VIF-GNN: A Novel Agent Trajectory Prediction Model based on Virtual Interaction Force and GNN

1st Yuning Wang
School of Vehicle and Mobility
Tsinghua University
Beijing, China
wangyn20@mails.tsinghua.edu.cn

2nd Zhiyuan Liu
Xingjian College
Tsinghua University
Beijing, China
liuzhiyu20@mails.tsinghua.edu.cn

3rd Haotian Lin
Xingjian College
Tsinghua University
Beijing, China
linht20@mails.tsinghua.edu.cn

4th Jinhao Li
Xingjian College
Tsinghua University
Beijing, China
lijinhao20@mails.tsinghua.edu.cn

5th Ruochen Li
School of Vehicle and Mobility
Tsinghua University
Beijing, China
lirc19@mails.tsinghua.edu.cn

6th Jianqiang Wang*
School of Vehicle and Mobility
Tsinghua University
Beijing, China
wjqlws@tsinghua.edu.cn

2023 IEEE Intelligent Vehicles Symposium (IV) | 979-8-3503-4691-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/IV55152.2023.10186793

***Abstract*—Agent trajectory prediction of traffic scenarios is a significant module of environment reasoning and autonomous vehicle decision, and the core challenge is the ability to interaction reasoning under complex scenes. Previous prediction models are either not precise enough or require massive computational costs. In this paper, we propose VIF-GNN, a novel traffic agent trajectory prediction framework based on the Virtual Interaction Force (VIF) concept and Graph Neural Network, which consists of semantic feature engineering, a subgraph encoder, a global graph, and the trajectory decoder. In particular, this method extracts vectorized features including VIF adjacent matrix from raw inputs and transfers them into graph nodes through the subgraph encoder. The global graph module obtains spatiotemporal reasoning information from four various interaction layers combined with the VIF prior knowledge. And the decoder translates the graph into trajectories of the target agent. Experiments prove that VIF-GNN could achieve precise forecasting on both single and multi-modal prediction task compared with the baselines while maintaining a relatively light parameter size scale, ensuring the real-time performance of vehicle platform applications.**

Keywords—Autonomous vehicle, trajectory prediction, interaction reasoning, graph neural network.

I. Introduction

Autonomous Vehicle (AV) has aroused extensive interest with the expectation of improving transportation safety, comfort, and efficiency. From the view of technique, AV can generally be decomposed into four modules: perception, environment assessment, decision, and control [1]. To realize AV in complicated traffic scenarios, environmental assessment is a pivotal process, reasoning the surrounding context and forming restrictions for later decisions [2, 3], and the trajectory prediction of other agents is an important component of environment assessment [4].

The methods of trajectory prediction could be divided into two categories: rule-based and deep-learning-based models. The rule-based ones predict the future motions of agents based on kinetic models, such as constant velocity and constant acceleration models. Although interpretable, they are too rudimentary to cope with the highly interactive real traffic scenarios. In recent years, deep learning (DL) has been widely applied to trajectory prediction and achieved significant processes on accuracy and prediction time duration [5]. However, although the DL-based methods have performed well in regular scenarios, there is still room for improvement in complex interaction scenes where multiple agents influence each other, such as crowded intersections with no signal lights, crossing exposed to conflict vehicles, etc. [6].

One of the core challenges for trajectory prediction in complication scenarios is the understanding and modeling of context interaction. In the previous method, the context was usually expressed by simple physical indicators such as time headway and time to collision [7, 8], which lost substantial interaction features. Some other research utilized large-scale DL models with an input with huge dimensions to extract the context features with a black-box method. Stacked Convolutional Neural Network (CNN) layers with rasterized sequential image inputs or multi-head transformers could filter part of connections among agents' behaviors [9-11]. However, because of the large number of parameters and the high requirements for bird-eye view inputs, the models are computationally expensive, resulting in difficulty in applications on real vehicles. In addition, due to the lack of a systematic understanding of the traffic environment, the generalization ability of these models is insufficient, failing to capture the essential features of context understanding.

In recent research, graph-based methods have been proven to be a good medium to represent the traffic environment [12]. Fig. 1 shows the structure of a Dynamic Directed Graph (DDG). The nodes denote the various elements within the scene including vehicles, lanes, time frame, etc., and the directed edges represent the interaction features. DDG has a strong capability of expressing relationships between agents meanwhile being expandable [13]. And together with the vectorized scenario features and Graph Neural Networks (GNN), DDG naturally extract road topology features with minor information loss and potentially less computational cost.

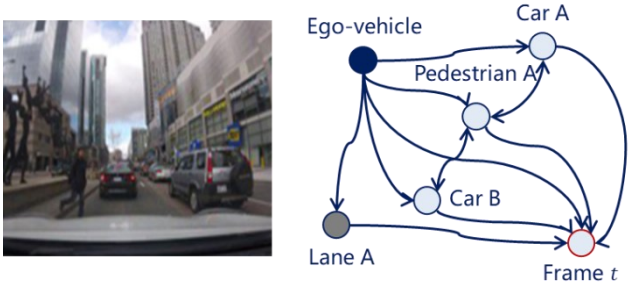

Fig. 1. Example of a dynamic directed graph for a traffic scene.

