Interactive Behavior Prediction for Heterogeneous Traffic Participants in the Urban Road: A Graph-Neural-Network-Based Multitask Learning Framework

Zirui Li, Jianwei Gong, *Member, IEEE*, Chao Lu, *Member, IEEE*, and Yangtian Yi

***Abstract*—Effectively predicting interactive behaviors of traffic participants in the urban road is the key to successful decision-making and motion planning of intelligent vehicles. In this article, based on the data collected from vehicle on-board sensors, a graph-neural-network-based multitask learning framework (GNN-MTLF) is proposed to accurately predict trajectories of traffic participants with interactive behaviors. The interactive behavior considered in this research includes interactive events and trajectories that are modeled as spatial-temporal graphs using the GNN. Under the GNN-MTLF, the prediction process contains two main parts: recognition of interactive events and prediction of interactive trajectories. An integrated loss function is designed for multitask learning with the purpose of prediction and recognition. The proposed framework is verified using naturalistic driving data in the urban road. Experimental results show a superior performance of the GNN-MTLF compared to baseline methods and the potential for improving the road mobility.**

***Index Terms*—Graph neural network (GNN), interactive behavior modeling, multitask learning.**

I. Introduction

BUILDING and developing intelligent transportation systems (ITSs) is important to improve the road mobility. Considering the safely automated driving in complicated and dynamic scenarios, precisely modeling and predicting behaviors of traffic participants based on vehicle on-board sensors plays an important role in decision making and motion planning of intelligent vehicles. Previous researches on behavior modeling of traffic participants can be categorized into three folds: physical-model-based methods, maneuver-based methods, and interaction-aware methods [1].

Physical-model-based methods are developed based on vehicle kinematic and dynamic models to predict future trajectories. Widely used algorithms applied vehicle properties (e.g., mass and wheel base) environmental conditions (e.g., speed limit and road type) and control inputs (e.g., acceleration, velocity, and steering) into consideration. However, physical-model-based methods have a poor ability in modeling pedestrians, which do not have physical models and constraints. In addition, explicit intentions and maneuvers of different traffic participants are not considered in the physical model.

Due to aforementioned disadvantages, maneuver-based methods are proposed, which model behaviors of traffic participants in two stages: maneuver recognition and behavior prediction. For example, in the lane-changing scenario, predefined maneuvers are left lane changing, right lane changing and lane keeping, based on recognized results in the first step, the step of behavior prediction models and generates planning results. Methods developed in the first step include support vector machine (SVM) [2], [3], hidden Markov model [2], [4], Gaussian mixture model [5], Bayesian network [6], random forest classifiers [7], manifold alignment [8], and transfer learning [9]. And rapidly exploring random tree [10], Gaussian process [11], and cluster-based models [4], [12] are applied in the prediction step. Most of maneuver-based methods model behaviors of traffic participants individually, while the influence and interaction between traffic participants are not considered.

In the complicated and dynamic urban scenario, interaction-aware methods are proposed to provide a better understanding of interactive behaviors. Alahi *et al.* [13] first proposed social-aware long short-term memory (LSTM) (Social LSTM) to model interactions between pedestrians, which was developed by the social pooling mechanism. Based on social LSTM, social GAN was designed by combining with the generative adversarial network (GAN) [14]. To implicitly and quantitatively model interactions, some researches proposed to solve problems by a graph structure with nodes and edges. The graph neural network (GNN) obtains an outstanding performance in trajectory prediction, which can model the interaction of traffic participants based on the combination of nodes and edges in the

Manuscript received September 24, 2020; revised December 13, 2020 and January 28, 2021; accepted March 25, 2021. Date of publication April 16, 2021; date of current version June 15, 2021. Recommended by Technical Editor C. Lv and Senior Editor V. Ivanov. This work was supported by the National Natural Science Foundation of China under Grant 61703041 and Grant U19A2083. *(Corresponding author: Chao Lu.)*

The authors are with the School of Mechanical Engineering, Beijing Institute of Technology, Beijing 100081, China (e-mail: 3120195255@bit.edu.cn; gongjianwei@bit.edu.cn; chaolu@bit.edu.cn; 1120160838@bit.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMECH.2021.3073736.
Digital Object Identifier 10.1109/TMECH.2021.3073736

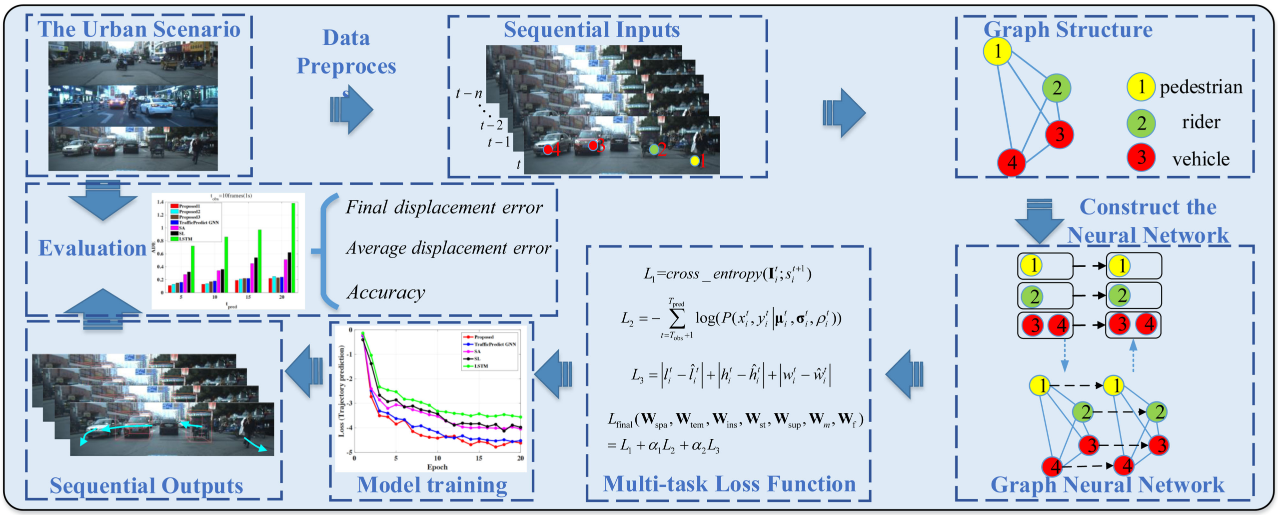

Fig. 1. Overall illustration of the proposed GNN-MTLF.

graph structure [15]. Therefore, the GNN is widely developed in prediction tasks of the ITS, interactive trajectory prediction, and interactive behavior modeling of traffic participants [13], [16]. Jain *et al.* [17] first proposed to apply the LSTM into the graph structure to model dynamic interactions between pedestrians. Based on [17], Ma *et al.* [18] developed a hierarchical GNN to model interactions between heterogeneous traffic participants. The majority of social-aware methods focus on the long-term prediction of single output (trajectory or bounding box). However, in the modeling process of interactive behaviors, limited researches pay attention on the simultaneous generation of multiple-related outputs with common representation and useful information, which can enhance the generalization performance on multiple tasks. The issue can be modeled and leveraged by multitask learning.

The multitask learning, as an important category of transfer learning, is widely developed in robot and autonomous driving [19]. In the research of driver behavior modeling, Xing *et al.* [20] proposed a unified and scalable multitask method for driver behavior reasoning, which combined driver body postures and mental behaviors in the joint feature learning framework. However, the interaction between different traffic participants is not modeled in [20]. Li *et al.* [21] developed a generic multiagent tracking and prediction algorithm to jointly track and predict continuous motions for interactive entities simultaneously. But specific maneuvers are not taken into consideration. In this research, based on the concept of multitask learning, the proposed framework focuses on combining maneuver-based and interaction-aware methods to accurately predict and recognize interactive behaviors of traffic participants.

