Information Fusion 107 (2024) 102309

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Reliable trajectory prediction in scene fusion based on spatio-temporal Structure Causal Model

Jianmin Liu [a,b], Hui Lin [a,b,*], Xiaoding Wang [a,b], Lizhao Wu [a,b], Sahil Garg [c,*], Mohammad Mehedi Hassan [d]

[a] *College of Computer and Cyber Security, Fujian Normal University, No. 8 Xuefu South Road, Fuzhou, 350117, Fujian, China*
[b] *Engineering Research Center of Cyber Security and Education Informatization, Fujian Province University, No. 8 Xuefu South Road, Fuzhou, 350117, Fujian, China*
[c] *École de technologie supérieure, Montreal, QC, H3C 1K3, Canada*
[d] *Department of Information Systems, College of Computer and Information Sciences, King Saud University, Riyadh, 11543, Saudi Arabia*

ARTICLE INFO

Keywords:
Trajectory prediction
Causal inference
Scene fusion
Intelligent transportation
Structure causal model

ABSTRACT

Existing methods for trajectory prediction predominantly employ scene fusion to enhance model performance. However, they fail to provide a rational explanation as to why the fusion of the scene context and trajectories improves model performance, which prevents them from identifying the fundamental factors limiting model performance. Hence, this paper introduces a Structured Causal Model for trajectory prediction based on causal inference, which elucidates the genuine reasons for the performance enhancement brought about by the scene context in trajectory prediction and analyzes the confounding path interference that curtails model performance. Specifically, this paper first employs the front-door criterion to eliminate the confounders during the feature extraction process, allowing the model to fairly incorporate the scene context into the spatio-temporal state. Subsequently, a spatio-temporal causal graph is generated to further extract the causal relationship of the trajectory in the current scene, serving as the spatio-temporal representation. Finally, the technique of counterfactual representation inference extrapolates the spatio-temporal features of the historical trajectory into future traffic scenes for trajectory prediction. The effectiveness and reliability of this proposed end-to-end method has been experimentally validated on two real-world datasets in real traffic scenarios, particularly in scenarios involving interactions between multiple agents.

1. Introduction

In recent years, trajectory prediction has garnered widespread attention in the domain of mobility analysis and applications. With the rapid advancement of mobile devices and sensor technology, trajectory prediction aims to predict future mobility patterns based on historical trajectory data. This technology finds extensive applications in various fields such as traffic management, intelligent navigation, autonomous driving, and logistics planning [1]. In particular, trajectory prediction plays a crucial role in the field of autonomous driving, where it is employed to predict the future motion trajectories of vehicles and other moving entities in the environment. This information is then utilized to plan the driving route of the autonomous vehicle.

Due to the complexity and uncertainty of trajectory data, trajectory prediction faces numerous challenges. Fortunately, there is a substantial body of work in the field of vehicle trajectory prediction, and with the rich scene context provided by high definition (HD) maps for trajectory prediction, significant progress has been made in this area. Previous approaches such as LaneGCN [2] and VectorNet [3], among others [4], integrate the scene context into the feature extraction process to enhance the model's predictive performance. However, they do not provide an explanation for the effectiveness of this fusion approach, and we intuitively understand that the scene context and trajectory data are related. This paper constructs a sound structural causal model from the perspective of causal inference to explain this effect, and enhances the accuracy of prediction by incorporating counterfactual reasoning.

We present the causal structure model of previous work as shown in Fig. 1(a). The introduction of the scene context in this causal model indeed achieves better performance than using only the trajectory data X_t. This model assumes that C_t and X_t are independent, but in reality, there exist some causal relationships between C_t and X_t. For instance, similar trajectory sequences in different scene contexts may display different trajectory features, as shown in Figs. 1(e) and 1(f). As we can see, the same straight-line vehicle historical tracks presents different

[*] Corresponding authors.
E-mail addresses: linhui@fjnu.edu.cn (H. Lin), sahil.garg@ieee.org (S. Garg).

https://doi.org/10.1016/j.inffus.2024.102309
Received 5 December 2023; Received in revised form 1 February 2024; Accepted 15 February 2024
Available online 28 February 2024
1566-2535/© 2024 Elsevier B.V. All rights reserved.

