEEE*.

Abstract—Precisely modeling interactions and accurately predicting trajectories of surrounding vehicles are essential to the decision-making and path-planning of intelligent vehicles. This paper proposes a novel framework based on ensemble learning to improve the performance of trajectory predictions in interactive scenarios. The framework is termed Interactive Ensemble Trajectory Predictor (IETP). IETP assembles interaction-aware trajectory predictors as base learners to build an ensemble learner. Firstly, each base learner in IETP observes historical trajectories of vehicles in the scene. Then each base learner handles interactions between vehicles to predict trajectories. Finally, an ensemble learner is built to predict trajectories by applying two ensemble strategies on the predictions from all base learners. Predictions generated by the ensemble learner are final outputs of IETP. In this study, three experiments using different data are conducted based on the NGSIM dataset. Experimental results show that IETP improves the predicting accuracy and decreases the variance of errors compared to base learners. In addition, IETP exceeds baseline models with 50% of the training data, indicating that IETP is data-efficient. Moreover, the implementation of IETP is publicly available at <https://github.com/BIT-Jack/IETP>.

I. INTRODUCTION

Developing intelligent vehicles with socially compliant and conventional driving behaviors is significant to traffic safety and road mobility [1]–[3]. In interactive scenarios such as urban roads and freeways, the dynamic motions of surrounding vehicles limit the availability of some driving actions. Thus, intelligent vehicles need to predict future trajectories of surrounding vehicles before making decisions, and planning paths [4]. Accurately predicting the future trajectories of surrounding vehicles becomes a fundamental ability of intelligent vehicles. However, predicting trajectories in interactive scenarios is challenging due to the complexity and uncertainty of interactions between vehicles. For example, various drivers' driving styles and different destinations of vehicles reveal the complexity. The randomness of driving

behaviors indicates the uncertainty [5]. Therefore, how to precisely model interactions between vehicles becomes a crucial problem of predicting vehicle trajectories in interactive scenarios.

Compared to traditional trajectory predicting methods such as Kalman filter [6] and social force model [7], deep learning-based methods have shown outstanding performance to model interactions between traffic agents such as vehicles and pedestrians [8]. In deep learning, a type of neural network termed Long Short-Term Memory (LSTM) has been shown to successfully handle tasks with sequential inputs and outputs. Since trajectory predictions can be viewed as such sequential to sequential tasks, many LSTM-based methods are proposed for trajectory predictions in interactive scenarios.

LSTM-based methods [9]–[11] utilize the pooling mechanism to model social interactions. In [9], social pooling layers are proposed to capture interactions among pedestrians. The spatial information of pedestrians is preserved through grid-based pooling. [10] extends the Social-LSTM [9] by applying convolutional layers to replace fully connected layers, which is proposed as a remedy to address generalization problems. [12]–[14] adopt the attention mechanism to deal with interactive information. [12] proposes a state refinement module for the LSTM network. Trajectories are predicted by utilizing the current intention of surrounding agents. In [14], a Social Relationship Attention LSTM (SRA-LSTM) is proposed to predict future trajectories. Social relationship attention to aggregate movement information from neighbor agents is utilized to model the interactions in SRA-LSTM. These LSTM-based methods focus on designing special mechanisms to model interactions. The mainly used networks are LSTM. Furthermore, more types of networks such as Generative Adversarial Networks (GAN), Graph-based networks, and Temporal Convolutional Networks (TCN) are utilized to improve the modeling and predicting performance.

In [15], tools from sequence prediction and generative adversarial networks are combined to predict socially plausible trajectories. [16] improves the Social-GAN model by introducing a flexible graph attention network. [17] extracts interactive information into social behavior graphs. The graph convolutional neural network is then applied to propagate social interaction information in such graphs. [18] proposes a graph-based information-sharing network (GIS-Net) to improve the accuracy of vehicle trajectory prediction compared to baselines in experiments. [19] proposes a graph-based temporal convolutional network. Both the accuracy and

¹Zirui Li and Yunlong Lin contribute equally.

*This work was supported by the National Natural Science Foundation of China under Grants 61703041 and U19A2083.

²Zirui Li, Yunlong Lin, Cheng Gong, Qi Liu, Jianwei Gong and Chao Lu are with the School of Mechanical Engineering, Beijing Institute of Technology, Beijing 100081, China.(E-mails: ziruili.work.bit@gmail.com; 1120181526@bit.edu.cn; chenggong@bit.edu.cn; 3120195257@bit.edu.cn; gongjianwei@bit.edu.cn; chaolu@bit.edu.cn)

³Zirui Li and Xinwei Wang are with the Department of Transport and Planning, Faculty of Civil Engineering and Geosciences, Delft University of Technology, Stevinweg 1, 2628 CN Delft, The Netherlands(E-mails: x.w.wang@tudelft.nl)

(Corresponding author: J. Gong)

efficiency of trajectory predictions are improved compared to baselines in experiments. [20] proposes a hierarchical Graph Neural Network (GNN) framework combined with LSTM to model interactions of heterogeneous traffic participants and predict their trajectories. Similarly, in [21], by using GNN, interactions and trajectories are firstly modeled as spatial-temporal graphs. Then a GNN-based multitask learning framework is proposed to accurately predict trajectories of vehicles and pedestrians.