Main contributions of this article are summarized as follows:

1) This article proposes a GNN-based multitask learning framework (GNN-MTLF) for interactive behaviors modeling, which can recognize and predict interactive behaviors of heterogeneous traffic participants simultaneously. Compared to the TrafficPredict GNN that focuses on accurately predicting the multistep trajectory, the GNN-MTLF simultaneously generates multiple related outputs with common representation and useful information, which extends the diversity of outputs and increases the performance of prediction and recognition.
2) An integrated and unified loss function is designed for the multitask learning. Besides the prediction of the trajectory in the TrafficPredict GNN, three-dimensional (3-D) bounding box and interactive events are also generated by the GNN-MTLF.

The rest of this article is organized as follows. Section II illustrates the problem formulation of interactive behavior modeling, describes the proposed multitask learning framework, and presents details of the graph structure. Experimental results and comparative studies are discussed in Section III. Finally, Section IV concludes this article.

II. GNN-Based Multitask Learning Framework (GNN-MTLF)

The proposed GNN-MTLF is shown in Fig. 1, which consists of three components: the construction of graph structure, the training of the GNN based on the integrated loss function, and the evaluation of multitask outputs. In the first component, nodes and edges of the graph are defined by features and connections of traffic participants, respectively. Then, the graph-based neural network with an integrated multitask loss function is trained. The GNN used in this research is combined by instance layer and category layer, which are designed to model heterogeneous traffic participants in specialty. The integrated loss function is proposed in this research to realize the multitasking learning for the trajectory prediction, 3-D bounding box prediction, and interactive events recognition. Interactive event is the explicit relationship with semantic definition between different traffic participants, which are not considered in Trafficpredict. In this research, the aforementioned definition is named as interactive events. Besides features of interactive trajectories, interactive events between two traffic participants can be applied to model interactive behaviors. Following [22], the interactive event considered in this article refers to the event involving at least two traffic participants when interactions happen between them. Typical interactive events include overtaking from left (right), driving away from left (right), parallel driving in left (right), etc., which can cover most of situations for vehicles, riders, and pedestrians. Detailed description of these events can be found from [22]. Finally, three matrices are selected for the evaluation of the performance in the comparative study.

The modeling process of interaction is decomposed by two steps. First, the graph structure is built to model spatial interactive behaviors, which represents the essential influence for traffic participants at the same time t. The characteristic of the spatial edge is calculated by the spatial distance (Euclidean distance) between two connecting nodes. Second, considering that the motion of traffic participants has the correlation and dependence in adjacent frames, the interaction in time series is modeled by LSTMs, which are applied on nodes and edges in the graph structure.

A. Problem Formulation

The GNN-MTLF is proposed to model interactive behaviors in the urban scenario. As of each traffic participant ith, the feature $\mathbf{f}^t$ is defined as

$$\mathbf{f}_i^t = [x_i^t, y_i^t, c_i^t, \mathbf{bbox}_i^t, a_i^t] \tag{1}$$

where x^t and y^t are locations in x- and y-axes directions, respectively. c_i^t is is the type of the ith traffic participant. a_i^t is the labeled interactive event at time t. The proposed framework is a generic method, which can be applied in the interactive scenario with various numbers of types. In this research, considering the selected dataset contains three kinds of traffic participants, the value of type is 3. $\mathbf{bbox}^t$ is the feature vector of the 3-D bounding box, which is represented by

$$\mathbf{bbox}^t = (l^t, w^t, h^t, \delta^t) \tag{2}$$

where l, w, and h are the length, width, and height, respectively. δ is the heading angle of the 3-D bounding box for ith traffic participants. Then, the framework built in this research can be formulated as

$$\mathbf{O}_{t:t+m} = \phi(x^{t-n:t}, y^{t-n:t}, c^{t-n:t}, \mathbf{bbox}^{t-n:t}, a^{t-n:t}) \tag{3}$$

where $\mathbf{O}_{t:t+m} = [x, y, \mathbf{bbox}, a]_{t:t+m}$ are predicted and recognized results from time t to $t+m$.

The graph structure in the GNN can be defined as follows:

$$G = (A_{\text{instance}}, A_{\text{category}}, E_{\text{Spatial}}, E_{\text{Temporal}}) \tag{4}$$

where the instance node A_{instance} represents traffic participants in the instance layer with feature $\mathbf{f}$. A_{category} is the supernode in the category layer. E_{Spatial} and E_{Temporal} are the spatial edge and temporal edge, respectively. The interaction between two traffic participants i and j at the time t is modeled as spatial edge $E_{\text{Spatial}}^{t,ij} = (A_i^t, A_j^t)$, which is defined by

$$\mathbf{f}_{ij}^t = (x_{ij}^t, y_{ij}^t, c_{ij}^t, \mathbf{bbox}_{ij}^t, a_{ij}^t) \tag{5}$$

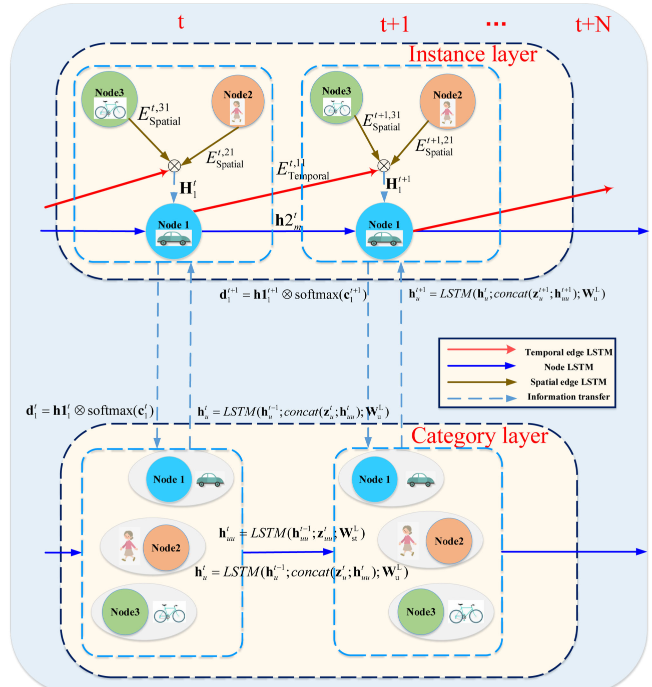

Fig. 2. Relationship and information passing between variables in the GNN.

where $x_{ij}^t = x_i^t - x_j^t$ and $y_{ij}^t = y_i^t - y_j^t$ $\mathbf{bbox}_{ij}^t = \mathbf{bbox}_i^t - \mathbf{bbox}_j^t$ is the relative distance of traffic participants and 3-D bounding boxes from A_j^t to A_i^t, respectively. $\mathbf{bbox}_{ij}^t = \mathbf{bbox}_i^t - \mathbf{bbox}_j^t$ are relative distances of bounding boxes and $\mathbf{bbox}_i^t$ are locations of box corners, which can be calculated by the combination of central position, length, width, and height. The unique encoder is applied to represent c_{ij}^t and a_{ij}^t. And the spatial edge from traffic participants A_j^t to A_i^t is defined as $E_{\text{Spatial}}^{t,ji} = (A_j^t, A_i^t)$. Similarly, the interaction of the same participant in time series is represented by $E_{\text{Temporal}}^{t,ii} = (A_i^t, A_i^{t+1})$ by substituting A_j^t with A_i^{t+1}, which is used to describe the correlation and dependence of same traffic agent in adjacent frames t and $t+1$.

The aforementioned formulation presents the relationship of instance nodes, spatial edges, and temporal edges in the instance layer. And in the category layer, the supernode A_{category} is defined to describe the characteristic of traffic participants with same types. Each instance node in the instance layer has an edge oriented toward to the corresponding supernode, which is applied to transfer the information between two layers. After modeling the similarity of movement patterns, the supernode passes back the information from category to instance layers. Similarly, spatial edges are also applied in the category layer to model spatial interactions of supernodes. The relationship of instance node, supernode, instance layer, and category layer and illustrated in Fig. 2. The temporal edge LSTM, node LSTM, spatial edge LSTM, and information transfer are represented by four types of arrows.