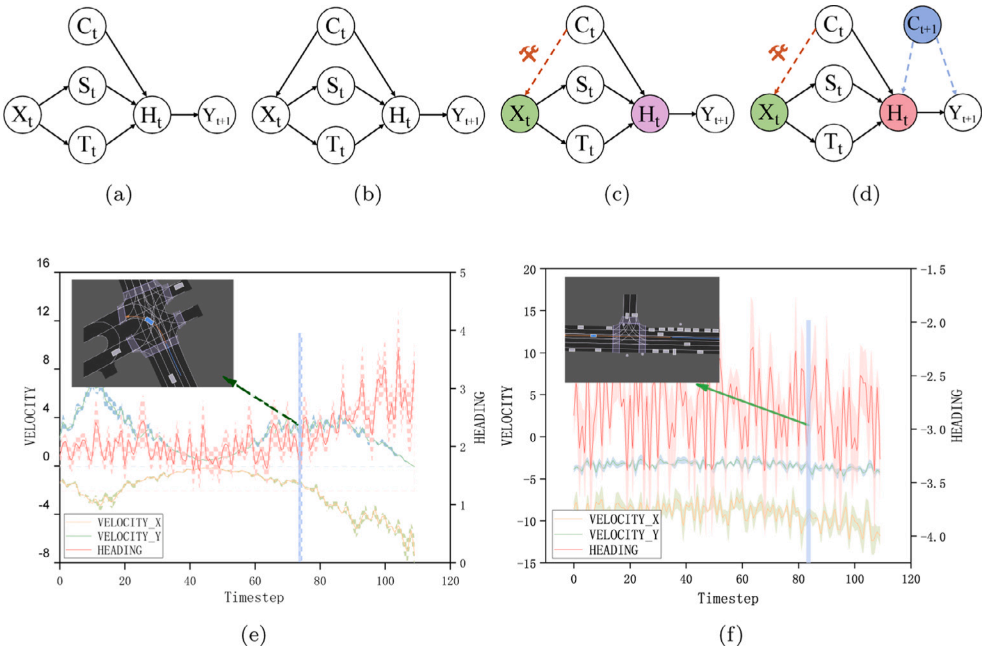


Fig. 1. Trajectory prediction from a perspective of causality. ((a) Structural causal model of previous work. (the X_t represents trajectory data, C_t denotes scene context features, S_t signifies the spatial features of the trajectory, T_t stands for the temporal features of the trajectory, and H_t is the spatio-temporal trajectory feature after fusion.) (b) Structural causal model of this work. (c) Causal intervention. (d) Counterfactual inference. (e) Features of trajectory in turning scenario. (f) Features of trajectory in straight scenario.)

trajectory features in different scene contexts. Specifically, as shown in Fig. 1(e), in turning scenario (intersections, pedestrians, traffic lights, and other uncertainty factors), the vehicle's trajectory direction, X-axis velocity, and Y-axis velocity all undergo significant changes at the moment of turning. However, in another scenario, as shown in Fig. 1(f), which differs from the above-mentioned scene context situations, the various features of the trajectory tend to be more stable.

In fact, the scene context is also a factor that influences trajectory features. The trajectory is constrained by its surrounding scene context. For instance, scene contexts such as traffic lights, intersections, and pedestrian crossings could lead to a decrease in speed, subsequently affecting the length of the trajectory.

Hence, we propose a realistic and rational causal structure model for trajectory prediction scenarios, as shown in Fig. 1(b). Due to the existence of the backdoor path $X_t \leftarrow C_t \rightarrow H_t$, there is a spurious association between X_t and H_t. This association can lead the model to learn more general features of the X_t distribution, neglecting the influence of specific scene context (such as rare slopes, accidents, surrounding vehicle behavior) C_t on trajectory features H_t. Therefore, we propose a causal intervention model, as shown in Fig. 1(c). Specifically, we utilize the front-door criterion to sever the spurious connection of the backdoor path, thereby obtaining a more accurate estimation of the causal effect. The aforementioned approach only considers the causal effect of historical scene context during the feature extraction process. To enhance the predictive capacity of the model, we employ counterfactual representation inference to incorporate future context scenes into the trajectory decoding stage, as shown in Fig. 1(d). This aids the model better understand the contextual information in the scene, which in turn improves the accuracy of trajectory prediction. Specifically, the contributions of this paper are as follows:

- This paper introduces causal inference into trajectory prediction, applying the front-door criterion based on causal intervention to eliminate confounding bias during the feature extraction process. This ensures that the model can more fairly fuse the spatio-temporal features of trajectory data with each scene context.
- This paper proposes a spatio-temporal causal graph of the trajectory in the current scene, serving as the spatio-temporal representation. This representation is then utilized to predict the future trajectory.
- We employ a counterfactual inference representation method to extrapolate the features of the factual scene's historical trajectory to future scene contexts. This enhances the understanding of future scene contexts and further improves the predictive performance of the model.
- Experiments on two real-world datasets demonstrate that the causal inference-based method exhibits excellent performance and robustness.

The remainder of the paper is organized as follows: Section 2 introduces related work on trajectory prediction and causal inference. In Section 3, we provide a task description of trajectory prediction from the perspective of causal inference. In Section 4, we detail the proposed spatio-temporal causal autoencoder model. In Section 5, we validate the method proposed in this paper through experiments, and finally, we conclude and discuss in the last Section 6.

2. Related works

2.1. Trajectory prediction

Trajectory prediction methods can be broadly divided into traditional trajectory prediction methods and data-driven trajectory prediction methods. The former utilizes probabilistic and statistical models to predict the probability of the target under different motion patterns [5,6], or directly employs kinematic models for prediction [7]. The latter used deep learning technologies to learn trajectory features and their interactive behaviors from a large amount of existing trajectory data, and then uses the learned model for prediction. From the simply using of recurrent neural networks (RNN) to model trajectories [8,9] for predicting future trajectories of vehicles on freeways. To later improvements by Deo et al. [10] who used a convolutional social pool as an upgraded version of the social pool layer on an LSTM encoder–decoder model, robustly learning the interdependencies of trajectories.

Table 1
Comparison of mainstream methods.