However, these previous studies only focus on the design of network structures to model interactions, and each model is used individually to predict trajectories. The combination of different models is ignored. Using an individual model to make predictions may lead to low predicting accuracy and the model can be sensitive to training data [22].

As discussed in [23] and [24], the individual model can be weak to make predictions. In comparison, ensemble learning methods that assemble individual models to make predictions can improve the predictive performance of an individual model [25]. For example, [22] proposes an ensemble learning model which improves the predicting accuracy for driver lane change intention inference. In [22], a data augmentation scheme is firstly designed to increase the data volume and generate multiple training sets. Then, based on the Bootstrap aggregating method, different RNN models are assembled to inference the lane change intention. Results show that the accuracy and robustness of the RNN models for intention inference are improved by applying the ensemble method. To the best of our knowledge, there are few ensemble learning studies focusing on vehicle trajectory prediction. Therefore, this paper proposes a novel ensemble learning framework for vehicle trajectory prediction in interactive scenarios, and we term it as Interactive Ensemble Trajectory Predictor (IETP). The main contributions of this paper are as follows:

- A novel ensemble learning framework is proposed for vehicle trajectory prediction in interactive scenarios. The framework assembles interaction-aware trajectory predictors as base learners to build an ensemble learner, improving the predicting performance of base learners.
- Two ensemble learning strategies are proposed for trajectory predictors to handle the maneuvers classification and trajectory prediction tasks, respectively.
- Three experiments using a different number of the data are conducted to evaluate the proposed framework. Moreover, a comparative study in the time-cost of the proposed framework is also presented.

The remainder of this paper is organized as follows. Section II introduces the construction of IETP. Section III shows experimental results and analysis. Lastly, the conclusion of this paper is presented in Section IV.

II. INTERACTIVE ENSEMBLE TRAJECTORY PREDICTOR

This section will introduce the proposed IETP, which is displayed in Fig. 1. IETP assembles interaction-aware models which consider interactions between vehicles as base learners to build an ensemble learner, predicting trajectories in interactive scenarios. Firstly, inputs of IETP are fed into

each base learner. Each base learner then predicts trajectories based on handling interactions between vehicles. Finally, IETP obtains outputs by applying ensemble strategies on predictions from all base learners. IETP aims to provide a general ensemble learning approach for trajectory prediction in interactive scenarios. Convolutional Social Pooling models proposed in [10] are adopted as base learners in this study. It can be regarded as an example of IETP instead of a fixed approach.

A. Problem Formulation

According to [10], the trajectory prediction is formulated as estimating the probability distribution of future positions of the predicted vehicle in this study. Inputs to IETP are historical trajectories of the predicted vehicle and surrounding vehicles. The historical trajectories can be described as:

$$\mathbf{X} = [\mathbf{h}^{(t-t_h)}, \dots, \mathbf{h}^{(t-1)}, \mathbf{h}^{(t)}], \quad (1)$$

where $\mathbf{h}^{(t)} = [x_0^{(t)}, y_0^{(t)}, x_1^{(t)}, y_1^{(t)}, \dots, x_i^{(t)}, y_i^{(t)}, \dots, x_n^{(t)}, y_n^{(t)}]$ are x and y coordinates at time t . Here, the y-axis points to the driving direction. The x-axis points to the lateral direction. In vector $\mathbf{h}^{(t)}$, $x_0^{(t)}$ and $y_0^{(t)}$ are coordinates of the vehicle being predicted at time t , while $x_i^{(t)}, y_i^{(t)}$ ($i=1,2,3,\dots,n$) represent the coordinates of different surrounding vehicles at time t . The output of IETP is a probability distribution over future coordinates of the predicted vehicle. If future coordinates are described as $\mathbf{Y}$, the output distribution can be described as:

$$P(\mathbf{Y}|\mathbf{X}) = \sum_{i=1}^6 P_G(\mathbf{Y}|d_i, \mathbf{X})P(d_i|\mathbf{X}), \quad (2)$$

where $\mathbf{G} = [\mathbf{G}^{(t+1)}, \dots, \mathbf{G}^{(t+t_f)}]$ are parameters of a bivariate Gaussian distribution at each time step in the prediction horizon. Detailed parameters of a bivariate can be described as:

$$\mathbf{G}^{(t)} = [m_x^{(t)}, m_y^{(t)}, s_x^{(t)}, s_y^{(t)}, r^{(t)}], \quad (3)$$

where $m_x^{(t)}$ and $m_y^{(t)}$ are mathematical expectations of the predicted future locations in the lateral and longitudinal direction at time t . While $s_x^{(t)}$ and $s_y^{(t)}$ are variances of $\mathbf{X}$ and $\mathbf{Y}$, and $r^{(t)}$ is the coefficient of association. d_i ($i=1, 2, 3, 4, 5, 6$) represents six driving maneuvers defined in [10]. Detailed types of maneuvers are also shown in Fig. 1.