TABLE I
MAIN SYMBOLS IN THE INSTANCE LAYER

Parameters	Annotations
A_{instance}	Instance nodes
E_{Spatial}	Spatial edges
E_{Temporal}	Temporal edges
L_{ij}	The spatial edge LSTM
L_{ii}	The temporal edge LSTM
L_i	The instance LSTM
$\mathbf{h}_{ij}^t$	The hidden state in spatial edge LSTM
$\mathbf{h}_{ii}^t$	The hidden state in temporal edge LSTM
$\mathbf{H}_i^t$	The weighted sum of $\mathbf{h}_{ij}^t$
$\mathbf{h1}_i^t$	The first hidden state of the instance node LSTM
$\mathbf{h2}_i^{t-1}$	The final hidden state of the instance node LSTM

The significance of bold entities in Table I represents the symbol is a vector or matrix.

B. Instance Layer

The instance layer is used to capture individual features of traffic participants. As for each instance node, an LSTM is applied to model instance features and output multistep and multitask predicted results. Considering that different traffic participants have various movement patterns and characteristics, only participants with the same type can share uniform parameters of LSTMs. In this research, the following three types of participants are modeled in the GNN-MTLF: vehicle, pedestrian, and rider. Therefore, three instance LSTMs are trained according to the corresponding node type, simultaneously. Two LSTMs are also applied in the spatial edge and temporal edge, which are designed to model spatial and temporal features. Primary variables in the instance layer are detailed in Table I.

At time t, features $\mathbf{f}_{ij}^t$ of spatial edges $E_{\text{Spatial}}^{t,ij} = (A_i^t, A_j^t)$ are encoded as a vector $\mathbf{z}_{ij}^t$, which are sent into spatial edge LSTMs L_{ij} and generate hidden states $\mathbf{h}_{ij}^t$, which contain the information of spatial interaction.

$$\mathbf{h}_{ij}^t = \text{LSTM}(\mathbf{h}_{ij}^{t-1}, \mathbf{z}_{ij}^t, \mathbf{W}_{\text{spa}}^{\text{L}}) \tag{6}$$

where $\mathbf{W}_{\text{spa}}^{\text{L}}$ are weights of $E_{\text{Spatial}}^{t,ij} = (A_i^t, A_j^t)$. The feature vector $\mathbf{z}_{ij}^t$ can be formulated as

$$\mathbf{z}_{ij}^t = \Omega(\mathbf{f}_{ij}^t, \mathbf{W}_{\text{spa}}^{\text{e}}) \tag{7}$$

where $\Omega(\cdot,\cdot)$ is the embedding function with weight $\mathbf{W}_{\text{spa}}^{\text{e}}$. The detailed description of the LSTM is shown in Fig. 3. According to the definition of the spatial-edge LSTM L_{ij}, the temporal-edge LSTMs $E_{\text{Temporal}}^{t,ij} = (A_i^t, A_i^{t+1})$ can be formulated in the same way. Similarly, hidden states of $\mathbf{h}_{ii}^t$ represent the information of time series.

In the urban scenario, each traffic participant always interacts with several surrounding participants. In the graph structure, the influence of edges between different node pairs is different. Therefore, in the proposed GNN-MTLF, the soft attention mechanism is applied to assign different weight w to different spatial edges of an instance node [23]

$$w(\mathbf{h}_{ij}^t) = \text{softmax}\left(\frac{k}{\sqrt{d_e}}\text{Dot}(\mathbf{W}_{ii}\mathbf{h}_{ii}^t, \mathbf{W}_{ij}\mathbf{h}_{ij}^t)\right) \tag{8}$$

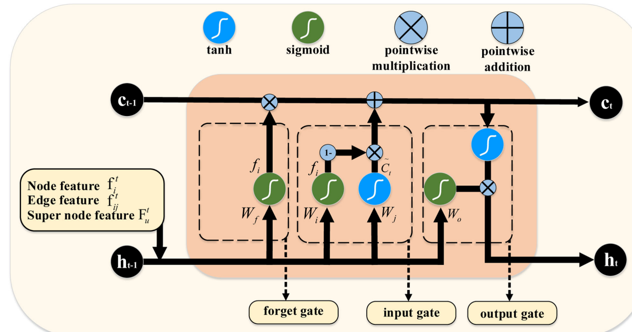

Fig. 3. Algorithmic detail of the LSTM.

where $\mathbf{W}_{ii}$ and $\mathbf{W}_{ij}$ are embedding weights for spatial and temporal edges, respectively. Dot$(\cdot,\cdot)$ is the dot product with the scaling factor $k/\sqrt{d_e}$. From the view of spatial interactions, the influence of different instance nodes can be calculated by the weighted sum of $\mathbf{h}_{ij}^t$

where

$$\mathbf{z}_i^t = \Omega(\mathbf{f}_i^t, \mathbf{W}_{\text{instance}}^{\text{node}}) \tag{9}$$

$$\mathbf{a}_i^t = \Omega(\text{concat}(\mathbf{h}_{ii}^t; \mathbf{H}_i^t); \mathbf{W}_{\text{instance}}^{\text{edge}}) \tag{10}$$

$$\mathbf{h1}_i^t = \text{LSTM}(\mathbf{h2}_i^{t-1}; \text{concat}(\mathbf{z}_i^t; \mathbf{a}_i^t); \mathbf{W}_{\text{instance}}^{\text{L}}). \tag{11}$$

Hidden states of spatial interactions $\mathbf{h}_{ij}^t$ donated from $\mathbf{H}_i^t$ and temporal interactions $\mathbf{h}_{ii}^t$ donated from $\mathbf{H}_i^t$ are concatenated and embedded into the fixed vector $\mathbf{a}_i^t$, which is concatenated with features of instance node as the input of the instance LSTM L_i. In (9)–(11), $\mathbf{W}_{\text{instance}}^{\text{node}}$ and $\mathbf{W}_{\text{instance}}^{\text{edge}}$ are embedding weights, and $\mathbf{W}_{\text{instance}}^{\text{L}}$ is the weight of the ith instance node LSTM cell. First and final states of L_i are represented by $\mathbf{h1}_i^t$ and $\mathbf{h2}_i^{t-1}$, respectively.

C. Category Layer

In the modeling process of interactive driver behaviors, Hou *et al.* [24] consider all agents as a homogeneous group. However, in the urban scenario, interactions between heterogeneous agents can also influence behaviors of on-road traffic participants. In this research, by distinguishing different categories of participants and setting parameters for each type in the category layer, the property of categories can be effectively considered in the proposed framework. As for each type of traffic participants, three supernodes are defined. Similar to LSTMs in the instance layer, three LSTMs are also designed for temporal interactions in the category layer, which consists of four parts: supernodes, temporal edges for supernodes, directed edges from instance nodes in the same group to supernodes, and directed edges from supernodes to instance nodes. Primary variables in the category layer are detailed in Table VII.

In the modeling process of the category layer, at time t, each traffic participant generates the hidden state $\mathbf{h1}_i^t$ and the state of the instance node $\mathbf{c}_i^t$. The aforementioned two features are calculated as the movement feature $\mathbf{d}_m^t$ for the mth instance

TABLE II
MAIN SYMBOLS IN THE CATEGORY LAYER

Parameters	Annotations
A_{Category}	Super nodes in the category layer
$\mathbf{d}_m^t$	The movement feature for m^{th} instance node
$\mathbf{F}_u^t$	The feature of the corresponding super node
$\mathbf{F}_{uu}^t$	The feature of temporal edge in the category layer
$\mathbf{h}_{uu}^t$	The hidden state of temporal edge LSTM
$\mathbf{h}_u^t$	The hidden state of category LSTM (super node)
$\mathbf{h1}_m^t$	The hidden state in spatial edge LSTM
$\mathbf{h2}_m^t$	The final output of instance node

node in the uth category

$$\mathbf{d}_m^t = \mathbf{h1}_i^t \otimes \text{softmax}(\mathbf{c}_i^t). \tag{12}$$

Based in the movement feature calculated previously, the feature of the supernode can be formulated by averaging all instance node $d = \{d_m^t\}_{m=1}^n$ in the uth category.

