Mixture of Experts based Scenario Prediction for Motion Forecasting

Zongwei Jia[1,2] and Peijun Ye[1(✉)]

[1] State Key Laboratory of Multimodal Artificial Intelligence Systems
Institute of Automation,
Chinese Academy of Sciences
Beijing, China
[2] The School of Artificial Intelligence
University of Chinese Academy of Sciences
Beijing, China
{jiazongwei2022,peijun.ye}@ia.ac.cn

Abstract. Vehicle motion prediction is of great significance in human-machine shared driving, as it is a basis for the virtual driver to make suitable driving strategies so as to better assist the human driver or even complete its own autonomous driving. While recent studies have achieved good results by applying an intention prediction module into the multi-modal trajectory generation strategy, they mostly give supervision between intention proposals and final predicted points with insufficient alignment method. This leads to a relatively low accuracy of the vehicle motion prediction. To tackle this challenge, we propose a graph-based model with mixture of experts (MOE) based intention prediction module, introducing an efficient scenario-based intention prediction mechanism to improve the performance of intention prediction task. Several testing was conducted on the Argoverse motion forecasting dataset, which showed that our model excels in predicting trajectories for multiple agents. And the ablation experiments have verified the efficency of our method.

Keywords: Motion Forecasting · Deep Learning · Graph Neural Networks

1 INTRODUCTION

In the context of human-machine shared driving, the virtual driver needs to provide prescriptions to the human driver as an intelligent assistant, or even performs self-driving when the human take over is absent. Its objective is to make correct as well as human-like strategies in an uncertain and dynamic environment so that the vehicle can successfully reach the designated waypoint. Apart from the intrinsic cognitive characteristics such as the personality and driving style learning from the human driver[1][2], the motion prediction from surrounded vehicles in a particular traffic scenario is another vital aspect for virtual driver to select a suitable action. In such a task, the virtual driver needs to deal with the temporal and spatial interactions, the impact of traffic signs, and the high-quality encoding of the historical motion information from different traffic participants, so as to accurately predict their possible future motions.

One of the tasks of virtual driver is to assist with driving and reduce the cognitive load on human drivers. Trajectory prediction can help anticipate the motion of surrounding vehicles, enabling intelligent decision-making for driving strategies and reducing the frequency and intensity of cognitive decision-making by human drivers. For autonomous driving systems, accurately predicting the movement of surrounding traffic participants is an important task, which is of great significance for guiding future safe driving behavior. This task causes complexity and has many challenges. It is necessary to deal with the time and space interaction of different types of traffic participants, consider the impact of traffic markers and traffic participants, and take into account the high-quality encoding of the historical motion information of traffic participants, so as to accurately predict the possible future motion of traffic participants and guide the automatic driving system to make safety decisions.

Basically, current learning-based methods for vehicle motion prediction can be roughly categorized into two types. One type of work applies many achievements in the field of computer vision. The traffic scene at each moment is rendered in the form of a bird eye view[3], and the CNN [4] is used to process. However, the rendered results lose a lot of original information. In order to ensure accuracy, it also takes up a lot of running space.

Due to high demand for computing and storage resources of visual methods. The other type of work mainly establishes various interactive relationships in the traffic environment in the form of graphs. Through the GNN [5] to process the abstract graph data, the decoder decodes the representation of the environment learned by the model, and obtains the prediction of the future trajectory of the vehicle. Some frameworks, such as VAE[6][7][8], GAN[9][10] widely applied in different interpolation and prediction tasks can also improve the performance in this task. In addition, in order to obtain more accurate prediction results, some works applied proposal module[11][12], which decouples the complex task into a target point prediction and trajectory interpolation task.

Our contributions are as follows:

- We present one graph neural networks model successfully capture the complex interactions of the traffic participants. To the best of our knowledge, we are the first to utilize the mixture of expert module to facilitate the efficiency of intention prediction task, enhancing the performance of the motion forecasting model.
- We employ the Mixture of Experts approach to self-supervised learning of category-specific features under various traffic scenarios. Given the embedded representation of a scenario, the routing network within MOE directs samples to distinct expert networks for processing, based on the learned category-specific features.
- Several ablation experiments were conducted on the argoverse dataset, which have demonstrated the efficiency of our approach.