This research was funded by the National Natural Science Foundation of China with award 52131201 (the key project) and the Tsinghua University Toyota Joint Research Center for AI Technology of Automated Vehicle (TTAD 2022-06). This research was also supported by the Heye project of Xingjian College, Tsinghua University.

Authorized licensed use limited to: Univ of Calif Berkeley. Downloaded on October 18,2025 at 17:29:36 UTC from IEEE Xplore. Restrictions apply.

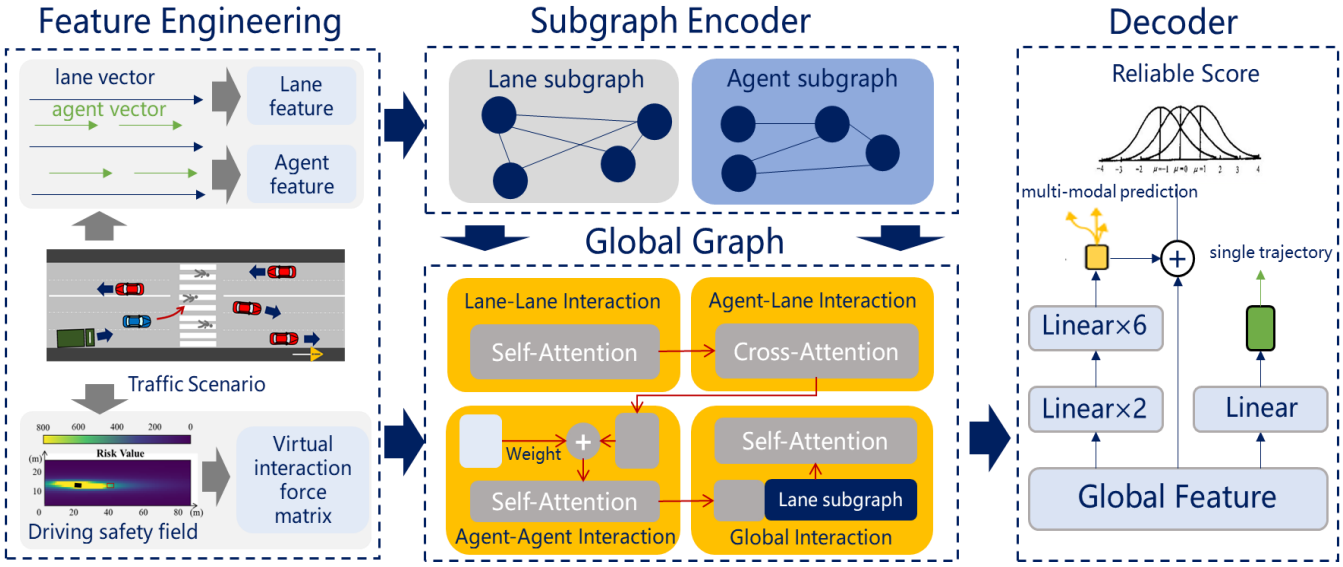

Fig. 2. The framework of VIF-GNN.

This paper proposes VIF-GNN, a novel trajectory prediction method of multi-agents based on the Virtual Interaction Force (VIF) concept and GNN architecture, as shown in Fig. 2. Our contributions are listed below:

- With the abstraction of the traffic context, we design a systemic semantic feature engineering module including the VIF adjacent matrix to extract the interaction among agents.
- Based on Graph Neural Networks, we establish a global scene graph consisting of four interaction layers and merge the VIF matrix into the network to enhance spatial interaction reasoning.
- The forecasting results show good accuracy with a relatively small scale compared with other models.

The rest of this paper is organized as follows. Section 2 reviewed the related works on feature engineering and backbone DL model design. Section 3 introduces our proposed agent trajectory prediction method from the aspect of the framework, feature engineering, subgraph and global graph design, and the decoder. In Section 4, we describe the experiment condition setting and present quantitative experiment results. Conclusions are given in Section 5.

II. Related Works

The architecture of a trajectory prediction structure can be decomposed into the feature engineering module and the backbone deep learning model. In this section, we review the latest research progress of these two aspects and extract valuable details that are worth referring to.

Feature engineering. The primary feature derived from the perception module is physical properties, which are indispensable for trajectory prediction. Yu et al. summarized that regular discrete agents' physical features include relative distance, velocity, acceleration, direction, and types [13], and Wang et al. mentioned that lanes could be represented by continuous vectors with fixed lengths [14]. Apart from fundamental characteristics, in recent years many high-order features have been utilized and found powerful for interaction relationship extraction. Attention distribution among the vehicles is a valuable and direct feature to capture connections, and Self-attention and cross-attention modules are two major modes [15, 16]. The M2I model introduced the concept of dual influencers and reactors to reflect whether a potential conflict on the road is possible [17]. STGM model preset seven sampling models with various heading angles [18] to fulfill the requirements under different situations. Such multi-head concept of feature engineering is popular in recent research. The energy field is another typical type to represent the interaction[3]. Field-based models assessed the driving risk as a continuous quantitative energy distribution on the local scenario so that by calculating the directed gradients the relative behavioral tendency was derived [19]. Wang et al. proposed the Driving Safety field to comprehensively consider the influences of common elements in traffic scenarios such as moving agents, static obstacles, curbs, etc. [20]. In summary, other than basic physical properties, high-order interactive features are helpful to understand traffic scenarios.