$$\mathbf{F}_u^t = \frac{1}{n}\sum_{m=1}^{n} d_m^t \tag{13}$$

It indicates that features $\mathbf{F}_u^t$ of supernodes uth consider each instance node in the corresponding category. Based on this mechanism, the information from the instance layer can be effectively transfer to the category layer. The temporal edge $\mathbf{F}_{uu}^t = \mathbf{F}_u^t - \mathbf{F}_u^{t-1}$ between supernodes is combined with the hidden state $\mathbf{h}_{uu}^t$, which is formulated as follows:

$$\mathbf{z}_{uu}^t = \Omega(\mathbf{F}_{uu}^t, \mathbf{W}_{\text{st}}^{\text{e}}) \tag{14}$$

$$\mathbf{h}_{uu}^t = \text{LSTM}(\mathbf{h}_{uu}^{t-1}; \mathbf{z}_{uu}^t; \mathbf{W}_{\text{st}}^{\text{L}}) \tag{15}$$

where $\mathbf{W}_{\text{st}}^{\text{e}}$ and $\mathbf{W}_{\text{st}}^{\text{L}}$ are weights of embedding layer and temporal LSTM cells, respectively. Features of instance group $\mathbf{F}_u^t$ and $\mathbf{h}_{uu}^t$ $\mathbf{h}_u^t$ are combined with features from hidden states $\mathbf{h}_u^t$ in the category layer to model the supernode LSTM as

$$\mathbf{z}_u^t = \Omega(\mathbf{F}_u^t, \mathbf{W}_{\text{u}}^{\text{node}}) \tag{16}$$

$$\mathbf{h}_u^t = \text{LSTM}(\mathbf{h}_u^{t-1}; \text{concat}(\mathbf{z}_u^t; \mathbf{h}_{uu}^t); \mathbf{W}_{\text{u}}^{\text{L}}) \tag{17}$$

where $\mathbf{W}_{\text{u}}^{\text{e}}$ and $\mathbf{W}_{\text{u}}^{\text{L}}$ are embedding weights and supernode LSTM cells, respectively. Finally, the hidden state $\mathbf{h}_u^t$ is concatenated with $\mathbf{h1}_m^t$. The result will be sent back to the instance node to generate the final output $\mathbf{h2}_m^t$

$$\mathbf{h2}_m^t = \Omega(\text{concat}(\mathbf{h1}_m^t; \mathbf{h}_u^t); \mathbf{W}_m^{\text{e}}) \tag{18}$$

where $\mathbf{W}_{\text{i2}}^{\text{e}}$ and $\mathbf{h2}_m^t$ are embedding weights and the final output of the mth instance node.

D. Loss Function for Multitask Learning

In this article, the target of the proposed framework is to recognize interactive events and predict trajectories of surrounding vehicles with bounding box by considering interactions between traffic participants. The predicted module assumes that future positions of traffic participants meet the bivariate Gaussian distribution with mean $\boldsymbol{\mu}_i^t = (\mu_x, \mu_y)_i^t$, standard deviation $\boldsymbol{\sigma}_i^t = (\sigma_x, \sigma_y)_i^t$, and correlation coefficient ρ_i^t. Positions of heterogeneous can be formulated by

$$(x_i^t, y_i^t) \sim [\boldsymbol{\mu}_i^t, \boldsymbol{\sigma}_i^t, \rho_i^t] \tag{19}$$

the hidden state in the instance node is applied to predict aforementioned parameters with the linear function $\Omega(\cdot,\cdot)$ as

$$[\boldsymbol{\mu}_i^t, \boldsymbol{\sigma}_i^t, \rho_i^t] = \Omega(\mathbf{h}_{i2}^{t-1}, \mathbf{W}_f). \tag{20}$$

The loss function of the proposed framework consists of three parts: recognition for interactive events, prediction for trajectories, and prediction for the 3-D boundingbox. The cross-entropy

(a)

(b)

Fig. 4. Two selected interactive scenarios in the BLVD dataset.

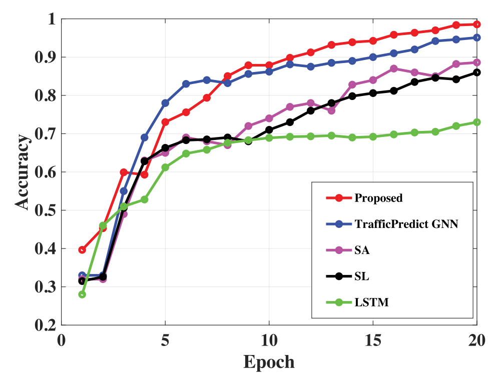

Fig. 5. Training process of recognition.

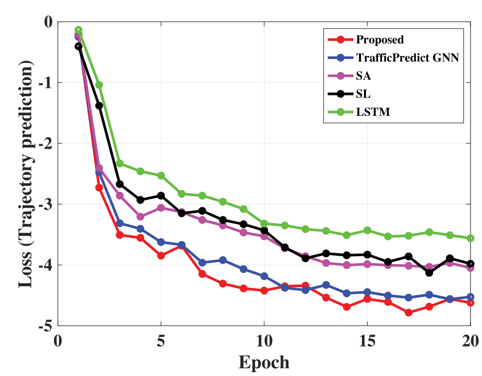

Fig. 6. Training process of the trajectory prediction.

TABLE III
COMPARATIVE RESULTS FOR RECOGNITION OF INTERACTIVE EVENTS

Metric	Participant	Observation	LSTM	SL	SA	Trafficpredict GNN	Proposed#2 (Events+Traj)	Proposed#1 (Events+BBX+Traj)
Accuracy	Pedestrian	10	0.95	0.94	0.95	0.97	0.96	**0.97**
		20	0.66	0.89	0.91	0.96	0.97	**0.98**
		30	0.58	0.85	0.89	0.94	0.98	**0.99**
	Rider	10	0.82	0.93	0.92	0.96	0.96	**0.98**
		20	0.64	0.87	0.89	0.95	**0.98**	**0.98**
		30	0.53	0.81	0.83	0.95	0.95	**0.97**
	Vehicle	10	0.92	0.93	0.92	0.95	0.94	**0.96**
		20	0.62	0.84	0.86	0.93	0.95	**0.98**
		30	0.59	0.8	0.82	0.93	0.97	**0.97**
	Average		0.701	0.873	0.887	0.948	0.962	**0.976**

The significance of bold entities in Table III represent that the number obtains the best performance in the comparative study.

TABLE IV
COMPARATIVE RESULTS FOR MULTISTEP PREDICTION OF INTERACTIVE TRAJECTORIES (ADE/FDE)

Metrics	Agents	LSTM	SL	SA	*Trafficpredict GNN*	Proposed#3 (BBX+Traj)	Proposed#2 (Events+Traj)	Proposed#1 (BBX+Events+traj)
ADE[m]	Pedestrian	0.27	0.14	0.13	0.12	0.11	**0.09**	**0.09**
	Rider	0.38	0.14	0.13	0.14	0.12	0.12	**0.10**
	Vehicle	0.38	0.14	0.12	0.09	0.1	0.08	**0.07**
	Average	0.347	0.14	0.127	0.117	0.11	0.097	**0.087**
FDE[m]	Pedestrian	0.39	0.25	0.24	0.21	0.18	0.17	**0.12**
	Rider	0.57	0.22	0.21	0.2	0.16	0.15	**0.13**
	Vehicle	0.56	0.18	0.17	0.18	0.18	0.14	**0.11**
	Average	0.507	0.217	0.207	0.197	0.174	0.153	**0.12**

The significance of bold entities in Table IV represent that the number obtains the best performance in the comparative study.