2 RELATED WORK

2.1 Latent Representation Sampling

There are two main approaches to address this problem

Rasterized Scene using CNN[4] The rasterized scene approach involves creating an image-like representation of the environment, where dynamic agents are encoded as pixels or channels in an image. This representation allows for the use of CNN, which are adept at handling spatial data and have achieved remarkable success in various computer vision tasks. The input to the CNN consists of the rasterized scene, and the output is the predicted future trajectories of the agents. This approach has shown promising results in motion forecasting tasks, particularly in the context of autonomous driving.

Graph-based Approach using GNN[5] The graph-based approach[13][14][15][16] models the interactions between dynamic agents using a graph structure. Each agent is represented as a node in the graph, and the edges between nodes capture the relationships between agents, such as their spatial proximity or relative movement. GNN are then used to process this graph-structured data, enabling the model to learn complex interactions and dependencies between agents. The input to the GNN is the graph representation of the scene, and the output is the predicted future trajectories of the agents. This approach has shown strong performance in various motion forecasting applications.

2.2 Intention Prediction

Intention prediction[11][12][17] plays a crucial role in motion forecasting tasks, especially for autonomous driving systems. Integrating intention prediction into motion forecasting models can significantly improve their performance, as it enables the models to capture the underlying intentions of agents.

Various approaches have been proposed for intention prediction. There are many methods to generate the intention point[11][12], which can be generated by decoding the environmental variables directly through the neural network, or by generating a set of GMM[17] through the neural network to represent the intention point. In addition, there are also works to generate the intention point by selecting the points near the road traffic line. The methods mentioned above have all, through various forms of expression, encoded intention for future automotive trajectories. However, these methods have not been able to learn potential intention in a self-supervised manner, requiring to a certain extent the supervision of manually annotated labels. In this context, we introduce a self-supervised learning mechanism for scenario intention based on a mixture of expert network. This mechanism is capable of autonomously optimizing the encoding of future automotive trajectory intention according to the task at hand, yielding favorable results.

2.3 Multi-Modal Prediction

Multi-modal prediction is an essential aspect of motion forecasting tasks, particularly for autonomous driving systems, where the ability to predict multiple plausible futures for other agents in the environment is crucial for robust and safe decision-making. The goal of multi-modal prediction is to estimate a diverse set of potential trajectories that agents might follow, considering the inherent uncertainty and ambiguity in their behavior and the surrounding context.

In motion forecasting tasks, multi-modal prediction aims to capture the various possible outcomes that can arise from agents' interactions with the environment and other agents. To achieve this, several approaches have been proposed, leveraging techniques such as GAN [9][10], and VAE [6][8]. These methods allow models to learn a distribution over future trajectories, enabling them to generate multiple plausible predictions [10][18][19][20].

2.4 Mixture of Experts

The Mixed Expert Networks are capable of allocating different types of samples to suitable expert networks for processing through a routing network, enabling the model to avoid compressing the knowledge of various sample types into a single network during the learning process. By distributing the processing of knowledge pertaining to different sample types, the model's performance can be enhanced more efficiently.

This network paradigm has been extensively applied in fields such as Natural Language Processing and Computer Vision, including works like Switch Transformer[21] and Gshard[22], where Mixed Expert Networks have demonstrated significant sparse learning effects. These effects allow the model to achieve superior performance under the premise of consistent operational parameters. Switch-Nerf[23] utilizes Mixed Expert Networks, enabling different experts to learn and model various types of scenes within large-scale scene reconstruction tasks, yielding impressive results. To our current knowledge, no work has considered the Mixed Expert Networks approach in the motion prediction domain. We have designed a motion prediction Mixed Expert Network capable of distributed processing of different traffic scene types. By specifically addressing the varying traffic scenarios in which the objects to be predicted are situated, we have developed a more efficient motion prediction model.