Backbone DL model. Sequential models used to be popular prediction backbone because of their strong ability to form temporal reasoning results. Recurrent neural networks and Long-short term memory networks are two typical examples [21, 22]. Due to the lack of spatial reasoning ability, the prediction accuracy of sequential models was limited, and in recent years scholars have turned to other backbone categories. Transformers were found to be powerful in trajectory prediction with the mechanism of extracting all correlations between agent pairs. Mercat et al. proposed a multi-head attention transformer method to consider the interactions of other agents [23]. Liu et al. used stacked transformers to further enhance spatial reasoning [24]. GNN is another backbone model which has a strong capability to analyze the interactive relationships within traffic scenarios. Compared with Transformer, GNN can not only derive agent-agent interaction but also derive agent-road interaction by abstracting lanes as subgraph nodes [25]. Combined with social pooling layers, GNN represents traffic scenes well while maintaining a relatively small amount of model parameters [26, 27].

Based on the background review above, in this work we design a novel semantic feature engineering and combine it with GNN to generate multi-modal trajectory prediction of vehicles, achieving accurate forecasts with fewer parameters.

Authorized licensed use limited to: Univ of Calif Berkeley. Downloaded on October 18,2025 at 17:29:36 UTC from IEEE Xplore. Restrictions apply.

III. METHODS

A. Model Framework

The framework of the proposed trajectory prediction model VIF-GNN is illustrated in Fig. 2, consisting of four modules: feature engineering, subgraph encoder, global graph, and trajectory decoder. First, the raw physical properties of agents and maps are transferred into vectorized features. Based on the driving safety field, we also proposed the concept of virtual interaction force (VIF) and extract the force matrix which is used as prior knowledge in the latter training process. The subgraph encoder organizes lane and agent features into the form of node features prepared for graph establishment. In the global graph module, four network blocks based on attention mechanism are designed and connected to obtain various kinds of interactive relationships. Combined with the prior knowledge derived from the VIF matrix, a global graph representing the traffic scenario context is finished. And a decoder based on a multi-layer perceptron translated the graph into continuous multi-modal agent trajectories. The details of the modules are introduced below.

B. Semantic Feature Engineering

The first process is to extract valuable features from raw agent information, which is could feature engineering. The Feature Engineering module is composed of three elements: agent feature, lane feature, and VIF adjacent matrix. The feature vector of an agent i at one frame V_i^a is defined as (1):

$$V_i^a = [\boldsymbol{loc}_i^{start} \quad \boldsymbol{loc}_i^{end} \quad \boldsymbol{v}_i \quad t_i \quad i_a] \tag{1}$$

$$\boldsymbol{loc}_i^{start} = [x^{start} \quad y^{start}],\ \boldsymbol{loc}_i^{end} = [x^{end} \quad y^{end}] \tag{2}$$

$$\boldsymbol{v}_i = [v_x \quad v_y] \tag{3}$$

where $\boldsymbol{loc}_i^{start}$ and $\boldsymbol{loc}_i^{end}$ is the location of the starting and ending point of the trajectory vector at this timestep which are two-dimension lists with the x-axis and y-axis coordinates as shown in (2), $\boldsymbol{v}_i$ is the velocity list containing x and y components as shown in (3), t_i is the timestamp, and i_a is the sorting rank position of the distance to the prediction target vehicle which is an integer from 1 to 15.

In complicated scenarios, the number of agents could be very large (e.g., in Argo 1 datasets, over 20% of events involve at least 20 agents), but most of them have little impact on the behavior of the target vehicle. Therefore, a filtering technique is needed to select those agents that are more possible to influence the motions of the vehicle, and relative distance is a simple but effective measurement. In this research, for each frame, we sort the distance between the target vehicle and other agents only retaining 15 nearest ones. Meanwhile, a sort serial mark is labeled in V_i^a to express this initial interaction relationship feature. For the events where there are fewer than 15 agents, to maintain the dimension of inputs equal, zero vectors are filled in. Hence, for each frame, the features of all agents considered consist of 15 individual vectors.