B. Ensemble Learning Approaches for IETP

Since the diversity of base learners is critical to the performance of the ensemble learner [24], this study applies the Bootstrap aggregating (bagging) method to obtain diverse base learners. Firstly, the NGSIM dataset [26] is processed as shown in Fig. 2. The processed dataset consists of trajectories information. Secondly, several sub-training sets are obtained by random sampling with replacement based on the processed dataset. In detail, every sub-training set has the same number of samples as the whole training set. This type of sampling is named Bootstrap sampling [27]. Then, base learners are trained on these sub-training sets, respectively. It should be noted that the trained weights of networks in these

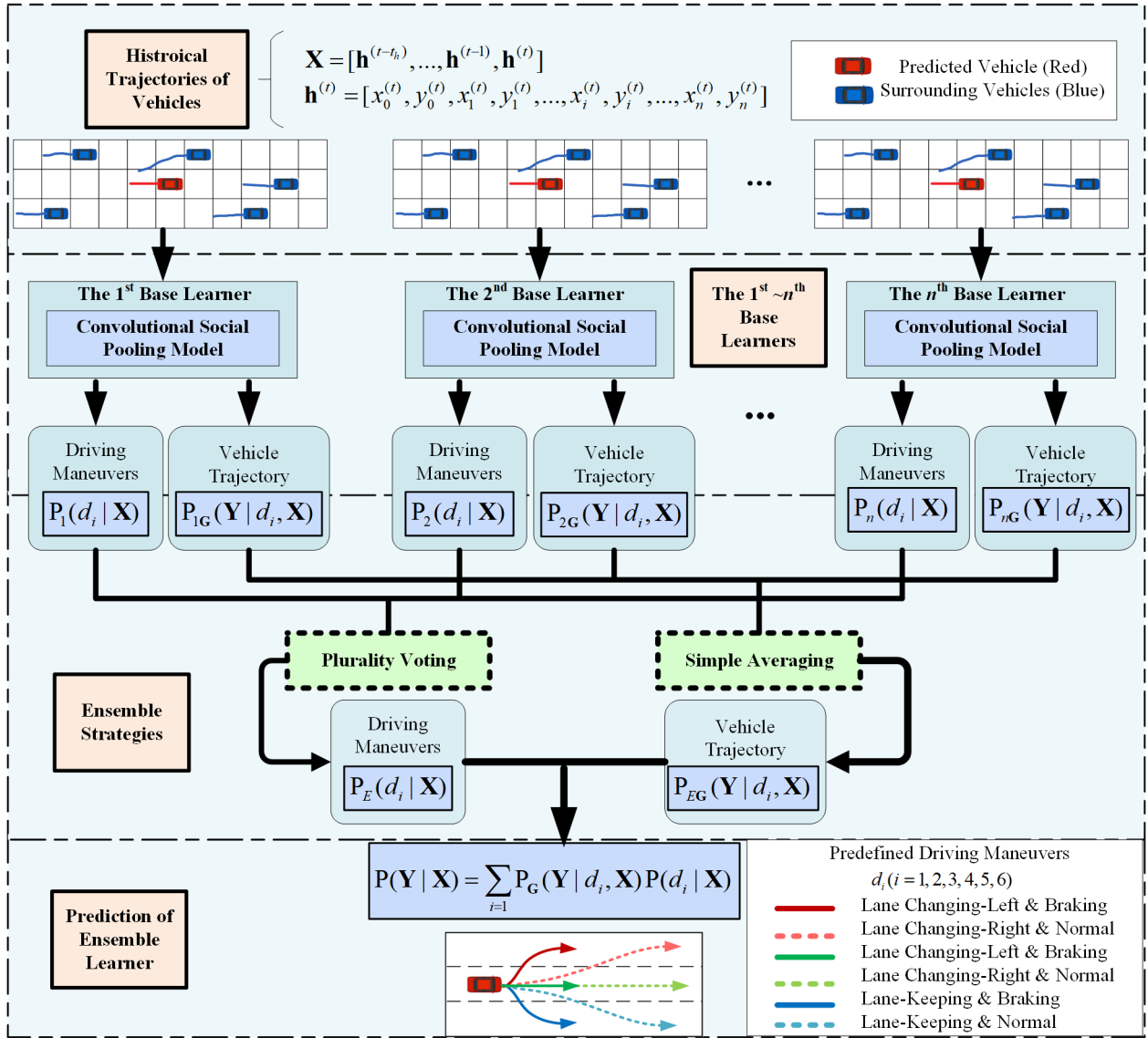


Fig. 1: IETP assembles n interaction-aware models as base learners. To demonstrate the performance of IETP, this study adopts Convolutional Social Pooling models as base learners in experiments. First, IETP assembles n base learners by Bootstrap aggregating. Then, two specific ensemble strategies including plurality voting and simple averaging are used to build the ensemble learner.

base learners are different due to diverse training samples. To distinguish different base learners, all base learners are numbered.