Literature	VM	RM	SF	TF	Interpretability	Interaction
VectorNet [3]	√	×	√	×	×	√
GANet [16]	√	×	√	×	×	√
QCNet [17]	√	×	√	√	×	√
GOHOME [18]	×	√	√	×	×	√
GoRela [28]	√	×	√	√	×	√
FRM [34]	√	×	√	√	×	√

The abbreviations mentioned above stand for Vector-maps(VM), Raster-maps(RM), Spatial features(SF) and Temporal features(TF), respectively.

Others [11] increased the order of the Markov model and neural network model to enhance the accuracy of trajectory prediction.

Early works also took into account scene context information. For example, early trajectory prediction work used classic convolutional neural networks to learn the scenes represented by multi-channel images [12–14]. However, due to the lossy nature, limited receptive field, and high cost of this rasterization method, vector-based encoding schemes were proposed [2,3,15]. Researchers proposed VectorNet [3] to introduce vectorized maps into trajectory prediction, and a large amount of work began to revolve around vector maps. Vector-based methods can efficiently aggregate sparse information in traffic scenes into trajectory features. On this basis, GANet [16] designed a fusion technique to perception future interaction in a small range near the endpoint. To avoid the redundancy of agent-centered modeling, QCNet [17] proposed a query-centered global spatio-temporal representation. After encoding history and local context, GOHOME [18] decoded and output heatmaps, and Gilles et al. [19] proposed an improved hierarchical heatmap processing to improve model performance. Other methods leveraging pooling [10,20], graph convolution [2,14,21–23], and attention mechanisms [24–27] have achieved great success. Among them, GoRela [28] proposed a method of pairwise relative position encoding, and this viewpoint-invariant method makes learning more efficient. Recently, several powerful trajectory prediction models have adopted Transformers [29] with decomposed attention as their encoders [26,30–32]. Under the powerful representation of the transformers, HPTR [33] allows the method of pairwise position representation to exhibit stronger performance, while FRM [34] pays more attention to future vehicle interactions. Although these models have improved the predictive ability of the model by integrating historical scene context into trajectory data, they do not provide a formal explanation. We compared the techniques used in mainstream methods, as shown in Table 1. In contrast, our work explains that the reason for doing so is to sever the spurious connections caused by scene context on trajectory data and spatio-temporal features. Furthermore, we use counterfactual inference to extend scene context to future trajectory prediction, further improving the predictive effect of the model.

2.2. Causal inference

Causal inference is a statistical tool that enables models to infer causal effects between variables of interest. It has been widely researched and applied in fields such as statistics, psychology, economics, sociology [35,36], computer vision [37,38], and robotics [39]. In recent years, causal inference has also been applied to the study of many computer-related problems [40,41]. We adopt the same graphical notation as proposed by Pearl in graphical models [42], and further extend it to trajectory prediction tasks. We propose a custom causal graph to explain the confounding factors affecting the model's performance in trajectory prediction, and use causal intervention and front-door adjustment to evaluate the causal effects.

3. A causal inference look at trajectory prediction

3.1. Structural causal model

We have defined the structural causal model, as shown in Fig. 1(b). In this model, X represents the trajectory data from the dataset, C represents external factors such as scene context, S represents spatial features of the trajectory, T represents temporal features of the trajectory, H represents highly coupled spatio-temporal features of the trajectory, and Y represents the model's prediction results. The subscript t denotes the current moment. We base this on the assumption that the temporal and spatial features of trajectories can be decoupled from spatio-temporal data features, and they are integrated to form a complete representation of spatio-temporal features of trajectories. Almost all current work is based on this assumption, as the extraction of temporal and spatial features always employs different feature extractors and is performed separately.

Typically, the trajectory prediction task is defined in the input of X_t, C_t, predicting a future trajectory of a vehicle, which can be represented as seeking $P(Y|X_t) = \sum_H P(Y|H)P(H|X_t)$. Here, $P(H|X_t)$ represents the process of the encoder extracting spatio-temporal causal features, and $P(Y|H)$ represents the trajectory decoding process of the decoder. Additionally, since we have introduced a counterfactual representation inference process, the hidden features of the trajectory are input into the decoder after going through the counterfactual representation inference. Therefore, the complete trajectory prediction task can be represented as Eq. (1).

$$P(Y|X_t, C_{t+1}) = \sum_H P(Y|H, C_{t+1})P(H|X_t). \tag{1}$$

3.2. Causal intervention

As can be seen from Fig. 1(c), due to the existence of the backdoor path $X_t \leftarrow C_t \rightarrow H_t$, there is a spurious correlation between the trajectory input data X_t and the spatio-temporal features H_t, which is not a causal relationship. (A backdoor path is a path that connects X_t and H_t, with one end pointing to X_t and the other pointing to H_t). This is because the scene context C is a common cause of X_t and H_t, which may lead to H_t learning more common features due to the limitations of the dataset, ignoring the differences in scene context, resulting in a biased H_t. For example, in the summer when the temperature rises, ice cream sales increase, and at the same time, there is an increase in drowning incidents. If do not consider the temperature as a variable, we can easily draw a false conclusion that there is a positive correlation between ice cream consumption and drowning incidents. This spurious correlation prevents the model from learning the true causal relationship $X_t \rightarrow (S_t, T_t) \rightarrow H_t \rightarrow Y$. However, the scene context C is a variable that cannot be exhausted, and we cannot adjust this backdoor path using backdoor adjustment because it does not meet the conditions of the backdoor criterion.