In HD maps, lanes are expressed in the form of basic geometry such as spline, which can be approximated as sequences of lane vector segments to acquire graphic representation. Consequently, the feature vector of one lane j at one frame V_j^l is defined as (4):

$$V_i^a = \left[\boldsymbol{loc}_j^{start} \quad \boldsymbol{loc}_j^{end} \quad t_j \quad \theta_j \quad its \quad sld\right] \tag{4}$$

where $\boldsymbol{loc}_j^{start}$ and $\boldsymbol{loc}_j^{end}$ are the starting and ending location vectors of the segment, t_j is the timestamp, and θ_j is the relative direction compared with the target vehicle, its and sld are binary flags to judge whether this lane segment is in an intersection and solid line, as defined in (5).

$$its = \begin{cases} 1, in\ intersection \\ 0, \quad else \end{cases} \qquad sld = \begin{cases} 1, solid \\ 0, else \end{cases} \tag{5}$$

Similarly, it is unrealistic to consider all lane segments with the region of interest. In this research, we select the nearest 40 segments as the input of the latter modules.

Other than physical properties, in this research, we innovatively designed a virtual field force adjacent matrix based on Driving Safety Field (DSF), which is a powerful feature to illustrate the relative importance among agents to the target vehicle. As mentioned above, DSF utilized an energy field model for the risk of an agent [20]. The risk energy of an agent locating (x_a, y_a) at generate on one spot (x_l, y_l) within the region of interest is calculated as (6):

$$E_{al} = E_0 \left(\frac{1}{{k_{x,0}}^2 (x_a - x_l)^2 + (y_a - y_l)^2} - \frac{1}{{r_a}^2} \right) \tag{6}$$

where E denotes the risk energy, E_0 is a constant base energy value (in our approach, $E_0 = 100$), r_a is the length of the agent, and $k_{x,0}$ is the direction gradient parameter, as demonstrated in (7) [28]:

$$k_{x,0} = \frac{\left[v_{\max} - v_l \cdot \tanh(x_a - x_l) \cdot \tanh(v_l - v_a)\right]^2}{\left[v_{\max} + v_a \cdot \tanh(x_a - x_l) \cdot \tanh(v_l - v_a)\right]^2} \tag{7}$$

where $v_{\max}$ represents the speed limit, v_l and v_a denote the velocity of the risk source and the target spot. Fig. 3 demonstrates a sample of DSF, where the black square represents the risk source agent (vehicle).

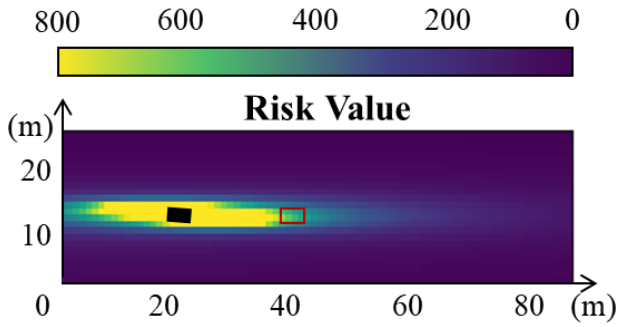

Fig. 3. An example of Driving Safety Field distribution.

Based on the field energy, we further develop the Virtual Interaction Force concept, denoting the average pooling of the field energy the risk source i generates within the range of the target vehicle j as shown in (8):

$$F_{ij} = \frac{\iint_{S_j} E_{ij} ds}{S_j} \tag{8}$$

where S_j is the cover area of the target vehicle and E_{ij} denotes the risk energy of the spots within S_j. In Fig. 2, VIF can be interpreted as the average energy intensity within the red box.

Authorized licensed use limited to: Univ of Calif Berkeley. Downloaded on October 18,2025 at 17:29:36 UTC from IEEE Xplore. Restrictions apply.

It is to be mentioned that if the force one object imposes on itself is defined as 0.

$$\overline{F_{ij}} = \frac{F_{ij} - \min\limits_{i,j} F_{ij}}{\max\limits_{i,j} F_{ij} - \min\limits_{i,j} F_{ij}} \quad (9)$$

$$\overline{\boldsymbol{F}} = \left[\overline{F_{ij}}\right] \quad (10)$$

After selecting the 15 nearest agents (including the ego vehicle), calculate VIF between all pairs and conduct normalization following (9). Then, arrange all results in the form of an adjacent matrix $\overline{\boldsymbol{F}}$ as (10).

So far, three types of features have been extracted through raw physical data. For one prediction task, assume H historical frames are considered, then agent features, lane features, and VIF matrix of past H frames are calculated and concatenated as sequential input tensors. The sizes of them are $H \times 15 \times 8$, $H \times 40 \times 8$, and $H \times 15 \times 15$.