After obtaining diverse base learners, as shown in Fig. 1, inputs described as (1) are fed into each base learner. Then, each base learner outputs prediction as (2). Finally, IETP makes predictions by assembling predictions from base learners through ensemble strategies including plurality voting and simple averaging.

Plurality voting is a commonly used ensemble strategy in classification task [23]. If labels in a classification task are defined as:

$$C = \{c_1, c_2, \dots, c_j, \dots, c_T\}, \quad (4)$$

the subscript j ($j = 1, 2, 3, \dots, T$) represents the class of label, while T is the total number of labels. Each base learner

will predict from set C . Thus, the prediction of each base learner on a sample $\mathbf{x}$, can be expressed as:

$$B_i(\mathbf{x}) = \{b_i^1(\mathbf{x}), b_i^2(\mathbf{x}), \dots, b_i^j(\mathbf{x}), \dots, b_i^T(\mathbf{x})\}. \quad (5)$$

In (5),

$$b_i^j(\mathbf{x}) = \begin{cases} 1, & \text{predicted} \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

is the prediction made by the base learner b_i on label c_j . Moreover, on every sample $\mathbf{x}$, only a label is predicted. Thus, the prediction made by each base learner can be seen as a one-hot vector. If n base learners are used to build an ensemble learner, the predicted result of the ensemble learner can be described as:

$$\begin{cases} c_E(\mathbf{x}) = c_j \\ j = \arg \max_j \sum_{i=1}^n b_i^j(\mathbf{x}) \end{cases}, \quad (7)$$

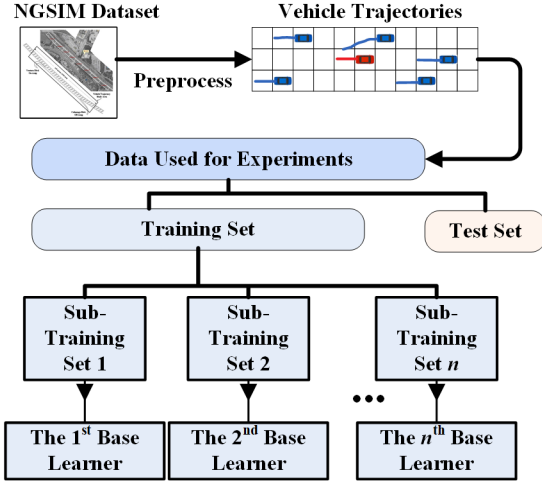


Fig. 2: Data-processing in experiments.

where $c_E(\mathbf{x})$ is the prediction on sample $\mathbf{x}$ made by the ensemble learner. The result is the most voted label predicted by n base learners. If more than one labels get the most votes, the result will be obtained from them randomly.

In IETP, each base learner makes trajectory prediction in terms of six driving maneuvers. Plurality voting method is applied to obtain ensemble driving maneuvers. It should be noted that, original predictions made by base learners are probability format, which can be expressed as:

$$\begin{cases} [p_1, p_2, p_3, p_4, p_5, p_6] \\ \sum_{i=1}^6 p_i = 1 \end{cases}, \quad (8)$$

where p_i ($i = 1, 2, 3, 4, 5, 6$) is the probability value corresponding to a class of driving maneuver. Firstly, each original prediction is encoded into a one-hot vector as described in (5). Then, the plurality voting method is applied to these one-hot vectors to obtain ensemble learning results. Finally, the predicted one-hot vector is decoded back into the probability format as (8).

Simple averaging is a widely used ensemble strategy when dealing with regression tasks [23]. Results from ensemble learners are obtained by averaging predicted values of all base learners. In this study, outputs of our proposed IETP are probability distributions as described in Section II-A. Since the probability is assumed with a bi-variate Gaussian distribution, specific outputs can be described by five parameters of Gaussian distribution in (3). Therefore, detailed outputs of IETP are averaged values of these parameters.

C. Loss Fuction

According to [10], in the training process, the negative log-likelihood can be described as:

$$-\log(P_G(\mathbf{Y}|d_{\text{true}}, \mathbf{X})P(d_{\text{true}}|\mathbf{X})). \quad (9)$$

Equation (9) is minimized overall training samples. In (9), d_{true} provided by each training instance, is the driving maneuver class that is actually performed.