Fortunately, the existence of the path $C_t \rightarrow X_t \rightarrow (S_t, T_t) \rightarrow H_t \leftarrow C_t$ allows us to cut off the aforementioned backdoor path. Therefore, we introduce the front-door criterion to cut off this spurious connection, allowing the model to learn the features of the trajectory more fairly. Simply put, we use the causal intervention operation on variable X to make the process of the model extracting spatio-temporal features represented as $P(H_t|do(X_t))$. The specific can be described as Eq. (2).

$$\begin{aligned} P(H_t|do(X_t)) &= \sum_{S_t,T_t} P(S_t, T_t|do(X_t))P(H_t|do(S_t, T_t)) \\ &= \sum_{S_t,T_t} \sum_{X'_t} P(H_t|S_t, T_t, X'_t)P(S_t, T_t|X_t)P(X'_t) \end{aligned} \tag{2}$$

It is worth noting that even if we intervene on X, it does not affect the response formula of the variable set (S_t, T_t) to X: $P(S_t, T_t|do(X_t)) = P(S_t, T_t|X_t)$. The $P(X')$ represents the intervention on X. For this part, we use a feature enhancement network to fit the input trajectory

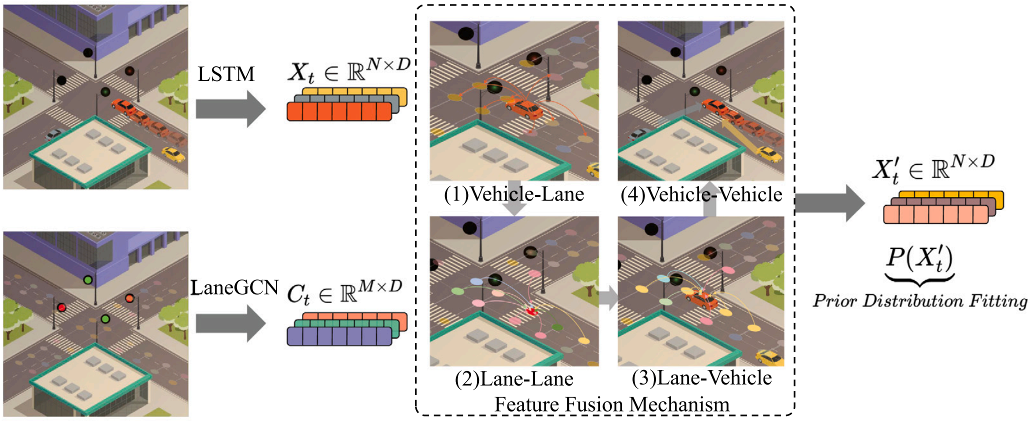

LSTM
$X_t \in \mathbb{R}^{N\times D}$
LaneGCN
$C_t \in \mathbb{R}^{M\times D}$
(1)Vehicle-Lane
(4)Vehicle-Vehicle
(2)Lane-Lane
(3)Lane-Vehicle
Feature Fusion Mechanism
$X'_t \in \mathbb{R}^{N\times D}$
$\underbrace{P(X'_t)}$
Prior Distribution Fitting

Fig. 2. Feature fusion network.

distribution in Section 4.1. $P(S_t, T_t|X_t)$ represents the process of extracting spatio-temporal features from the input noisy data. For this part, we embed the dynamic spatio-temporal causal relationship into the causal graph in Section 4.2. $P(H_t|S_t, T_t, X'_t)$ represents the generation of time-varying latent spatio-temporal state representations from features in Section 4.3, which further describe the inherent spatio-temporal patterns in the data. We propose a spatio-temporal causal graph convolution network to extract spatio-temporal state status and embed it into the GRU-cell to form a spatio-temporal structural causal unit. The encoder and decoder are composed of such spatio-temporal structural causal units.

4. Causal spatio-temporal autoencoder

4.1. Feature fusion network

To enhance the representation of trajectory features, as shown in Fig. 2, we introduce a feature fusion network that incorporates scene context features into the trajectory features.

The historical motion sequence of a self-driving vehicle can be represented as $Traj_{his} = J_T, J_{T-1}, \ldots, J_1$, where $J_T(P, H, V, ts)$ is a four-tuple describing the location and condition of the self-driving vehicle. $P = \{x, y\}$ denotes the vehicle's position. H represents the vehicle's heading at the moment. $V = \{V_X, V_Y\}$ indicates the instantaneous speed along the X,Y axes. and ts is the timestamp describing the current moment. We utilize the LSTM network to perform feature embedding on the aforementioned vehicle trajectory sequence features, as shown in Eq. (3), resulting in its feature embedding $H^J \in \mathbb{R}^{N\times D}$.