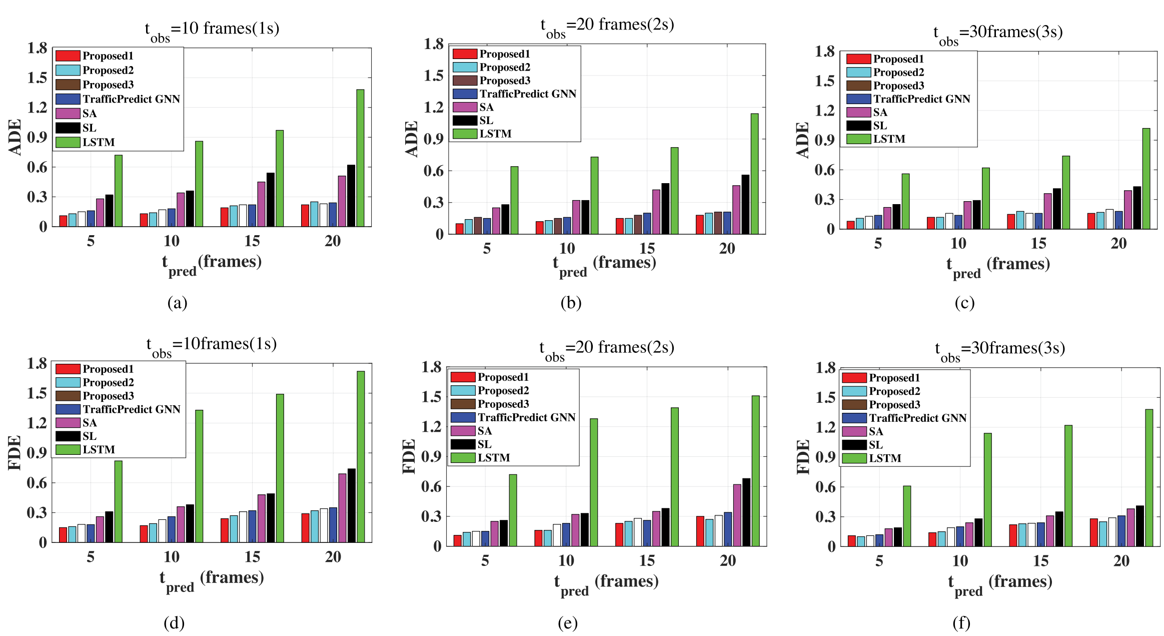

Fig. 7. Comparative results of the trajectory prediction with different input/output length. (a) ADE, $t_{\text{obs}} = 1$ s. (b) ADE, $t_{\text{obs}} = 2$ s. (c) ADE, $t_{\text{obs}} = 3$ s. (d) FDE, $t_{\text{obs}} = 1$ s. (e) FDE, $t_{\text{obs}} = 2$ s. (f) FDE, $t_{\text{obs}} = 3$ s.

function is selected to measure the loss for recognition due to its superiority in the classification problem [25].

$$L_1 = \text{cross\_entropy}(\mathbf{I}_i^t; s_i^{t+1}) \tag{21}$$

where s_i^{t+1} is the ground-truth label for the ith traffic participant. $\mathbf{I}_i^t = \Omega(\mathbf{h}2_i^t, \mathbf{W}_{\mathrm{i}})$ is the embedded vector, which is combined with the second hidden state and embedding weights $\mathbf{W}_{\mathrm{i}}$.

According to [18], the loss function for trajectories prediction is designed by negative log Likelihood as

$$L_2 = -\sum_{t=T_{\text{obs}}+1}^{T_{\text{pred}}} \log(P(x_i^t, y_i^t \,|\, \boldsymbol{\mu}_i^t, \boldsymbol{\sigma}_i^t, \rho_i^t)). \tag{22}$$

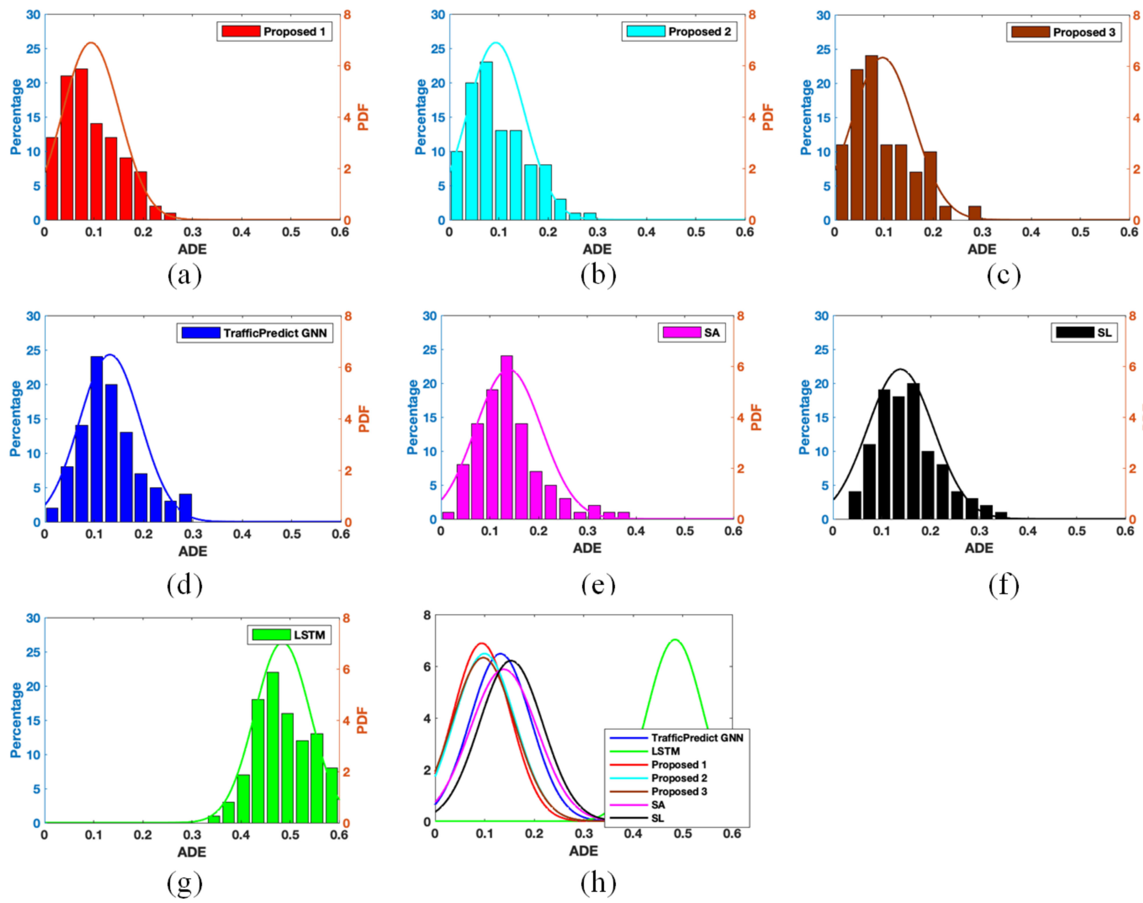

Fig. 8. (a)–(g) Comparative results of error distributions and probability density function (PDF) for ADE. The *X*-axis is the range of the predicted error for each trajectory samples, and the left *Y*-axis is the percentage of samples in different range of errors and the right *Y*-axis is the value of PDF. Error distributions for Proposed#1, Proposed#2, Proposed#3, TrafficPredict GNN, SA, SL, and LSTM are shown in red, cyan, brown, blue, pink, black, and green, respectively. The more the histogram approaches the origin of the *X*-axis, the better the trajectory prediction. (h) Combination of all PDFs in one figure for comparative study.

TABLE V
COMPARATIVE RESULTS FOR MULTISTEP PREDICTION OF THE 3-D BOUNDING BOX (ADE/FDE)

Metrics	Agents	Proposed#3 (BBX+Traj)	Proposed#1 (BBX+Events+Traj)
ADE[m]	Pedestrian	0.85	**0.78**
	Rider	1.02	**0.97**
	Vehicle	0.76	**0.72**
	Average	0.877	**0.797**
FDE[m]	Pedestrian	1.54	**1.39**
	Rider	1.25	**1.18**
	Vehicle	0.90	**0.83**
	Average	1.23	**1.13**

The significance of bold entities in Table V represent that the number obtains the best performance in the comparative study.