3 METHOD

Briefly, the proposed model, as shown in the Fig.1, uses graph neural network to encode geometric information of traffic participant tracks and environmental information, then obtains intention proposals through interactive coding module and scenario intention prediction module, and then obtains accurate future trajectory prediction by trajectory decoder module. Finally, the alignment method is applied to alleviate the problem mentioned above.

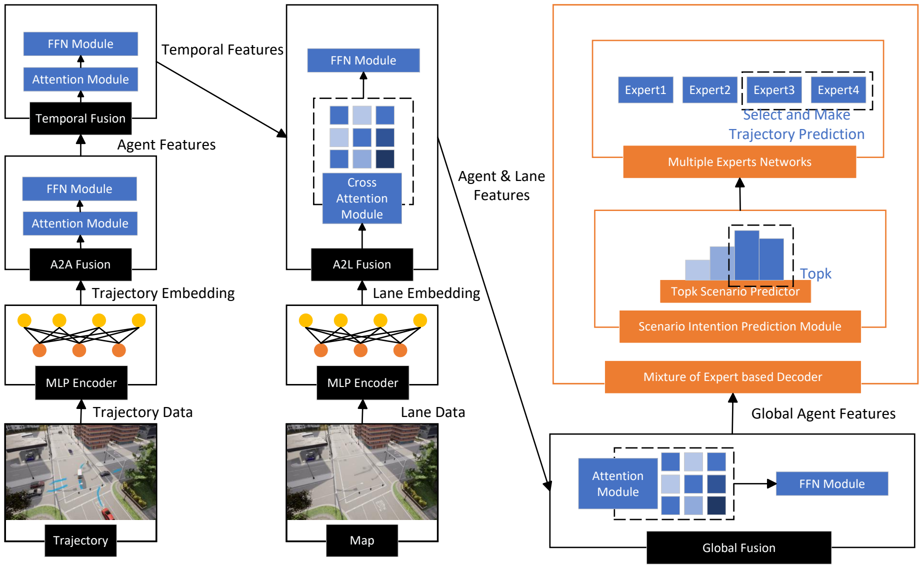

Fig. 1. Model Framework Description

3.1 Problem Formulation

Consider a set of N vehicles in a scene, indexed by $i \in \{1, 2, \ldots, N\}$. Let $x_i^{(t)} \in \mathbf{R}^2$ denote the 2D position of vehicle i at time t. The trajectory of vehicle i up to time t is given by $X_i^{(1:t)} = \{x_i^{(1)}, x_i^{(2)}, \ldots, x_i^{(t)}\}$. The goal of vehicle motion forecasting is to predict the future trajectory of each vehicle for a horizon of T time steps, i.e., $Y_i^{(t+1:t+T)} = \{x_i^{(t+1)}, x_i^{(t+2)}, \ldots, x_i^{(t+T)}\}$, based on their historical trajectories and possibly other contextual information.

3.2 Geometric Representation Encoder

Motion forecasting tasks involve predicting future trajectories of vehicles given their past positions and map information. In this approach, we encode both map information and trajectory information in a graph format, leveraging Transformers for prediction. The map information and trajectory information are encoded by MLPs respectively.

Vehicle Trajectories as Nodes Given a vehicle trajectory $Traj_i$ consisting of a sequence of (x, y) coordinates for time steps $t \in \{1, 2, \ldots, T\}$:

$$Traj_i = \{(x_{i1}, y_{i1}), (x_{i2}, y_{i2}), \ldots, (x_{iT}, y_{iT})\} \tag{1}$$

we can represent each trajectory as a node in the graph. The node features $\mathbf{F}_{Traj_i}$ can include the past positions, angles, and other descriptors of the vehicle state, where $attr_i$ represents additional attributes for vehicle i:

$$\mathbf{F}_{Traj_i} = Concat\{(x_{i1}, y_{i1}, attr_{i1}), \ldots, (x_{iT}, y_{iT}, attr_{iT})\} \tag{2}$$

Map Information as Nodes Map information M consists of various types of data, such as lane centerlines, road boundaries, and intersection regions. We can represent each map element as a node in the graph.

For example, if we consider lane centerlines, let L_j be a lane centerline consisting of a sequence of (x, y) coordinates:

$$L_j = \{(x_{j1}, y_{j1}), (x_{j2}, y_{j2}), \ldots, (x_{jN}, y_{jN})\} \tag{3}$$

where N is the number of points in the centerline.