$$H^J = LSTM(Traj_{his}) \in \mathbb{R}^{N\times D} \tag{3}$$

In order to better predict the position of the trajectory in the lane, we introduce semantic data C from high-precision maps, such as lane lines and intersections, as part of generating X. Specifically, we construct four directed graphs $G(V, suc, pre, left, right)$, where v represents the center point of the lane line. the topological structure of the lane line nodes is represented by four types of connecting relationships: predecessors, successors, left neighbors, and right neighbors. Using the lane graph convolution network (LaneGCN), the node information is convoluted and aggregated. Due to the ambiguity and low order in neighboring directions, the convolution operation is carried out just once. Considering the continuity of the road on the predecessor and successor nodes, to increase the receptive area in the track direction, we introduced the concept of dilated convolution. The convolution process is carried out k times, which captures long-distance connections along the direction of the lane. Therefore, the formula for extracting the semantic info of the scene H^C can be expressed as Eq. (4):

$$\begin{aligned} H^C = CW_0 + \sum_{i\in left,right} A_i C W_i + \sum_{k=1}^{K} (A_{pre}^{k} C W_{pre,k} \\ + A_{suc}^{k} C W_{suc,k}) \in \mathbb{R}^{M\times D}, \end{aligned} \tag{4}$$

where C is the feature of lane node. A_{pre} is the adjacency matrix representing the pre type relationship. A_i^k is the matrix raised to the power of k representing the i-type. W denotes the parameter that can be trained. The variable k indicates the order of dilated convolution. Note that expansion only happens on predecessor and successor nodes.

For each moment's trajectory feature of vehicle i representation h_i^J, we consider the node j of the lane within a certain range, and use the attention fusion method to integrate the scene context h_j^C into the vehicle's trajectory features as X'_i. To capture the rich interaction between vehicles and lanes, we employed feature fusion by concatenating trajectory features, lane features, and the Euclidean distance between them. This allows for the integration of information from both trajectories and lanes. The specific fusion method can be expressed as formula (5).

$$X'_i = h_i^J W_0 + \sum_j \sigma(concat(h_i^J, \Delta_{i,j}, h_j^C) W_1) W_2 \in \mathbb{R}^{N\times D}, \tag{5}$$

where W_0, W_1, W_2 are trainable parameters, $\Delta_{i,j}$ is the Euclidean distance difference between nodes i and j, and σ is the operation of the non-linear activation function Relu. This fusion method of vehicle features h^J and scene context features h^C is denoted as $h^J \textcircled{\bowtie} h^C$.

4.2. Spatio-temporal causal interaction graph convolution

The dynamic causal relationship of the trajectory is complex. Existing methods focus on the spatial dependency relationship of nodes, but pay insufficient attention to the temporal causal effect of nodes. To better utilize the temporal features of the trajectory, we concatenate the hidden feature $H_{t-1} \in \mathbb{R}^{N\times D}$ from the previous moment with the feature $X'_t \in \mathbb{R}^{N\times D}$ inputted at each moment, resulting in a new feature representation that contains temporal causality. this feature is then transformed into a spatio-temporal causality graph to represent the spatio-temporal causal effects between nodes. Specifically, after this concated feature is inputted, it is transformed to $I_t \in \mathbb{R}^{N\times D}$ through a fully connected layer, as shown in Eq. (6).

$$I_t = FC(X'_t \parallel H_{t-1}) \in \mathbb{R}^{N\times D} \tag{6}$$

We perform global average pooling on all features of each node, represented as a scalar, which indicates the importance of this node.

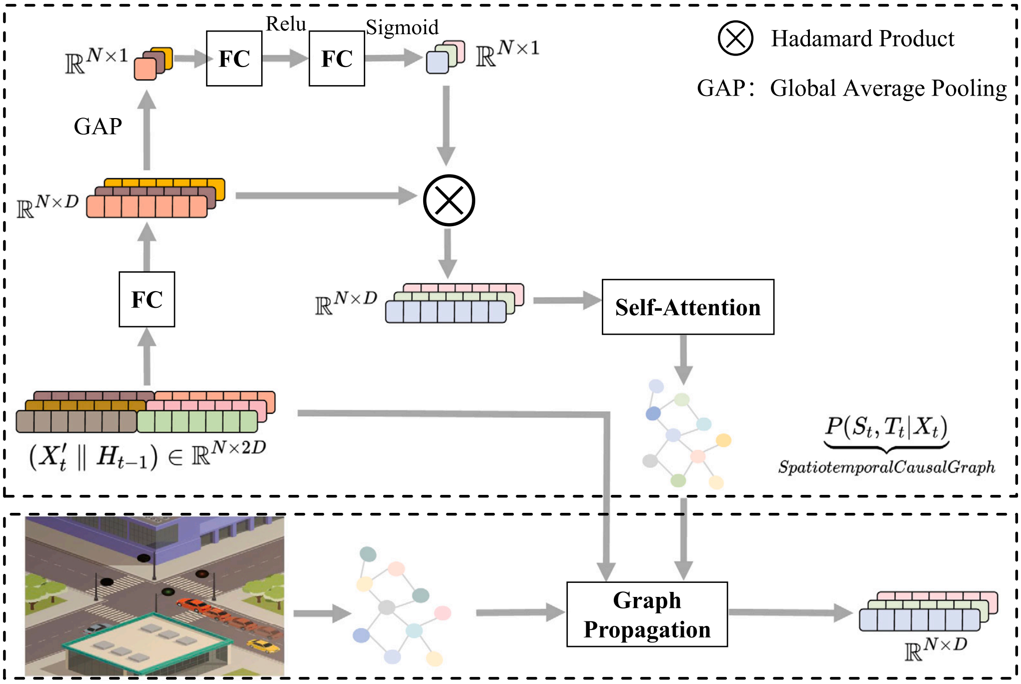


Fig. 3. Spatio-temporal causal graph convolutional network.