As for the prediction of the 3-D bounding box, the loss function is defined as

$$L_3 = \left|l_i^t - \hat{l}_i^t\right| + \left|h_i^t - \hat{h}_i^t\right| + \left|w_i^t - \hat{w}_i^t\right| + \left|\delta_i^t - \hat{\delta}_i^t\right| \quad (23)$$

where l, h, and w are the length, height, and width for the 3-D bounding box, respectively. The total loss for the proposed framework L_{final} is the sum of L_1, L_2, and L_3 with factors α_1 and α_2 as

TABLE VI
MAIN PARAMETERS IN EXPERIMENTS

Parameters	Values
Temporal edge cell	128
Spatial edge cell	128
Node cell	64
Embedding layer	64
Learning rate	0.001
Epoch	20

$$\begin{aligned} &L_{\text{Final}}(\mathbf{W}_{\text{spa}}, \mathbf{W}_{\text{tem}}, \mathbf{W}_{\text{ins}}, \mathbf{W}_{\text{st}}, \mathbf{W}_{\text{sup}}, \mathbf{W}_m, \mathbf{W}_{\text{f}}) \\ &= L_1 + \alpha_1 L_2 + \alpha_2 L_3 \end{aligned} \quad (24)$$

where $\mathbf{W} = (\mathbf{W}_{\text{spa}}, \mathbf{W}_{\text{tem}}, \mathbf{W}_{\text{ins}}, \mathbf{W}_{\text{st}}, \mathbf{W}_{\text{sup}}, \mathbf{W}_m, \mathbf{W}_{\text{f}})$ are weights in the GNN, which can be optimized in the process of back propagation.

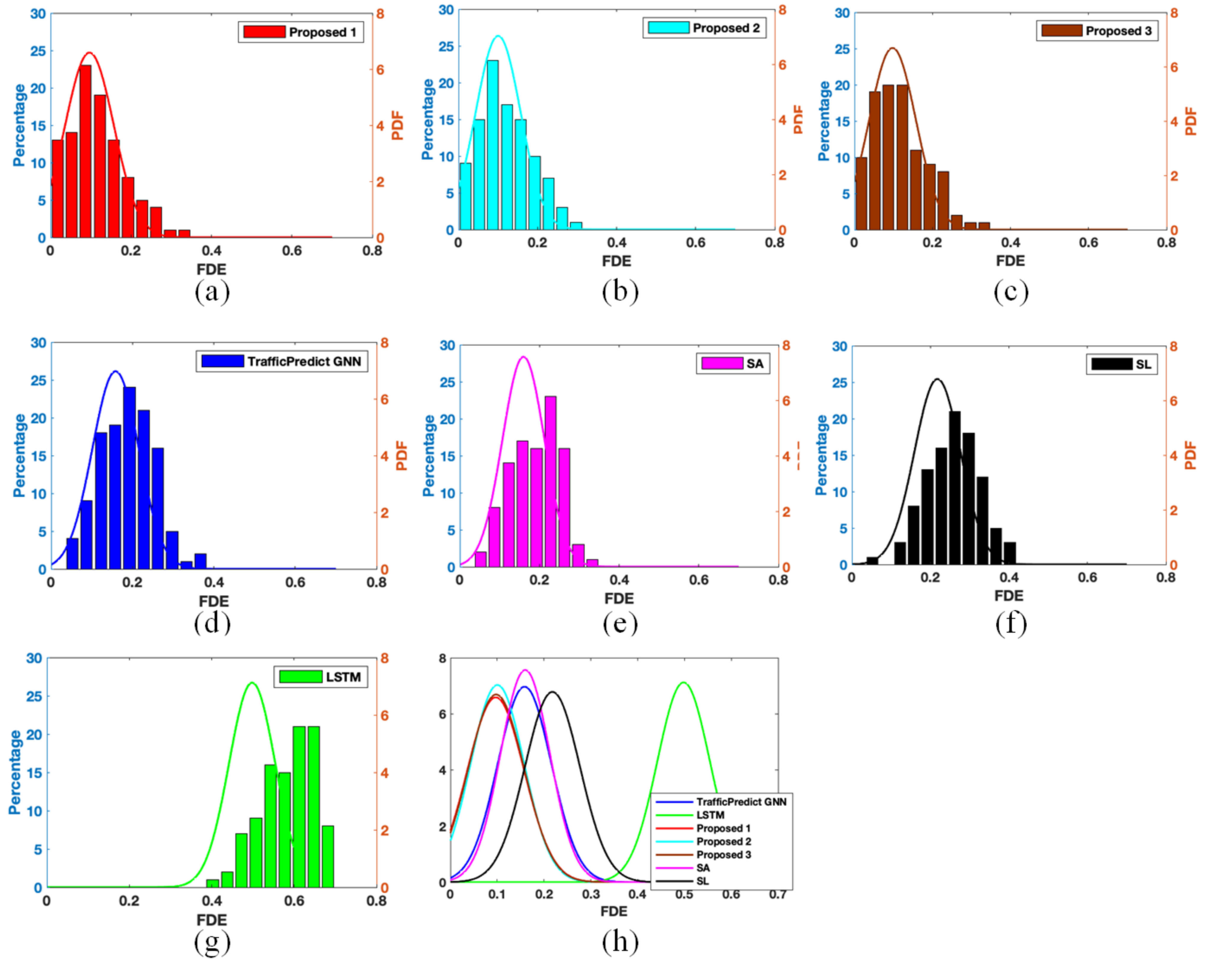

Fig. 9. (a)–(g) Comparative results of error distributions and PDF for FDE. The *X*-axis is the range of the predicted error for each trajectory samples, and the left *Y*-axis is the percentage of samples in different range of errors and the right *Y*-axis is the value of PDF. Error distributions for Proposed#1, Proposed#2, Proposed#3, TrafficPredict GNN, SA, SL, and LSTM are shown in red, cyan, brown, blue, pink, black, and green, respectively. The more the histogram approaches the origin of the *X*-axis, the better the trajectory prediction. (h) Combination of all PDFs in one figure for comparative study.

III. Experiments

In order to verify the proposed GNN-MTLF, the urban scenario with heterogeneous traffic participants and multiple interactive behaviors is selected. Xue *et al.* [22] built a large-scale 5-D semantics benchmark named as built large-scale 5D (BLVD), which contains various labels for interactive events and trajectories. The framework is implemented in Pytorch by a PC with an Intel Core i9 at 2.3-GHz, 16-GB RAM and Intel UHD Graphics 630 (1536 MB). In this section, the introduction of BLVD dataset, baseline methods, experimental results, and comparative analyses (qualitative and quantitative analyses) are detailed in following parts.

A. Dataset

Several researches verify their algorithms by using Next Generation Simulation (NGSIM) dataset, which has following limitations:

1) NGSIM dataset only contains trajectories of vehicles, which is relatively simple compared to urban scenarios;
2) NGSIM only collects one type of traffic participants (vehicles), the influence of heterogeneous traffic participants is not considered in this dataset;
3) NGSIM dataset do not provides specific labels of interactive events.

In order to verify and test the proposed GNN-MTLF, BLVD dataset is selected in this research [22]. BLVD adds temporal and interactive information to the traditional 3-D benchmark, and provides a more comprehensive dynamic 5-D semantic benchmark. BLVD collected information in four scenarios: daytime and low densities, night time and low densities, daytime and high densities, and night time and high densities, and obtained 654 calibrated video clips, which included vehicles, pedestrians, and riders (cyclists and motorcycles). In addition, in BLVD, 13, 8, and 7 interactive events between ego vehicles and traffic participants are marked for vehicles, pedestrians, and riders.

Two selected interactive scenarios with heterogeneous traffic participants are shown in Fig. 4 .