The node features $\mathbf{F}_{L_j}$ can include the positions of the centerline points, and other lane descriptors:

$$\mathbf{F}_{L_j} = Concat\{(x_{i1}, y_{i1}, attr_{i1}), \ldots, (x_{iT}, y_{iT}, attr_{iT})\} \tag{4}$$

$attr_j$ represents additional attributes for lane j.

The encoding process involves passing the trajectory graph and the map graph, consisting of their respective node features $\mathbf{F}_{Traj}$ and $\mathbf{F}_L$ through separate MLPs:

$$\mathbf{H}_T = MLP_{Traj}(\mathbf{F}_{Traj}) \tag{5}$$

$$\mathbf{H}_L = MLP_L(\mathbf{F}_L) \tag{6}$$

where $\mathbf{H}_{Traj}$ and $\mathbf{H}_L$ are the output hidden representations of the nodes in the trajectory and map graph respectively.

3.3 Interaction Fusion Blocks

Transformer Architecture The Transformer [24] is an attention-based architecture designed for sequence-to-sequence tasks. It consists of an encoder and a decoder, each composed of multiple identical layers. In our scenario, we focus on the encoder part of the Transformer to learn interactions among map nodes and trajectory nodes.

Each layer in the Transformer encoder consists of two main components: a multi-head self-attention mechanism and a position-wise feed-forward network. Additionally, there are residual connections around each of the two components, followed by layer normalization.

Multi-Head Self-Attention The multi-head self-attention mechanism allows the model to jointly attend to information from different representation subspaces at different positions. Given a set of input vectors, the self-attention mechanism computes an output vector for each input vector by taking a weighted sum of all input vectors, where the weights are determined by the compatibility of each pair of input vectors. The multi-head self-attention is computed as follows:

$$MultiHead(Q, K, V) = Concat(head_1, \ldots, head_H)W^O \tag{7}$$

where Q, K, and V are the query, key, and value matrices, respectively, and H is the number of attention heads. Each head is computed as:

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{8}$$

where W_i^Q, W_i^K, and W_i^V are the linear projection matrices for the i-th head. The attention function, $Attention(Q, K, V)$, is the scaled dot-product attention:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{9}$$

where d_k is the dimension of the key vectors.

Position-wise Feed-Forward Network The position-wise feed-forward network consists of two linear transformations with a ReLU activation function in between:

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \tag{10}$$

where W_1, W_2, b_1, and b_2 are the learned weights and biases.

Encoding Process

$$\mathbf{Z}_{T,t} = Transformer(\mathbf{H}_{T,t}, \mathbf{H}_{T,t}) \tag{11}$$

$$\mathbf{Z}_T = TemporalEncoder(\mathbf{Z}_{T,t}, \mathbf{Z}_{T,t}) \tag{12}$$

$$\mathbf{Z}_L = Transformer(\mathbf{Z}_T, \mathbf{H}_L) \tag{13}$$

$$\mathbf{Z}_{global} = Transformer(\mathbf{Z}_L, \mathbf{Z}_L) \tag{14}$$

3.4 Scenario-based Intention Prediction Module

In this approach, we extend the previously described method with a mixture of experts mechanism as a decoder module, implemented as MLPs and Router Networks. These modules are applied after the global interaction learning stage to generate scenario intention predictions and decode them into final trajectory predictions.