TABLE I: KEY PARAMETERS OF EACH BASE LEARNER IN IETP

Order	Layers	Hidden states	Size	Depth
1	LSTM encoder	64	-	-
2	Social tensor	-	13x3	-
3	Convolutional layer	-	3x3	64
4	Convolutional layer	-	3x1	16
5	Max pooling layer	-	2x1	-
6	LSTM decoder	128	-	-

D. Implementation Details

In this study, models observe historical trajectories for three seconds and predict trajectories for next five seconds. Moreover, each base learner in IETP is trained using the Adam [28] optimizer with a learning rate of 0.001. According to [10], the key parameters of each base learner is shown in Table I. The leaky-ReLU activation with α equals 0.1 is used for all layers. The model is implemented using PyTorch¹.

III. EXPERIMENTS

This section presents results and statistical analysis of experiments based on the NGSIM dataset. Three experiments using a different number of the data are conducted to evaluate IETP. In each experiment, 20 sub-training sets are obtained by the Bootstrap sampling. Then, these sub-training sets are used to train 20 base learners, respectively. Finally, 20 ensemble learners are built by assembling different base learners. Detailed experiments are introduced in the following.

The first experiment is to evaluate the predicting accuracy of IETP. The training and test sets are split from the whole processed NGSIM datasets. The second experiment compares the performance of IETP and the base learners, where training and test sets are split from the “US-101-0750-0805” segment. The third experiment is to evaluate the data efficiency of IETP comparing to several baselines. Each base learner in IETP is trained with 50% of the training data compared to the first experiment. Then, IETP is tested on the same test set in the first experiment.

A. Dataset and Data-processing

The publicly available dataset NGSIM including US-101 and I-80, is used in this study. The dataset consists of real freeway traffic and traffic contexts, which can present inter-active scenarios in freeways. In each experiment, features including time frames, coordinates of vehicles are firstly extracted. Then, the dataset is split into a training set and a test set. The test set has a fourth of samples from the dataset. Finally, as described in Section II, 20 sub-training sets are obtained by the Bootstrap sampling based on the whole training set.

B. Evaluating Metrics and Baseline Models

According to [10], the root of mean squared error (RMSE) and negative log-likelihood (NLL) are used to evaluate

¹<https://pytorch.org>

models in this study. In detail, RMSE is computed by

$$RMSE = \sqrt{\frac{1}{k} \sum_{i=1}^k (\mathbf{Y}_i - \mathbf{Y}_{i,true})^2} \quad (10)$$

and NLL is computed by

$$NLL = -\log \left(\sum_i P_G(\mathbf{Y}|d_i, \mathbf{X}) P(d_i|\mathbf{X}) \right). \quad (11)$$

It should be noted that, in (10), the predicted trajectory $\mathbf{Y}_i$ is the one under the maneuver with the most probability when using multi-modal outputs. And $\mathbf{Y}_{i,true}$ is the ground truth. It should also be noted that k in (10) is the number of data samples, and d_i in (11) represents different driving maneuvers.

IETP is compared with following models, which all consider interactions between surrounding vehicles.

- C-VGMM + VIM: Variational Gaussian mixture models with a Markov random field based on the vehicle interaction module [4]. It is modified to use maneuvers classes to allow a fair comparison as described in [10].
- GAIL-GRU: Generative adversarial imitation learning model described in [29].
- M-LSTM: Maneuvers-LSTM model described in [30].
- S-LSTM: Social Pooling model described in [9].
- NLS-LSTM: Non-local Social Pooling model described in [31].
- CS-LSTM: Convolutional Social Pooling without maneuvers described in [10].
- CS-LSTM-M: Convolutional Social Pooling with maneuvers (CS-LSTM-M) described in [10].

The system settings of IETP include two types: one uses CS-LSTM as base learners, and the other uses CS-LSTM-M as base learners. These two settings are termed IETP-CS and IETP-CS-M, respectively.

C. Experimental Results and Discussion

1) *Experimental Results:* Table II shows the comparisons of RMSE over the prediction horizon in the first experiment. The RMSE values of baselines are results from [10], [31]. Results of IETP-CS-M and IETP-CS are average results of all ensemble learners. Experimental results show that IETP-CS has the lowest RMSE over the prediction horizon compared to all models. Results also show that IETP-CS and IETP-CS-M decrease the RMSE compared to the individual CS-LSTM and CS-LSTM-M. Fig. 3 shows the detailed comparisons of RMSE and NLL in the second experiment. The dot markers in Fig. 3 represent the performance of base learners (CS-LSTM-M), and the triangle markers with lines represent the performance of ensemble learners (IETP-CS-M). It should be explained that the n^{th} base learner is the individual base learner trained on the n^{th} sampling set. And the n^{th} ensemble learner is built by assembling n base learner (from the 1st to the n^{th}). It can be found that ensemble learners have lower RMSE and NLL most of the time. These results show that IETP improves the predicting accuracy of base learners. Fig. 4 shows the variance of the RMSE and NLL from 20 base learners (CS-LSTM-M) and 20 ensemble learners (IETP-CS-M) in the second experiment. As shown in Fig. 4, base learners have a higher variance of the RMSE and NLL than ensemble learners. Moreover, the RMSE and NLL are decreased significantly