B. Details of Training Process

In order to fully test the performance of the proposed framework, 187, 1039, and 2222 trajectories of pedestrians, riders, and vehicles are selected from BLVD, respectively. 3103 trajectories (90%) are randomly chosen as training data, while the rest (10%) is divided as testing data for cross validation (CV). Different lengths of input and output sequences are selected to evaluate the performance in various conditions. The length of input observation is set as $10\text{frames}(1.0\,\text{s})$, $20\text{frames}(2.0\,\text{s})$, and $30\text{frames}(3.0\,\text{s})$. And the length of the output prediction is set as $5\text{frames}(0.5\,\text{s})$, $10\text{frames}(1.0\,\text{s})$, $15\text{frames}(1.5\,\text{s})$, and $20\text{frames}(2.0\,\text{s})$. Pairs of observation-prediction are selected from alternative length of inputs and outputs, respectively. Detailed settings of parameters are listed in Table VI.

C. Baseline Methods and Evaluated Metrics

The output of the proposed GNN-MTLF includes multistep recognition and prediction. According to [21], three following metrics are selected for the evaluation of performance.

1) Classification Accuracy: The accuracy of recognized interactive events is calculated by

$$\text{Accuracy} = \frac{\Sigma_{i=1}^{N} a_i}{N} \tag{25}$$

where a_t^i represents whether the recognition of interactive events for the ith instance is accurate at time t. a_t^i=0 stands for a correct recognition and a_t^i= 1 stands for a wrong one. N is the total number of samples.

2) Average Displacement Errors (ADE): ADE represents the average Euclidean distance between all predicted and ground-truth positions, which can be formulated as

$$\text{ADE} = \frac{\Sigma_{i=1}^{M} \Sigma_{t=t_{\text{obs}}+1}^{t_{\text{obs}}+t_{\text{pred}}} \sqrt{\left(x_t^i - \hat{x}_t^i\right)^2 + \left(y_t^i - \hat{y}_t^i\right)^2}}{M * t_{\text{pred}}}. \tag{26}$$

3) final Displacement Errors (FDE): FDE represents the Euclidean distance between the predicted and ground truth of final positions, which can be formulated as

$$\text{FDE} = \frac{\Sigma_{i=1}^{M} \sqrt{\left(x_{t_{\text{pred}}}^i - \hat{x}_{t_{\text{pred}}}^i\right)^2 + \left(y_{t_{\text{pred}}}^i - \hat{y}_{t_{\text{pred}}}^i\right)^2}}{M} \tag{27}$$

where (x_t^i, y_t^i) and $(\hat{x}_t^i, \hat{y}_t^i)$ are ground truth and predicted locations of the ith observed instance at time t. M is the total number of traffic participants.

Several algorithms are chosen as baseline methods, which are listed as follows.

1) *Social LSTM (SL):* SL proposes to model interactive behaviors by social pooling module, which considers the influence of neighboring crowds [13].
2) *Social Attention (SA):* SA proposes to apply the attention mechanism in the modeling of spatial relations [17].
3) *LSTM:* LSTM is a general and basic method to model time-series problems in the temporal space [26].
4) *Trafficpredict GNN:* The proposed GNN-MTLF is developed based on the trafficPredict GNN, which is the main baseline method in the comparative study [18]. The TrafficPredict GNN is a hierarchical GNN, which is developed based on the Structure RNN. The TrafficPredict GNN proposes a novel layer to describe the heterogeneous traffic participants. Different kinds of participants (vehicles, pedestrians, and riders) are modeled, respectively, in the category layer. The output of the TrafficPredict GNN is the multistep probabilistic trajectory. The recognition of specific interactive events and the prediction of the 3-D bounding box are not considered in the TrafficPredict GNN.
5) *Proposed#1 (Events+BBX+Traj):* The proposed framework with the integrated loss function, which combines L_1 (recognition), L_2(trajectory prediction), and $L3$ (3-D bounding box prediction).
6) *Proposed#2 (Events+Traj):* The proposed framework with the loss function that contains L_1 (recognition) and L_2 (trajectory prediction).
7) *Proposed#3 (BBX+Traj):* The proposed framework with the loss function that contains L_2 (trajectory prediction) and L_3 (3-D bounding box prediction).

D. Experimental Results

Several experiments are conducted to evaluate the performance of the proposed framework. The variation of loss in the training process is presented in Figs. 5 and 6. Recognized results of interactive events is detailed in Table III, while Tables IV and V show prediction results of the multistep trajectory and 3-D bounding box. To further discuss and fully present the comparative result, the distribution of error and the influence of the trajectory length are shown in Figs. 7–9. Two analyses (qualitative analyses and quantitative analyses) are presented in the following part.

E. Analysis

1) Qualitative Analyses: In order to qualitatively study comparative results, besides four baseline methods (LSTM, SA, SL, and TrafficPredict GNN), results of three proposed methods (Proposed#1, Proposed#2, and Proposed#3) are also detailed in experiments. In Table III, with different length of observations, the proposed#1 (Events+BBX+Traj) obtains the best results in the recognition of interactive events for pedestrians, riders, and vehicles. It indicates that the proposed integrated loss function for multitask learning can improve the performance in the multistep recognition. Comparing Proposed#2(Events+Traj) with Proposed#1(Events+BBX+Traj), the module of the 3-D bounding box prediction in the loss function has positive affect in the recognition (Average accuracy: Proposed#2, 0.962; Proposed#1, 0.976). Meanwhile, in Fig. 5, the variation of the accuracy in the training process indicates that the proposed framework obtains the best performance when the value of epoch reaches 8.

As for results of the trajectory prediction shown in Table IV, social-aware methods (SL, SA, TrafficPredict GNN, and Proposed) present a better performance compared to the general LSTM, which reflects the importance of interactive behavior modeling in trajectory prediction. Comparing Proposed#1 with the TrafficPredict GNN, the GNN-MTLF can reduce ADE and FDE with the integrated loss function (ADE average: 0117m→ 9.987 m; FDE average: 0.197 m→ 0.12 m). The loss of the trajectory prediction in the training process is detailed in Fig. 5, which also presents the similar trend.

The prediction of the 3-D bounding box in this research is the combination of the trajectory prediction and the shape of the bounding box. Therefore, the performance of the prediction is closely related to the trajectory prediction. Comparative results of ADE and FDE in Table V is calculated by the sum of key points in the 3-D bounding box, which is similar to the evaluation of the trajectory prediction. By comparing Proposed#1 and Proposed#3 in Table V, Experimental results show that the information from interactive events can improve the performance of prediction.

2) Quantitative Analyses: The aforementioned qualitative analysis illustrates the overall conclusion based on comparative results. Then, the quantitative analysis will present a further study for the proposed GNN-MTLF. In the recognition of interactive events, as for average results, the best result in baseline methods is 0.948 obtained from the Trafficpredict GNN. It indicates that the modeling of heterogeneous traffic participants in the category layer can promote the performance compared to other baseline methods. The proposed framework improves the recognized accuracy from 0.948 to 0.976 by 2.8%, which demonstrates the contribution of the multitask loss function. Meanwhile, the accuracy of Proposed#1 concentrates from 0.96 to 0.99, while the variation of Trafficpredict GNN is from 0.93 to 0.97. In the training process, social-aware methods obtain further improvements after the 8 epoch (SL: from 0.71 to 0.86; SA: from 0.74 to 0.886; Trafficpredict GNN: from 0.862 to 0.951; and Proposed#1: from 0.878 to 0.985), while the accuracy of the general LSTM only increase from 0.689 to 0.73. In the prediction of trajectory, the average ADE and FDE of the Proposed#1 are 0.087 and 0.12 m, respectively. Compared to Trafficpredict (ADE: 0.11 m and 0.174 m), predicted errors decrease by 20.9% and 31%, respectively. Comparative results between Proposed#1, #2, and #3 verify the primary conclusion in the qualitative analysis, which is also demonstrated in results of the 3-D bounding box prediction.

To further present the performance of proposed and baseline methods, comparative results of the trajectory prediction with different input/output length and comparative results of error distributions are shown in Figs. 7–9. In Fig. 7, the length of sequential input changes from 10 frames to 30 frames (1 s to 3 s), while the length of the sequential output changes from 5 to 20. In general, the proposed GNN-MTLF obtains best performance of ADE and FDE in the most of conditions. And social-aware methods make obvious improvements compared to the general LSTM. The conclusion from Fig. 7 can also be obtained from comparative results of error distributions, which is detailed in Figs. 8 and 9.