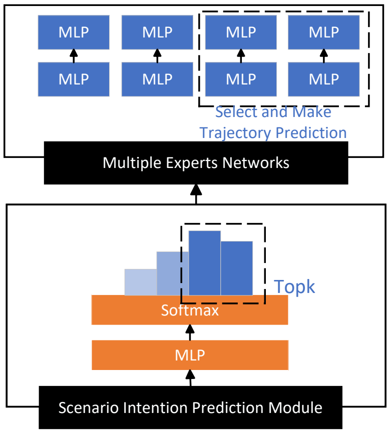

Fig. 2. Mixture of Expert based Scenario Intention Prediction Module

Scenario Intention Proposal Module in Mixture of Experts Mechanism As demonstrated in Figure 2, the scenario intention proposal module takes the global interaction output $\mathbf{Z}_{global}$ as input and generates a set of candidate scenario proposals. Then the router network directs the samples towards various expert networks can tackling different scenarios for processing based on the generated scenario proposals. In actuality, the scenario intention proposal module is characterized by two parameters, **N** and **K**, where **N** signifies the total number of expert networks. This parameter, to a certain extent, represents the sparsity of the network as well as the diversity of types that the network can process. On the other hand, **K** denotes the number of expert networks selected for sample processing after the routing network completes the allocation process and determines the proposal degree for each expert network, thereby selecting the top-ranked proposals:

$$\mathbf{T}_{prop} = Router_{n,k}(\mathbf{Z}_{global}, \mathbf{Z}_L) \tag{15}$$

Decoder Module The decoder module is responsible for refining the trajectory proposals generated by the target proposal module. It takes both the global interaction output $\mathbf{Z}_{global}$ and the lane representations $\mathbf{Z}_L$ as input and produces the final trajectory predictions. We aspire to convey a novel perspective wherein for each contextual embedding, a routing network selects several expert networks for processing. This approach ultimately results in each expert network acquiring relatively unique experiential characteristics. Such a method effectively decouples tasks, enabling the intrinsic feature attributes of these tasks to be processed in a categorized manner. Consequently, this significantly enhances model performance:

$$\mathbf{T}_{pred} = Expert_{topk}(\mathbf{Z}_{global}, \mathbf{Z}_L) \tag{16}$$

3.5 Loss Functions

In this section, we describe the loss function for trajectory prediction in the context of motion forecasting tasks. The loss function aims to minimize the discrepancy between the predicted trajectories and the ground truth trajectories.

$$\mathcal{L} = \mathcal{L}_{reg(1/2)} + \mathcal{L}_{cls} + \mathcal{L}_{balance} \tag{17}$$

Regression Loss The regression loss $\mathcal{L}_{reg1}$ measures the distance between the predicted trajectories and the ground truth trajectories. It is defined as the average Euclidean distance between corresponding points in the predicted and ground truth trajectories. Note that q in $\mathcal{L}_{reg2}$ is the Laplace distribution function, following [25]. The loss function mentioned below are both tested in ablation experiments:

$$\mathcal{L}_{reg1}(\mathbf{T}_{pred}, \mathbf{T}_{gt}) = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}|\mathbf{p}_{pred,i,t} - \mathbf{p}_{gt,i,t}| \tag{18}$$

$$\mathcal{L}_{reg2}(\mathbf{T}_{pred}, \mathbf{T}_{gt}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\log q(\mathbf{p}_{gt,i,t}|\mathbf{p}_{pred,i,t}, b_{i,t}) = \log 2b + \frac{|\mathbf{p}_{gt,i,t} - \mathbf{p}_{pred,i,t}|}{b_{i,t}} \tag{19}$$

Classification Loss The classification loss $\mathcal{L}_{cls}$ measures the discrepancy between the predicted trajectory classes and the ground truth trajectory classes. It is defined as the cross-entropy loss between the predicted class probabilities and the ground truth class labels. The main purpose of this loss function is to enhance the consistency between the highest-confidence trajectory predicted by the model in multimodal forecasting and the ground truth. Through this approach, the aim is to ensure that among the various potential future trajectories provided by the model, the one with the highest predicted confidence aligns more closely with the actual observed trajectory:

$$\mathcal{L}_{cls}(\mathbf{T}_{pred}, \mathbf{T}_{gt}) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}\mathbf{y}_{gt,i,c}\log \mathbf{y}_{pred,i,c} \tag{20}$$