After performing this operation on all nodes, we obtain the importance index sequence of this group of nodes. In this case, we employed the SE(squeeze-and-excitation) network which involves a fully connected layer and ReLU activation function for feature squeeze. Following this, a fully connected layer and sigmoid function are applied to obtain gate factors α for excitation, with each value ranging between 0 and 1. The calculation process is as follows:

$$\alpha = \sigma(FC(ReLU(FC(\frac{1}{d}\sum_{c=1}^{D} I_t[:,c])))) \in \mathbb{R}^{N} \tag{7}$$

The gating factor α and the feature I_t are subjected to a Hadamard element product to obtain the dynamic spatio-temporal causality representation DST_t of this node at time t. The specific can be described as formula (8).

$$DST_t = I_t \odot \alpha \in \mathbb{R}^{N\times D} \tag{8}$$

Finally, the dynamic causal relationship is embedded into the dynamic spatio-temporal causality graph through the self-attention mechanism by Eq. (9).

$$A_t^{dst} = ReLU(\phi(\frac{DST_t DST_t^T}{\sqrt{d}})) \in \mathbb{R}^{N\times N} \tag{9}$$

where ϕ is the tanh function. The above part of Fig. 3 illustrates the aforementioned process.

The interaction relationship of a vehicle in the environment can be represented by establishing a spatio-temporal interaction graph $G^{int}(V,E,A)$, where V is a collection of vehicles, pedestrians, etc., each agent is considered as a node. Edges E represent the presence of interaction between different agents. The adjacency matrix A is obtained using formula (10). The Euclidean distance between intelligent agents at the current moment is calculated based on their coordinate locations. If it is less than a $threshold$, it is deemed that there is an interaction relationship between agents. In practice, we set this threshold to 100. For each subgraph, we believe that there is an interaction relationship between all vehicles, hence a bidirectional fully connected graph is constructed.

$$A_{i,j}^{int} = \begin{cases} \frac{\exp(dis_{i,j})}{\sum_{k=1}^{N}\exp(dis_{i,k})}, & dis_{i,j} \leqslant threshold \\ 0, & otherwise \end{cases} \tag{10}$$

From this, we obtain the spatio-temporal causal feature graph of the trajectory A^{DST} and the spatio-temporal interaction graph of the trajectory A^{int}. In order to enable the model to effectively integrate causal features and interaction features, we propose a new spatio-temporal causal graph convolution network (STCGCN).

Specifically, at time t, the input feature $X_t^{(0)}$ can be represented as Eq. (11):

$$X_t^{(0)} = (X' \parallel H_{t-1}) \in \mathbb{R}^{N\times 2D}, \tag{11}$$

where X' is the output of the feature enhancement network, and H_{t-1} is the hidden feature passed from the previous moment to the current moment. Therefore, the spatio-temporal causal interaction graph convolution network is defined as Eq. (12):

$$X_t^{(l)} = \theta_0 X_t^{(l-1)} + \theta_1 A^{dst} X_t^{(l-1)} + \theta_2 A^{int} X_t^{(l-1)} \in \mathbb{R}^{N\times 2D}, \tag{12}$$

where l represents the layer. $\theta_0, \theta_1, \theta_2$ are three weight coefficients that are learnable parameters. Eq. (12) corresponds to the lower half of Fig. 3.

In fact, this step corresponds to $P(S_t,T_t|X_t)\sum_{X'_t} P(H_t|S_t,T_t,X'_t)$ $P(X'_t)$ in Eq. (1). To obtain all potential time-varying spatial features, i.e., $\sum_{S_t,T_t}\sum_{X'_t} P(H_t|S_t,T_t,X'_t)P(S_t,T_t|X_t)P(X'_t)$. Finally, we train a linear network to further extract features from the STCGCN output and restore them to their original feature dimensions by Eq. (13).

$$\tilde{X}_t^{out} = ReLU(\sum_{k=0}^{l} X_t^{(k)} W^{(k)} + b^{(k)}) \in \mathbb{R}^{N\times D}. \tag{13}$$

The above steps implement the process of extracting spatio-temporal causal feature effects from trajectory data under the intervention of X in formula (14). We denote the above process as