TABLE VII
DETAILED DESCRIPTION OF INTERACTIVE EVENTS

Type	Index	Interactive events
Vehicle	1	Overtaking from left
	2	Overtaking from right
	3	Driving away to left
	4	Driving away to right
	5	Driving in from left
	6	Driving in from right
	7	Straight accelerating
	8	Straight decelerating
	9	Parallel driving in left
	10	Parallel driving in right
	11	Uniformly straight driving
	12	Stopping
	13	Others
Pedestrian	1	Walking away from ego-vehicle and getting closer
	2	Walking up to ego-vehicle and getting closer
	3	Crossing quickly from right
	4	Crossing slowly from right
	5	Crossing quickly from left
	6	Crossing slowly from left
	7	Stopping
	8	Others
Rider	1	Riding away and getting closer
	2	Riding away and getting farther
	3	Riding up and getting closer
	4	Crossing from right
	5	Crossing from left
	6	Stopping
	7	Others

IV. CONCLUSION

In this article, a GNN-based multitask learning framework is proposed for interactive behavior modeling based on the GNN, in which an integrated loss function is designed for trajectory prediction, interactive events recognition, and 3-D bounding box prediction. The proposed framework is verified using BLVD dataset, which contains labeled trajectories, events, and bounding boxes obtained from the vehicle onboard sensor. Experimental results indicate that the framework can improve the performance of both recognition and prediction by multitask learning compared to baseline methods. It can benefit intelligent mechatronic systems toward more accurate prediction and recognition of traffic participants. The proposed framework is a generic multitask learning method for interactive behavior modeling and can be applied in various scenarios.

The proposed framework in this article mainly constructs an end-to-end GNN model to recognize and predict interactive behaviors. The constraint of vehicle kinematics and dynamics are not included in this framework, which will be considered and discussed in the future work.

APPENDIX

The following table presents the detailed description of interactive events for vehicles, pedestrians, and riders, respectively.

REFERENCES

[1] S. Lefévre, D. Vasquez, and C. Laugier, “A survey on motion prediction and risk assessment for intelligent vehicles,” *Robomech J.*, vol. 1, no. 1, pp. 1–14, 2014.

[2] G. S. Aoude, V. R. Desaraju, L. H. Stephens, and J. P. How, "Driver behavior classification algorithms at intersections and validation using naturalistic data," in *Proc. Intell. Veh. Symp.*, 2011, pp. 601–606.
[3] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, "Learning-based approach for online lane change intention prediction," in *Proc. IEEE Intell. Veh. Symp.*, 2013, pp. 797–802.
[4] B. T. Morris and M. M. Trivedi, "Trajectory learning for activity understanding: Unsupervised, multilevel, and long-term adaptive approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2287–301, Nov. 2011.
[5] Z. Li, B. Wang, J. Gong, T. Gao, C. Lu, and G. Wang, "Development and evaluation of two learning-based personalized driver models for pure pursuit path-tracking behaviors," in *Proc. IEEE Intell. Veh. Symp.*, 2018, pp. 79–84.
[6] M. Schreier, V. Willert, and J. Adamy, "Bayesian, maneuver-based, long-term trajectory prediction and criticality assessment for driver assistance systems," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst.*, 2014, pp. 334–341.
[7] J. Schlechtriemen, F. Wirthmueller, A. Wedel, G. Breuel, and K.-D. Kuhnert, "When will it change the lane? A probabilistic regression approach for rarely occurring events," in *Proc. IEEE Intell. Veh. Symp.*, 2015, pp. 1373–1379.
[8] 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 Trans. Veh. Technol.*, vol. 68, no. 7, pp. 6391–6402, Jul. 2019.
[9] Z. Li *et al.*, "Transferable driver behavior learning via distribution adaption in the lane change scenario," in *Proc. IEEE Intell. Veh. Symp.*, 2019, pp. 193–200.
[10] G. Aoude, J. Joseph, N. Roy, and J. How, "Mobile agent trajectory prediction using Bayesian nonparametric reachability trees," in *Proc. AIAA Infotech Aerosp. Conf.*, 2011, p. 1512.
[11] C. Laugier *et al.*, "Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety," *IEEE Intell. Transp. Syst. Mag.*, vol. 3, no. 4, pp. 4–19, Dec.–Mar. 2011.
[12] J. Wiest, M. Höffken, U. Kreßel, and K. Dietmayer, "Probabilistic trajectory prediction with Gaussian mixture models," in *Proc. IEEE Intell. Veh. Symp.*, 2012, pp. 141–146.
[13] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 961–971.
[14] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social GaN: Socially acceptable trajectories with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2255–2264.
[15] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, Jan. 2021.
[16] L. Zhao *et al.*, "T-GCN: A temporal graph convolutional network for traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3848–3858, Sep. 2020.
[17] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5308–5317.
[18] Y. Ma, X. Zhu, S. Zhang, R. Yang, and D. Manocha, "Trafficpredict: Trajectory prediction for heterogeneous traffic-agents," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, pp. 6120–6127.
[19] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[20] Y. Xing, C. Lv, W. Wang, D. Cao, and E. Velenis, "A unified multi-scale and multi-task learning framework for driver behaviors reasoning," 2020, *arXiv:2003.08026*.
[21] J. Li, W. Zhan, Y. Hu, and M. Tomizuka, "Generic tracking and probabilistic prediction framework and its application in autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3634–3649, Sep. 2020.
[22] J. Xue *et al.*, "Blvd: Building a large-scale 5D semantics benchmark for autonomous driving," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 6685–6691.
[23] A. Vemula, K. Muelling, and J. Oh, "Social attention: Modeling attention in human crowds," in *Proc. IEEE Int.Conf. Robot. Automat.*, 2018, pp. 1–7.
[24] L. Hou, L. Xin, S. E. Li, B. Cheng, and W. Wang, "Interactive trajectory prediction of surrounding road users for autonomous driving using structural-LSTM network," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4615–4625, Nov. 2020.
[25] J. Shore and R. Johnson, "Properties of cross-entropy minimization," *IEEE Trans. Inf. Theory*, vol. 27, no. 4, pp. 472–482, Jul. 1981.
[26] F. Altche and D. L. F. Arnaud, "An LSTM network for highway trajectory prediction," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst.*, 2017, pp. 353–359.

Zirui Li received the B.S. degree, in 2019, from the Beijing Institute of Technology, Beijing, China, where he is currently working toward the Ph.D. degree, both in mechanical engineering.

His research interests include intelligent vehicles, driver behavior modeling, and transfer learning.

Jianwei Gong (Member, IEEE) received the B.S. degree in mechanical engineering from the National University of Défense Technology, Changsha, China, in 1992, and the Ph.D. degree from the Beijing Institute of Technology, Beijing, China, in 2002.

Between 2011 and 2012, he was a Visiting Scientist with the Robotic Mobility Group, Massachusetts Institute of Technology, Cambridge, MA, USA. He is currently a Professor and the Director with the Intelligent Vehicle Research Centre, School of Mechanical Engineering, Beijing Institute of Technology. His research interests include intelligent vehicle environment perception and understanding, decision making, path/motion planning, and control.

Chao Lu (Member, IEEE) received the B.S. degree in transport engineering from the Beijing Institute of Technology (BIT), Beijing, China, in 2009, and the Ph.D. degree in transport studies from the University of Leeds, Leeds, U.K., in 2015.

In 2017, he was a Visiting Researcher with the Advanced Vehicle Engineering Centre, Cranfield University, Cranfield, U.K. He is currently a Lecturer with the School of Mechanical Engineering, BIT. His research interests include intelligent transportation and vehicular systems, driver behavior modeling, reinforcement learning, and transfer learning and its applications.

Yangtian Yi received the B.E. degree, in 2020, from the Beijing Institute of Technology, Beijing, China, where he is currently working toward the M.A.Sc. degree, both in mechanical engineering.

His research interests include graph neural network and intelligent vehicles.