Load Balance Loss In $\mathcal{L}_{balance}$ the m_e is mean gates per expert, c_e/S is the fraction of input routed to each expert. The objective of the load balance loss is to ensure equitable distribution of sample volumes across various expert networks, preventing scenarios where only a few expert networks are selected, leading to a lack of training for others. This approach facilitates each expert network in learning relatively unique features, thereby aiding in enhancing the overall performance of the model:

$$\mathcal{L}_{balance} = m_e \frac{c_e}{S} \tag{21}$$

4 EXPERIMENTAL EVALUATION

4.1 Datasets

The proposed method is evaluated on the Argoverse Motion Forecasting dataset[26], which provides rich contextual information, including high-definition maps with lane centerlines, as well as trajectories of multiple agents in the scene. Each trajectory in the dataset consists of 2 seconds of historical data and 3 seconds of future data, sampled at 10 Hz. The historical data is used to predict the future trajectories of the agents. In addition to vehicle trajectories, the dataset also provides high-definition maps with accurate lane centerline information, which can be utilized for incorporating map context into prediction models.

The dataset is split into three subsets: a train set, a validation set, and a test set. The train set consists of 205,942 sequences, the validation set contains 39,472 sequences, and the test set has 78,143 sequences. Each sequence represents a unique scene, with multiple agent trajectories and associated map data. The Argoverse Motion Forecasting benchmark evaluates the performance of vehicle motion forecasting models using various evaluation metrics, such as Average Displacement Error, Final Displacement Error and Miss Rate.

4.2 Metrics

In this section, we describe various evaluation metrics used for trajectory prediction tasks: Average Displacement Error (ADE), Final Displacement Error (FDE), Miss Rate (MR), minADE@K, minFDE@K, and MR@K.

4.3 Experiment Details

The model is trained with a batch size of 64 for 32 epochs. The learning rate is initialized at $1e-3$. We use the Adam optimizer [27] with a weight decay of $1e-3$ for regularization. The CosineAnnealing learning rate scheduler was applied in training steps.

4.4 Results Analysis

The results obtained from the validation set and the ablation study conducted on the validation set are presented in Table 1 and Table 2. A meticulous analysis of Table 1 reveals critical insights into the performance of our proposed mixture of experts based scenario intention prediction mechanism. It is evident that employing our proposed mechanism in motion prediction tasks significantly

Table 1. Results on the validation set of the Argoverse dataset

Model	N	K	minADE@6	minFDE@6	minMR@6
ours	1	1	0.693	1.040	0.102
ours	5	1	0.708	1.080	0.116
ours	10	1	0.713	1.110	0.121
ours	5	2	**0.685**	**1.020**	**0.099**
MultiLane[28]			0.84	1.42	0.11
Luo et. al.[29]			1.05	2.06	-
TNT[11]			0.72	1.29	**0.09**
LaneGCN[13]			**0.71**	1.08	0.10
DenseTNT[12]			0.73	**1.05**	-

outperforms the models that listed in Table 1, regardless of the combination of hyperparameters selection. Incorporating our mixture of experts based scenario intention prediction mechanism into the training process leads to superior performance across all official evaluation metrics on the Argoverse dataset.

The closer examination of the results shows that, for the official evaluation metrics of the Argoverse dataset, the introduction of $\mathcal{L}_{reg2}$ alone yields promising performance. In fact, by allocating a variance estimate for each position within the trajectory prediction outcomes, the efficacy of the model has been substantially enhanced. This methodology allows the model to avoid the compulsion to approximate a point as necessitated in $\mathcal{L}_{reg1}$, thus providing predictions with a certain degree of uncertainty variance effectively mitigates issues associated with overly rigid problem-solving approaches. By introducing a more softened loss function, the robustness of the model is augmented, thereby improving the overall performance of the model. Regarding the loss function $\mathcal{L}_{cls}$, it serves a supervisory role in the learning of confidence in multimodal trajectory predictions, evidencing that the application of this loss function can significantly enhance model performance. On the other hand, $\mathcal{L}_{balance}$ is capable of optimizing the balance in sample allocation, ensuring that all networks within the multi-expert framework are well-trained, thereby achieving the objective of self-supervised construction of scene classification.