$$\tilde{X}_t^{out} = \theta \star_G (X'_t \parallel H_{t-1}). \tag{14}$$

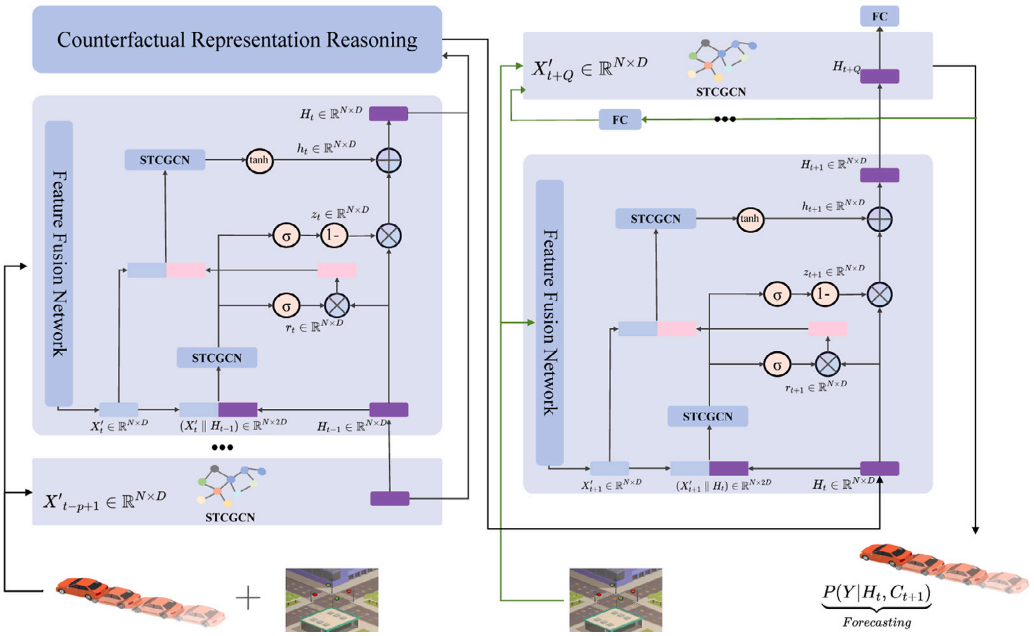

Counterfactual Representation Reasoning
Feature Fusion Network
STCGCN
tanh
$H_t \in \mathbb{R}^{N\times D}$
$h_t \in \mathbb{R}^{N\times D}$
$z_t \in \mathbb{R}^{N\times D}$
σ
1-
$r_t \in \mathbb{R}^{N\times D}$
$X'_t \in \mathbb{R}^{N\times D}$
$(X'_t \parallel H_{t-1}) \in \mathbb{R}^{N\times 2D}$
$H_{t-1} \in \mathbb{R}^{N\times D}$
$X'_{t-p+1} \in \mathbb{R}^{N\times D}$
FC
$X'_{t+Q} \in \mathbb{R}^{N\times D}$
H_{t+Q}
$H_{t+1} \in \mathbb{R}^{N\times D}$
$h_{t+1} \in \mathbb{R}^{N\times D}$
$z_{t+1} \in \mathbb{R}^{N\times D}$
$r_{t+1} \in \mathbb{R}^{N\times D}$
$X'_{t+1} \in \mathbb{R}^{N\times D}$
$(X'_{t+1} \parallel H_t) \in \mathbb{R}^{N\times 2D}$
$H_t \in \mathbb{R}^{N\times D}$
$P(Y|H_t, C_{t+1})$
Forecasting

Fig. 4. Spatio-temporal structural causal unit.

4.3. Spatio-temporal structural causal unit

To more clearly describe the operation logic of our model, we abstract the model as a nested GRU network structure, which we call a causal spatio-temporal autoencoder, as depicted in Fig. 4. We replaced the ordinary convolution with spatio-temporal causal graph convolutions to capture causal relationships in the spatio-temporal domain. Each cell unit of the encoder describes the complete causal intervention effect $P(H_t|do(X_t))$, where the feature enhancement network describes $P(X')$, and the spatio-temporal causal graph convolution describes the $\sum_{S_t,T_t}\sum_{X'_t} P(H_t|S_t,T_t,X'_t)P(S_t,T_t|X_t)P(X'_t)$ process.

Specifically, the calculation process of the spatio-temporal causal unit (SCU) in the Causal Spatio-temporal Autoencoder is as Eq. (15):

$$\begin{aligned}
r_t &= \sigma(\theta_r \star_G (X'_t \parallel H_{t-1}) + b_r)\\
z_t &= \sigma(\theta_z \star_G (X'_t \parallel H_{t-1}) + b_z)\\
\tilde{h}_t &= \sigma(\theta_h \star_G (X'_t \parallel r_t \odot H_{t-1}) + b_h)\\
H_t &= z_t \odot H_{t-1} + (1 - z_t) \odot \tilde{h}_t
\end{aligned} \tag{15}$$

where $\odot$ is the hadamard product, i.e., element-wise multiplication. θ_r, θ_z, θ_h, b_r, b_z, b_h are all learnable parameters, $\star_G$ is the STCGCN convolution process defined in the above formula. H_t is the spatio-temporal state of the structural causal unit (SCU) at time t. The core framework of the spatiotemporal causal convolutional unit is similar to that of a GRU network. Therefore, it also consists of three gate controls: r_t, z_t, and h_t. Among them, h_t is the candidate for the current state, r_t is the reset gate, and the element-wise product of h_t and H_{t-1} yields the history state that has not been reset. z_t represents how much of the historical state can be retained for the next time step.

Multiple structural causal units are stacked to form an encoder–decoder structure. The main structure of both the encoder and decoder is the structural causal units, so we do not provide a detailed description of the components of the decoder. The scene context and the historical trajectory sequence are input into the encoder. The spatio-temporal causal features extracted in the encoder are sent to the counterfactual inference module to focus more on the trajectory features that are more similar to the future scene, thereby further enhancing the model's understanding of the future scene context. Finally, the output of the counterfactual inference module is sent to the decoder, and the future scene context is also sent to the decoder.

4.4. Counterfactual representation inference

In counterfactual reasoning, we assume a counterfactual scenario where a causal relationship occurs, meaning the value of a dependent variable has changed while keeping the values of other variables constant. Then, by comparing the data from the counterfactual scenario with the actual scenario, we can infer the possible outcomes of other variables.