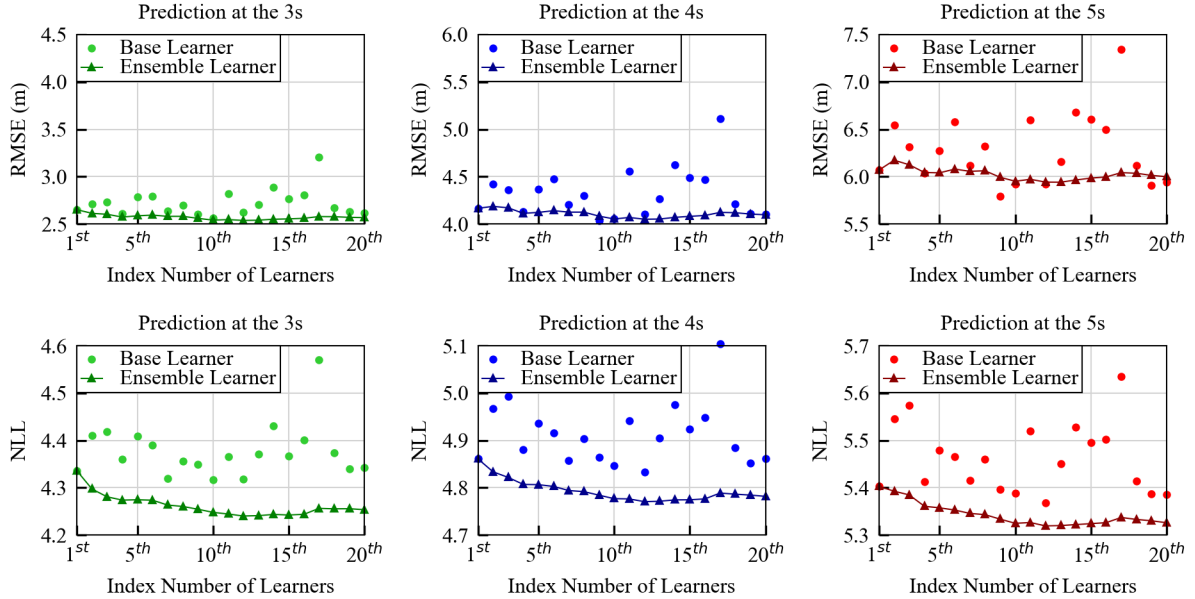


Fig. 3: Comparisons of RMSE and NLL at the 3s, 4s and 5s in the second experiment. The dot markers represent the performance of base learners, and the triangle markers represent the performance of ensemble learners. In this experiment, the base learners are CS-LSTM-M trained with the “US-101-0750-0805” segment of the NGSIM dataset, and the ensemble learners are IETP-CS-M. The test set is also split from the processed “US-101-0750-0805” segment.

TABLE II: COMPARISON OF RMSE OVER THE PREDICTION HORIZON

Prediction horizon (s)	C-VGMM+VIM	GAIL-GRU	M-LSTM	S-LSTM	NLS-LSTM	CS-LSTM-M	CS-LSTM	IETP-CS-M (ours)	IETP-CS (ours)
1	0.66	0.69	0.58	0.65	0.56	0.62	0.61	0.54	0.52
2	1.56	1.51	1.26	1.31	1.22	1.29	1.27	1.22	1.16
3	2.75	2.55	2.12	2.16	2.02	2.13	2.09	2.06	1.94
4	4.24	3.65	3.24	3.25	3.03	3.20	3.10	3.15	2.95
5	5.99	4.71	4.66	4.55	4.30	4.52	4.37	4.52	4.24

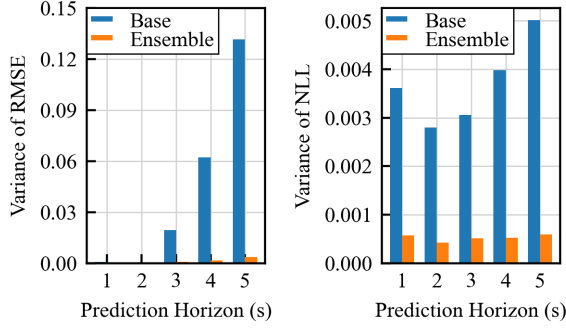


Fig. 4: Comparisons of variances of RMSE (left) and NLL (right) on all base learners and all ensemble learners.