Table 2. Ablation Studies on the validation set

$\mathcal{L}_{reg1/2}$	$\mathcal{L}_{cls}$	$\mathcal{L}_{balance}$	minADE@6	minFDE@6	minMR@6
1	✓	✓	0.732	1.080	0.111
2	✗	✓	0.720	1.130	0.125
2	✓	✗	0.691	1.040	0.101
2	✓	✓	**0.685**	**1.020**	**0.099**

Regarding the hyperparameters N and K, it is observed that a relatively superior model evaluation performance is predominantly concentrated around a median value of N. Both an insufficient and an excessive number of expert networks can lead to certain degrees of performance degradation. Moreover, an overly high quantity of expert networks may result in an excessively sparse network configuration, thereby adversely affecting model performance. As for the parameter K, it is generally observed that selecting up to two expert networks as proposals tends to demand significant resource allocation. It can be seen that the involvement of more expert networks in the model inference process enables the integration of a broader array of local key features.

5 CONCLUSIONS

In this work, we propose a method for performing scenario based intention prediction of vehicle motion forecasting tasks. By encoding road topology information, motion information of the ego vehicle and surrounding vehicles through Graph Neural Networks, and conducting global interaction learning via Transformers, we perform scenario based intention prediction to send different types of traffic scenario embedding to the specific expert networks. We then integrate the aforementioned information to predict multimodal trajectories, while simultaneously applying our proposed scenario based intention prediction methods. Our approach achieves favorable results on the introduced evaluation metrics of the argoverse dataset. The experimental results demonstrate that our proposed mixture of expert based scenario intention prediction method has achieved its design objectives, realizing improved model prediction performance.

In future work, we will further explore the application of mixture of expert based scenario intention prediction mechanisms in motion forecasting tasks, and investigate more effective model encoding structures to enhance the performance of the model in motion forecasting tasks.

References

1. Ye, P., Wang, X., Zheng, W., Wei, Q. & Wang, F. Parallel cognition: Hybrid intelligence for human-machine interaction and management. *Frontiers Of Information Technology & Electronic Engineering.* **23**, pp. 1765-1779 (2022)
2. Ye, P., Wang, X., Xiong, G., Chen, S. & Wang, F. TiDEC: A two-layered integrated decision cycle for population evolution. *IEEE Transactions On Cybernetics.* **51**, pp. 5897-5906 (2020)
3. Gilles, T., Sabatini, S., Tsishkou, D., Stanciulescu, B. & Moutarde, F. Home: Heatmap output for future motion estimation. *2021 IEEE International Intelligent Transportation Systems Conference.* pp. 500-507 (2021)
4. Lecun, Y., Bottou, L., Bengio, Y. & Haffner, P. Gradient-based learning applied to document recognition. *Proceedings Of The IEEE.* **86**, pp. 2278-2324 (1998)
5. Scarselli, F., Gori, M., Tsoi, A., Hagenbuchner, M. & Monfardini, G. The graph neural network model. *IEEE Transactions On Neural Networks.* **20**, pp. 61-80 (2008)
6. Kingma, D. & Welling, M. Auto-Encoding Variational Bayes. *International Conference On Learning Representations.* (2014)