C. Subgraph Encoder

To transfer vector inputs of agent and lane features into learnable nodes of GNN, a polyline subgraph encoder is used before establishing the global graph. Polyline subgraph is a hierarchical approach to exploit spatial and semantic locality proposed by VectorNet [29]. The architecture can be described as three stacked layers of subgraph operator in (11):

$$\boldsymbol{V}_i^{(l)} = \varphi_{rel}\left(g_{enc}\left(\boldsymbol{V}_i^{(l)}\right), \varphi_{agg}\left(\left\{g_{enc}\left(\boldsymbol{V}_j^{(l)}\right)\right\}\right)\right) \quad (11)$$

where $\boldsymbol{V}_i^{(l)}$ represents the i-th node feature for the l^{th} layer of the subgraph network. Each layer is considered a node encoder, permutation invariant aggregator, and output node feature. Node encoder g_{enc} is designed to transform individual node features, implemented by multi-layer perceptron (MLP) containing contains linear-layer, ReLU activation, and layer normalization. The aggregator φ_{agg} is the max-pooling operation intended to model the interaction between different polylines. The results of former two operators are concatenated into output node feature by φ_{rel} to constrained subgraph connectivity based on polyline groups. Three identical levels of this structure introduced are stacked together to model higher order of connectivity and to further ensure aggregation effectiveness. Finally, a max-pooling operation is applied to obtain polyline-level features p in (12):

$$p = \varphi_{agg}\left(\boldsymbol{V}_i^{(3)}\right) \quad (12)$$

Conduct polyline subgraph encoding to both agent and lane features and all outputs p are served as the input node features of the global graph.

D. Global Graph Network

Vehicle future trajectory is influenced not only by past trajectory and lane constraints (including lane direction, traffic sign, etc.) but also by surrounding agents. To model higher-order interaction for both agent-to-agent and agent-lane, we design an attention-based global graph network composed of four attention layers. Multi-head attention[16] can capture long-range dependency by taking the entire context into consideration, and reasoning spatiotemporal interactions in sequential data problems.

The structure of the global graph neural network is shown in Fig. 2, consisting of lane-lane, lane-agent, agent-agent, and global interaction layer, which are all realized by the attention model. The attention mechanism is defined as (13) and (14):

$$\mathbf{Q} = \boldsymbol{W}^{\boldsymbol{Q}}\boldsymbol{X}, \qquad \mathbf{K} = \boldsymbol{W}^{\boldsymbol{Q}}\boldsymbol{Y}, \qquad \mathbf{V} = \boldsymbol{W}^{\boldsymbol{Q}}\boldsymbol{Y} \quad (13)$$

$$\text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\boldsymbol{T}}}{\sqrt{d_{\text{k}}}}\right)\boldsymbol{V} \quad (14)$$

where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are input features, $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$ are attention query, key, and value respectively, and d_{k} is the key channel (the size of $\boldsymbol{K}$'s first dimension). For multi-headed conditions, $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$ are further split to the same size in the last dimension as heads. They are sent into the attention mechanism respectively and combined together as output. When $\boldsymbol{X}$ equals to $\boldsymbol{Y}$, it is defined as self-attention; otherwise, it is called cross-attention.

The first lane-lane interaction layer extracts the road restrictions by considering the connection relationship and whether they are drivable. The calculation follows (13) and (14), where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are both lane node features. Subsequently, the agent-lane layer could form elementary reasoning since vehicles drive along lanes. The modeling is similar to the lane-lane layer except that $\boldsymbol{X}$ are lane nodes and $\boldsymbol{Y}$ are agent nodes.

The next step is to consider the influences of other agents. In feature engineering, we have derived the VIF adjacent matrix $\overline{\boldsymbol{F}}$ in (10). To merge the prior semantic knowledge and the results of network training, we design a trainable weight w to adjust the input features automatically as demonstrated in (15), where $\boldsymbol{X}$ are the agent nodes updated by the former two layers. Weighted inputs are imported into a self-attention layer.

$$\boldsymbol{X}' = w\boldsymbol{X} + (1 - w)\overline{\boldsymbol{F}} \quad (15)$$

The final layer is global interaction, which takes all elements to derive above into comprehensive account. The global input feature is the concatenation of the third-layer outputs and the initial lane nodes. The context extraction also utilized a self-attention mechanism. Finally, a global graph representing the traffic scenario context is established.

E. Multi-modal Decoder

After the global graph block, the feature of the target agent is obtained, which contains the information on interactions between the target agent and traffic elements. The last step toward trajectory prediction is to decode the semantic context global graph into precise routes.

We apply a three-layer MLP to generate 6 predicted trajectories and another three-layer MLP to get the trust scores of all the trajectories which are formed as one-dimension vectors. For the m^{th} agent, we apply a residual block and a linear layer in the regression branch to regress the sequences of trajectory coordinates, as illustrated in (16):

$$P_m = \left\{\left(T_{m,1}^k, T_{m,2}^k, T_{m,3}^k, \ldots, T_{m,t}^k\right)\right\}, k \in [1,6] \cap N \quad (16)$$

where $T_{m,t}^k$ is the predicted m-th agent's trajectory coordinates of the k-th mode at the t^{th} time step. The single trajectory prediction follows the same method. Similarly, for the classification branch, we apply an MLP to P_m to get six-distance embedding. Then concatenate each distance embedding with the agent feature, and add a residual block and a linear layer to output six reliability scores $\{S_m^k\}$. After the process of feature engineering, subgraph extraction,

Authorized licensed use limited to: Univ of Calif Berkeley. Downloaded on October 18,2025 at 17:29:36 UTC from IEEE Xplore. Restrictions apply.

global graph establishment, and multi-modal decoding, the trajectory prediction results of agents could be derived.