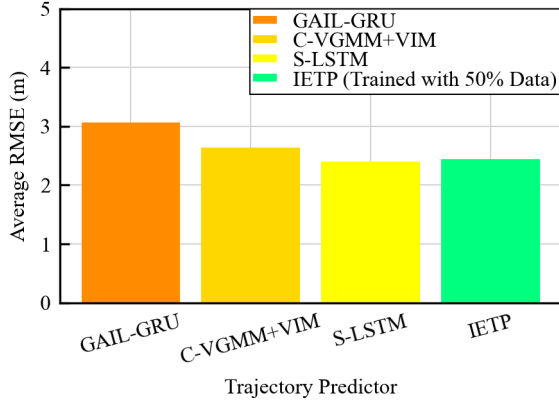
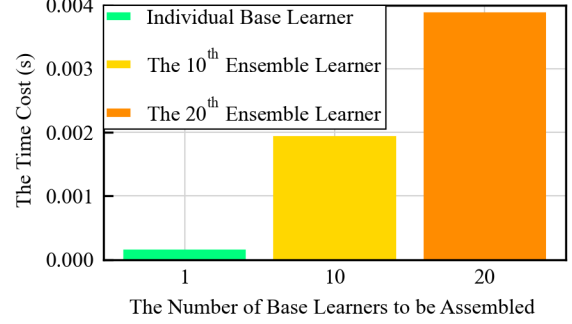


Fig. 5: Average RMSE over prediction horizon. The IETP refers to the IETP-CS trained with 50% data of the whole NGSIM training set.

by applying the ensemble learning approach, especially when the prediction horizon extends further. For example, at the 4s, the variance of RMSE and NLL are decreased by 98% and 87%. And at the 5s, RMSE and NLL are reduced by 97% and 88%, respectively. These results can also be found intuitively in Fig. 3. As shown in Fig. 3, the performance of base learners varies wider than the ensemble learners among 20 groups of testing. Moreover, the gap between the best and the worst performance of base learners can be enormous. For example, at the 5s, the gap between the highest and the lowest RMSE of base learners is 1.56 m. In comparison, the largest gap of ensemble learners is 0.23 m. These results show that IETP has a more stable performance than base learners.

Fig. 5 shows the comparison of average RMSE over


 Fig. 6: The time cost to predict trajectories on a sample: base learner (left), the 10th ensemble learner (middle), and the 20th ensemble learner (right).

the prediction horizon in the third experiment. From Fig. 5, IETP-CS trained with 50% data outperforms C-VGMM+VIM and GAIL-GRU. Besides, the average RMSE values of S-LSTM and IETP-CS are 2.38m and 2.42m, which are very close.

2) *Comparative Study in the Time-cost*: Compared to the individual model, more models need to be loaded in IETP. As a result, it takes more time for ensemble learners to predict trajectories. The time cost of models to predict trajectories on each sample is recorded in this study. Recordings show that the relationship between the time cost and the number of assembled base learners is almost linear. Fig. 6 shows the time cost of models to predict trajectories on each sample. However, we found that using more base learners to build ensemble learners does not significantly improve the performance. For example, from Fig. 3, the performance of the 10th ensemble learner and the 20th ensemble learner are close to each other. In contrast, the 10th ensemble learner has a lower time cost.

The best-performed ensemble learner is different among experiments. This paper may not propose a specific number of base learners to be used. Nevertheless, even the highest time cost, i.e., the time cost of the 20th ensemble learner in experiments, is approximately four milliseconds. Since the timescale for vehicles to change lanes or make a brake action in interactive scenarios is about a few seconds [22], the time cost of IETP is acceptable.

IV. CONCLUSIONS

In this paper, an ensemble learning-based framework is proposed for vehicle trajectory prediction in interactive

scenarios. The proposed framework is built by assembling interaction-aware trajectory predictors through the Bootstrap aggregating. Two ensemble strategies are applied to solve the driving maneuvers classification and trajectory prediction tasks. By adopting Convolutional Social Pooling models as base learners, three experiments are conducted on the NGSIM dataset to evaluate the framework.

Firstly, experimental results show that the proposed framework improves the predicting accuracy compared to base learners. Secondly, experimental results show that the performance variance can be significantly decreased by applying the proposed framework. Since the variance measures the dispersion of the data, it shows that individual models can have a more stable performance by applying the proposed framework. Moreover, the proposed framework with less training data outperforms the GAIL-GRU [29] and C-VGMM+VIM [4] models while reaching a similar performance of the S-LSTM [9]. It indicates that the proposed framework is data-efficient.

In this study, specific base learners in our ensemble learning framework are Convolutional Social Pooling models. However, the proposed ensemble learning framework provides a general ensemble learning-based approach to predict trajectories in interactive scenarios. Since the characteristics and number of base learners are critical to the performance of ensemble learners, selecting different types and numbers of models as base learners will be considered and discussed in future works.