7. Ye, P., Zhu, F., Lv, Y., Wang, X. & Chen, Y. Efficient Calibration of Agent-Based Traffic Simulation Using Variational Auto-Encoder. *IEEE International Conference On Intelligent Transportation Systems.* pp. 3077-3082 (2022)
8. Sohn, K., Lee, H. & Yan, X. Learning structured output representation using deep conditional generative models. *Advances In Neural Information Processing Systems.* **28** (2015)
9. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. Generative Adversarial Nets. *Advances In Neural Information Processing Systems.* **27** (2014)
10. Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S. & Alahi, A. Social gan: Socially acceptable trajectories with generative adversarial networks. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition.* pp. 2255-2264 (2018)
11. Zhao, H., Gao, J., Lan, T., Sun, C., Sapp, B., Varadarajan, B., Shen, Y., Shen, Y., Chai, Y., Schmid, C. & Others Tnt: Target-driven trajectory prediction. *Conference On Robot Learning.* pp. 895-904 (2021)
12. Gu, J., Sun, C. & Zhao, H. Densetnt: End-to-end trajectory prediction from dense goal sets. *Proceedings Of The IEEE/CVF International Conference On Computer Vision.* pp. 15303-15312 (2021)
13. Liang, M., Yang, B., Hu, R., Chen, Y., Liao, R., Feng, S. & Urtasun, R. Learning lane graph representations for motion forecasting. *European Conference on Computer Vision.* pp. 541-556 (2020)
14. Zeng, W., Liang, M., Liao, R. & Urtasun, R. Lanercnn: Distributed representations for graph-centric motion forecasting. *IEEE/RSJ International Conference On Intelligent Robots And Systems.* pp. 532-539 (2021)
15. Gao, J., Sun, C., Zhao, H., Shen, Y., Anguelov, D., Li, C. & Schmid, C. Vectornet: Encoding hd maps and agent dynamics from vectorized representation. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition.* pp. 11525-11533 (2020)
16. Gilles, T., Sabatini, S., Tsishkou, D., Stanciulescu, B. & Moutarde, F. Gohome: Graph-oriented heatmap output for future motion estimation. *International Conference On Robotics And Automation.* pp. 9107-9114 (2022)
17. Varadarajan, B., Hefny, A., Srivastava, A., Refaat, K., Nayakanti, N., Cornman, A., Chen, K., Douillard, B., Lam, C., Anguelov, D. & Others Multipath++: Efficient information fusion and trajectory aggregation for behavior prediction. *International Conference On Robotics And Automation.* pp. 7814-7821 (2022)
18. Cui, H., Radosavljevic, V., Chou, F., Lin, T., Nguyen, T., Huang, T., Schneider, J. & Djuric, N. Multimodal trajectory predictions for autonomous driving using deep convolutional networks. *International Conference On Robotics And Automation.* pp. 2090-2096 (2019)
19. Huang, Z., Mo, X. & Lv, C. Multi-modal motion prediction with transformer-based neural network for autonomous driving. *International Conference On Robotics And Automation.* pp. 2605-2611 (2022)
20. Lee, N., Choi, W., Vernaza, P., Choy, C., Torr, P. & Chandraker, M. Desire: Distant future prediction in dynamic scenes with interacting agents. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition.* pp. 336-345 (2017)
21. Fedus, W., Zoph, B. & Shazeer, N. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal Of Machine Learning Research.* **23**, pp. 1-39 (2022)

22. Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., Krikun, M., Shazeer, N. & Chen, Z. Gshard: Scaling giant models with conditional computation and automatic sharding. *ArXiv Preprint ArXiv:2006.16668*. (2020)
23. Mi, Z. & Xu, D. Switch-NeRF: Learning Scene Decomposition with Mixture of Experts for Large-scale Neural Radiance Fields. *International Conference On Learning Representations*. (2023), https://openreview.net/forum?id=PQ2zoIZqvm
24. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł. & Polosukhin, I. Attention is all you need. *Advances In Neural Information Processing Systems*. **30** (2017)
25. Zhou, Z., Ye, L., Wang, J., Wu, K. & Lu, K. Hivt: Hierarchical vector transformer for multi-agent motion prediction. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*. pp. 8823-8833 (2022)
26. Chang, M., Lambert, J., Sangkloy, P., Singh, J., Bak, S., Hartnett, A., Wang, D., Carr, P., Lucey, S., Ramanan, D. & Others Argoverse: 3d tracking and forecasting with rich maps. *Proceedings Of The IEEE/CVF Conference On Computer Vision And Pattern Recognition*. pp. 8748-8757 (2019)
27. Kingma, D. & Ba, J. Adam: A Method for Stochastic Optimization. *International Conference On Learning Representations*. (2015)
28. Sierra-Gonzalez, D., Paigwar, A., Erkent, O. & Laugier, C. MultiLane: Lane Intention Prediction and Sensible Lane-Oriented Trajectory Forecasting on Centerline Graphs. *IEEE International Conference On Intelligent Transportation Systems*. pp. 3657-3664 (2022)
29. Luo, C., Sun, L., Dabiri, D. & Yuille, A. Probabilistic multi-modal trajectory prediction with lane attention for autonomous vehicles. *IEEE/RSJ International Conference On Intelligent Robots And Systems*. pp. 2370-2376 (2020)