IV. Experiments

A. Experiment Condition Setting

In this research, we conducted training, testing, and validation on Argoverse (Argo) tracking dataset [30]. Argo provides abundant precise trajectories of all agents within over 300k scenarios while offering a high-definition semantic map containing lane, traffic light phase, and other environmental information. Each instance is sampled in 10 HZ and trajectories are presented as 5-second-long sequences, where the former is 2 seconds as history trajectory and the latter 3 seconds as prediction ground truth. Moreover, a large number of scenarios in Argo are complicated ones such as unprotected steering, lane changing, etc., requiring a strong ability to understand complex traffic contexts. During the feature engineering of the experiment, we extracted 205,942 training data and 39,472 validating data.

Minimum Final Displacement Error (minFDE) and minimum Average Displacement error (minADE) are used as evaluation metrics of the models. MinFDE is defined as the minimum L2 distance error between the endpoint of the forecast trajectories and the ground truth. And minADE is defined as the L2 distance error between ground truth and the forecast trajectory with the lowest FDE.

The specific task is to use the history information of the past two seconds to predict the trajectory in future three seconds. In this experiment, we test the performance of our model on both single trajectory and multi-modal trajectory prediction ($k = 6$), as mentioned in (16).

B. Baseline Models

To make a comprehensive comparison with current prediction models, we select three different types of models as baselines which are Sequential models, transformer-based models, and other graph-based models. To consider the real-time requirements while applied to real vehicles, we only select models with fewer than 1000k parameters.

1) Sequential model. As introduced in Section 2, sequential models including LSTM, RNN, etc. have shown remarkable performance in dealing with sequential data. In this experiment, by referring to prior research we adapt LSTM to multi-modal prediction tasks by first using a subgraph encoder to preprocess the input agent feature [21]. The architecture is shown in Fig. 4. After an additional linear layer, LSTM layers are directly used to process the target feature. The output is further placed in multi-modal or single decoder.

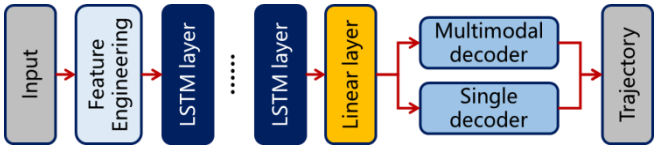

Fig. 4. The architecture of the LSTM baseline model.

2) Transformer model. Transformer-based methods are capable of capturing interaction dependency in sequential data and thus possess long-term memory. In this experiment, we adapt the transformer structure introduced in [24] to serve as a baseline, consisting of an encoder block and decoder block, as shown in Fig. 5. Agent features are sent into masked multi-head attention. All layers are associated with residue connection. Apart from the transformer decoder, we attached a multi-modal decoder to adapt a particular prediction task.

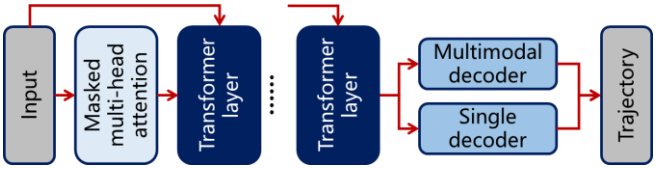

Fig. 5. The architecture of the LSTM baseline model.

3) GNN models. We also selected several other graph-based models as baselines, including VectorNet [29], TNT [31], and GOHOME [25]. These methods contained various backbone designs and techniques of feature engineering, such as graph convolution, heat map weighting, etc. Because the authors of these models have conducted experiments on the same Argo dataset, in this research we directly use the data recorded in their publications (VectorNet for single prediction and the other two for multi-modal prediction).

C. Results

After training for 50 epochs combined with linear learning rate decay and a large amount of model tuning, the model converges to the best performance on single and multi-modal trajectory prediction. The comparison results are shown in Table. I and Table. II.

TABLE I. Single Trajectory Prediction Results

Model	minADE(m)	minFDE(m)
LSTM	1.66	3.74
Transformer	1.54	3.45
VectorNet	1.66	3.67
Ours	1.40	3.06

Our method leads to the performance of a single trajectory. The minADE is 1.40 m, and the minFDE is 3.06 m, which are both the best results among all baselines. Due to the novel design of a four-layer global graph architecture to extract interactive relationships, our model emphasizes the spatial reasoning of the traffic scenarios, which is lacking in the baselines.

TABLE II. Multi-modal Trajectory Prediction Results

Model	minADE(m)	minFDE(m)
LSTM	0.90	1.65
Transformer	0.80	1.36
TNT	0.73	1.29
GOHOME	--[a]	1.26
Ours	0.74	1.17

[a] GOHOME only published minFDE results

As for multi-modal trajectory prediction, the minADE and minFDE of our model is 0.74 m and 1.17 m. Although the minADE is slightly higher than TNT, our minFDE metric is 9.3% better than TNT, making the ending point of the trajectory more precise. In summary, our model achieves good

Authorized licensed use limited to: Univ of Calif Berkeley. Downloaded on October 18,2025 at 17:29:36 UTC from IEEE Xplore. Restrictions apply.

performances on both metrics compared with current small-scale multi-modal trajectory prediction models. To be mentioned, the final parameter size of our best model is 438k.