REFERENCES

- [1] Z. Li, J. Gong, C. Lu, and J. Li, "Personalized driver braking behavior modelling in the car-following scenario: An importance weight-based transfer learning approach," *IEEE Transactions on Industrial Electronics*, 2022.
- [2] C. Lu, F. Hu, D. Cao, J. Gong, Y. Xing, and Z. Li, "Transfer learning for driver model adaptation in lane-changing scenarios using manifold alignment," *IEEE transactions on intelligent transportation systems*, vol. 21, no. 8, pp. 3281–3293, 2019.
- [3] C. Lu, F. Hu, D. Cao, J. Gong, Y. Xing, and Z. Li, "Virtual-to-real knowledge transfer for driving behavior recognition: Framework and a case study," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 7, pp. 6391–6402, 2019.
- [4] N. Deo, A. Rangesh, and M. M. Trivedi, "How would surround vehicles move? a unified framework for maneuver classification and motion prediction," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 2, pp. 129–140, 2018.
- [5] Z. Li, J. Gong, C. Lu, and J. Xi, "Importance weighted gaussian process regression for transferable driver behaviour learning in the lane change scenario," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 12497–12509, 2020.
- [6] A. Ess, K. Schindler, B. Leibe, and L. Van Gool, "Object detection and tracking for autonomous navigation in dynamic environments," *The International Journal of Robotics Research*, vol. 29, no. 14, pp. 1707–1725, 2010.
- [7] M. Luber, J. A. Stork, G. D. Tipaldi, and K. O. Arras, "People tracking with human motion predictions from social forces," in *2010 IEEE International Conference on Robotics and Automation*, pp. 464–469, IEEE, 2010.
- [8] J. Liu, X. Mao, Y. Fang, D. Zhu, and M. Q.-H. Meng, "A survey on deep-learning approaches for vehicle trajectory prediction in autonomous driving," *arXiv preprint arXiv:2110.10436*, 2021.
- [9] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 961–971, 2016.
- [10] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1468–1476, 2018.
- [11] K. Xu, Z. Qin, G. Wang, K. Huang, S. Ye, and H. Zhang, "Collision-free lstm for human trajectory prediction," in *International Conference on Multimedia Modeling*, pp. 106–116, Springer, 2018.
- [12] P. Zhang, W. Ouyang, P. Zhang, J. Xue, and N. Zheng, "Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12085–12094, 2019.
- [13] X. Zhao, Y. Chen, J. Guo, and D. Zhao, "A spatial-temporal attention model for human trajectory prediction," *IEEE CAA J. Autom. Sinica*, vol. 7, no. 4, pp. 965–974, 2020.
- [14] Y. Peng, G. Zhang, J. Shi, B. Xu, and L. Zheng, "Sra-lstm: Social relationship attention lstm for human trajectory prediction," *arXiv preprint arXiv:2103.17045*, 2021.
- [15] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2255–2264, 2018.
- [16] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. Reid, S. H. Rezatofighi, and S. Savarese, "Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks," *arXiv preprint arXiv:1907.03395*, 2019.
- [17] J. Sun, Q. Jiang, and C. Lu, "Recursive social behavior graph for trajectory prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 660–669, 2020.
- [18] Z. Zhao, H. Fang, Z. Jin, and Q. Qiu, "Gisnet: Graph-based information sharing network for vehicle trajectory prediction," in *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, IEEE, 2020.
- [19] C. Wang, S. Cai, and G. Tan, "Graphtcn: Spatio-temporal interaction modeling for human trajectory prediction," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3450–3459, 2021.
- [20] Z. Li, C. Lu, Y. Yi, and J. Gong, "A hierarchical framework for interactive behaviour prediction of heterogeneous traffic participants based on graph neural network," *IEEE Transactions on Intelligent Transportation Systems*, 2021.
- [21] Z. Li, J. Gong, C. Lu, and Y. Yi, "Interactive behavior prediction for heterogeneous traffic participants in the urban road: A graph-neural-network-based multitask learning framework," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 3, pp. 1339–1349, 2021.
- [22] Y. Xing, C. Lv, H. Wang, D. Cao, and E. Velenis, "An ensemble deep learning approach for driver lane change intention inference," *Transportation Research Part C: Emerging Technologies*, vol. 115, p. 102615, 2020.
- [23] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, p. e1249, 2018.
- [24] X. Dong, Z. Yu, W. Cao, Y. Shi, and Q. Ma, "A survey on ensemble learning," *Frontiers of Computer Science*, vol. 14, no. 2, pp. 241–258, 2020.
- [25] T. G. Dietterich *et al.*, "Ensemble learning," *The handbook of brain theory and neural networks*, vol. 2, no. 1, pp. 110–125, 2002.
- [26] B. Coifman and L. Li, "A critical evaluation of the next generation simulation (ngsim) vehicle trajectory dataset," *Transportation Research Part B: Methodological*, vol. 105, pp. 362–377, 2017.
- [27] B. Efron and R. J. Tibshirani, *An introduction to the bootstrap*. CRC press, 1994.
- [28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [29] A. Kuefler, J. Morton, T. Wheeler, and M. Kochenderfer, "Imitating driver behavior with generative adversarial networks," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 204–211, IEEE, 2017.
- [30] N. Deo and M. M. Trivedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based lstms," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1179–1184, IEEE, 2018.
- [31] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Non-local social pooling for vehicle trajectory prediction," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, pp. 975–980, IEEE, 2019.