TABLE III. ABLATION RESULTS OF VIRTUAL INTERACTION FORCE

w	minADE(m)	minFDE(m)
1	0.78	1.22
0.25	0.75	1.19
0.5	0.74	1.17

To further verify the proposed VIF concept, we conduct an ablation study to test its impact on it the prediction precision. In the backbone model, the weight between the VIF matrix and the attention result w is automatically learned through training process, which is about 0.5 in the final results. To illustrate the effect of VIF concept, in this ablation experiment we adjust it into preset values to test the performance difference. The closer w is to 1, the less force matrix is considered in the model, and the results are recorded in Table. III.

The results of ablation study prove that the application of the VIF adjacent matrix improves the trajectory prediction performance effectively. When no prior knowledge is used, the minADE and minFDE are 0.04 m and 0.05 m worse than the best model respectively. The field energy and force express the interactions between agents, offering significant spatial reasoning information to the global graph.

Fig. 6 illustrates some typical prediction examples of complicated scenarios, including car following, intersection, unprotected steering, etc. The black lines are lanes, red point denotes the target agent, the blue points denote other related agents, the green lines denote the multi-modal predictions of the target agent's trajectory, and the red line denotes the ground truth. As shown in Fig. 6, our model could make appropriate predictions on various kinds of scenarios considering interactions and conflicts.

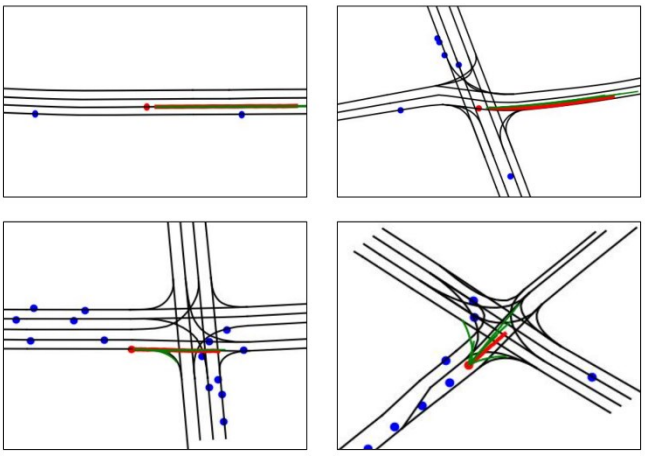

Fig. 6. A typical prediction example of VIF-GNN.

V. CONCLUSIONS AND DISCUSSION

This paper presents a novel method of traffic agent trajectory prediction model named VIF-GNN. Based on Virtual Interaction Force and other semantic feature engineering, the original scenes are transferred into vectorized context information. We also design a global graph based on GNN consisting of four interaction layers.

The experiments on Argo dataset show that VIF-GNN can achieve precise prediction accuracy in both single and multi-modal tasks compared with the baselines. For single trajectory prediction, the minADE and minFDE are 9.1% and 11.3% better, and for multi-modal the minFDE is 9.3% better with a roughly identical minADE. Examples show that VIF-GNN manages to conduct forecasting under complicated scenarios. The ablation study also verified the positive impact of VIF.

In the future, we will further test the generalization ability of VIF-GNN by enhancing experiments on various datasets with different features (highway, roundabout, etc.) and various downstream tasks other than trajectory prediction such as risk assessment, behavioral decision, etc.

REFERENCES

[1] Z.-X. Xia *et al.*, "A Human-Like Traffic Scene Understanding System: A Survey," *IEEE Ind. Electron. Mag.,* vol. 15, no. 1, pp. 6-15, 2020.

[2] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annual Review of Control, Robotics, and Autonomous Systems,* vol. 1, pp. 187-210, 2018.

[3] W. Wang, L. Wang, C. Zhang, C. Liu, and L. Sun, "Social interactions for autonomous driving: A review and perspectives," *Found. Trends Robot.,* vol. 10, no. 3-4, pp. 198-376, 2022.

[4] V. Ramanishka, Y.-T. Chen, T. Misu, and K. Saenko, "Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 7699-7707.

[5] F. Leon and M. Gavrilescu, "A review of tracking and trajectory prediction methods for autonomous driving," *Mathematics,* vol. 9, no. 6, p. 660, 2021.

[6] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings, and A. Mouzakitis, "Deep learning-based vehicle behavior prediction for autonomous driving applications: A review," *IEEE trans. Intell. Transp. Syst.,* vol. 23, no. 1, pp. 33-47, 2020.

[7] A. Abdelraouf, M. Abdel-Aty, Z. Wang, and O. Zheng, "Trajectory Prediction for Vehicle Conflict Identification at Intersections Using Sequence-to-Sequence Recurrent Neural Networks," *arXiv preprint arXiv:2210.08009,* 2022.

[8] Y. Wu, J. Hou, G. Chen, and A. Knoll, "Trajectory prediction based on planning method considering collision risk," in *2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM)*, 2020, pp. 466-470.

[9] Y. Chai, B. Sapp, M. Bansal, and D. Anguelov, "Multipath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction," *arXiv preprint arXiv:1910.05449,* 2019.

[10] J. Hong, B. Sapp, and J. Philbin, "Rules of the road: Predicting driving behavior with a convolutional model of semantic interactions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 8454-8462.

[11] L. L. Li *et al.*, "End-to-end contextual perception and prediction with interaction transformer," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 5784-5791.

[12] B. Liu *et al.*, "Spatiotemporal relationship reasoning for pedestrian intent prediction," *IEEE Robot. Autom. Lett.,* vol. 5, no. 2, pp. 3485-3492, 2020.

[13] S.-Y. Yu, A. V. Malawade, D. Muthirayan, P. P. Khargonekar, and M. A. Al Faruque, "Scene-graph augmented data-driven risk assessment of autonomous vehicle decisions," *IEEE trans. Intell. Transp. Syst.,* 2021.

[14] J. Wang, T. Ye, Z. Gu, and J. Chen, "LTP: Lane-Based Trajectory Prediction for Autonomous Driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 17134-17142.

[15] K. Messaoud, N. Deo, M. M. Trivedi, and F. Nashashibi, "Trajectory prediction for autonomous driving based on multi-head attention with joint agent-map representation," in *2021 IEEE Intelligent Vehicles Symposium (IV)*, 2021, pp. 165-170.

[16] L. Ye, Z. Wang, X. Chen, J. Wang, K. Wu, and K. Lu, "GSAN: Graph self-attention network for learning spatial-temporal interaction representation in autonomous driving," *IEEE Internet Things J.,* 2021.

[17] Q. Sun, X. Huang, J. Gu, B. C. Williams, and H. Zhao, "M2I: From Factored Marginal Trajectory Prediction to Interactive Prediction," in

Authorized licensed use limited to: Univ of Calif Berkeley. Downloaded on October 18,2025 at 17:29:36 UTC from IEEE Xplore. Restrictions apply.

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 6543-6552.

[18] Z. Zhong, Y. Luo, and W. Liang, "STGM: Vehicle Trajectory Prediction Based on Generative Model for Spatial-Temporal Features," *IEEE trans. Intell. Transp. Syst.,* 2022.

[19] F. Bounini, D. Gingras, H. Pollart, and D. Gruyer, "Modified artificial potential field method for online path planning applications," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 180-185.

[20] J. Wang, J. Wu, and Y. Li, "The driving safety field based on driver–vehicle–road interactions," *IEEE trans. Intell. Transp. Syst.,* vol. 16, no. 4, pp. 2203-2214, 2015.

[21] F. Altché and A. de La Fortelle, "An LSTM network for highway trajectory prediction," in *2017 IEEE 20th international conference on intelligent transportation systems (ITSC)*, 2017, pp. 353-359.

[22] S. Choi, J. Kim, and H. Yeo, "Attention-based recurrent neural network for urban vehicle trajectory prediction," *Procedia Computer Science,* vol. 151, pp. 327-334, 2019.

[23] J. Mercat, T. Gilles, N. El Zoghby, G. Sandou, D. Beauvois, and G. P. Gil, "Multi-head attention for multi-modal joint vehicle motion forecasting," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 9638-9644.

[24] Y. Liu, J. Zhang, L. Fang, Q. Jiang, and B. Zhou, "Multimodal motion prediction with stacked transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 7577-7586.

[25] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde, "Gohome: Graph-oriented heatmap output for future motion estimation," *arXiv preprint arXiv:2109.01827,* 2021.

[26] M. Liang *et al.*, "Learning lane graph representations for motion forecasting," in *European Conference on Computer Vision*, 2020, pp. 541-556.

[27] 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, 2019.

[28] Y. Wang, S. Yang, J. Li, S. Xu, and J. Wang, "An Emergency Driving Intervention System Designed for Driver Disability Scenarios Based on Emergency Risk Field," *Int. J. Environ. Res. Public Health,* vol. 20, no. 3, p. 2278, 2023. [Online]. Available: https://www.mdpi.com/1660-4601/20/3/2278.

[29] J. Gao *et al.*, "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, pp. 11525-11533.

[30] M.-F. Chang *et al.*, "Argoverse: 3d tracking and forecasting with rich maps," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2019, pp. 8748-8757.

[31] H. Zhao *et al.*, "Tnt: Target-driven trajectory prediction," in *Conference on Robot Learning*, 2021, pp. 895-904.

Authorized licensed use limited to: Univ of Calif Berkeley. Downloaded on October 18,2025 at 17:29:36 UTC from IEEE Xplore. Restrictions apply.