The trajectory data X_t is generated under the scene C_t, and we can extract the spatio-temporal causal features H_t under this factual scene. However, the current H_t mostly contains the spatio-temporal features under the factual scene C_t, and the spatio-temporal features may not be the same for the future scene context C_{t+1}. Therefore, we need to use the trajectory data X_t under the factual scene to calculate the future trajectory under the counterfactual scene C_{t+1}. Formally, we can obtain by Eq. (16):

$$\begin{aligned}
P(Y_{C_{t+1}}|X_t) &= \sum_{H'_t} P(Y_{C_{t+1}}|do(H'_t), do(C_{t+1}), X_t)P(H'_t|X_t, do(C_{t+1}))\\
&= \sum_{H'_t} P(H'_t|X_t, C_{t+1})P(Y_{C_{t+1}}|H'_t, C_{t+1})
\end{aligned} \tag{16}$$

where $do(C_{t+1})$ indicates that this process is an artificially set counterfactual scenario. $P(H'_t|X_t, C_{t+1})$ represents the counterfactual inference process, and $P(Y_{C_{t+1}}|H'_t, C_{t+1})$ represents the prediction process based on counterfactual representation.

In the counterfactual inference process, our goal is to make the model pay more attention to the parts that are similar to the historical scene under the condition of future scene context, which would help the model understand the spatio-temporal state under C_{t+1}. To represent the degree of attention to this scene context, we introduce scaled dot-product attention to compute the attention of historical and future scene contexts, and use this attention to transform the historical trajectory spatio-temporal features into the input of the decoder. The implementation details are illustrated in Fig. 5.

Specifically, we use the self-attention mechanism to calculate the similarity of the scene context on each road node, using the global scene context C_g corresponding to the current trajectory. In the counterfactual inference process, we use this attention matrix to transform the output of the encoder into the input of the decoder. The formula (17)

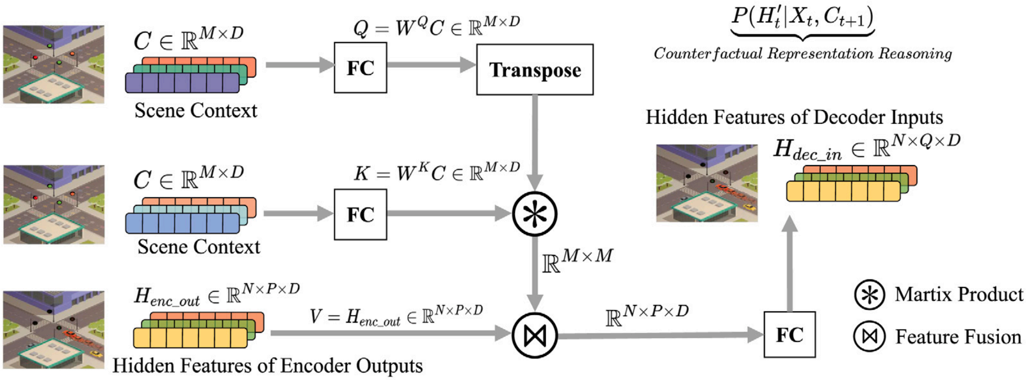

$C \in \mathbb{R}^{M\times D}$
Scene Context
$Q = W^Q C \in \mathbb{R}^{M\times D}$
FC
Transpose
$\underbrace{P(H_t'|X_t, C_{t+1})}$
$Counterfactual\ Representation\ Reasoning$
Hidden Features of Decoder Inputs
$H_{dec\_in} \in \mathbb{R}^{N\times Q\times D}$
$C \in \mathbb{R}^{M\times D}$
Scene Context
$K = W^K C \in \mathbb{R}^{M\times D}$
FC
$\mathbb{R}^{M\times M}$
$H_{enc\_out} \in \mathbb{R}^{N\times P\times D}$
$V = H_{enc\_out} \in \mathbb{R}^{N\times P\times D}$
$\mathbb{R}^{N\times P\times D}$
FC
Hidden Features of Encoder Outputs
Martix Product
Feature Fusion

Fig. 5. Counterfactual representation inference.

for the counterfactual representation inference module is:

$$
\begin{aligned}
Q &= C_g W^Q \in \mathbb{R}^{M\times D} \\
K &= C_g W^K \in \mathbb{R}^{M\times D} \\
V &= H^{his} W^V \in \mathbb{R}^{M\times D}
\end{aligned}
\tag{17}
$$

where the dimension of the features of D, W^Q, W^K, and W^V are three sets of trainable parameters. From this, we obtain the decoder's input $H_{pred} = softmax(\frac{QK^T}{\sqrt{D}}) \bowtie V$. $\bowtie$ is the attention aggregation method mentioned as Eq. (5).

At this point, we have obtained the representation transformation of the trajectory features on the scene context nodes, which is handed over to the decoder to output the prediction results.

5. Experimental results and analysis

5.1. Experimental setup

5.1.1. Datasets

Argoverse is a large-scale multimodal sensor dataset provided by Argo AI, aiming to drive autonomous driving and machine learning research. The trajectory prediction dataset focuses on predicting the future movements of traffic participants in autonomous driving scenarios.

Argoverse 1.1 motion forecasting [43]: This version of the dataset contains about 324,557 trajectories during over 1000 h in Miami and Pittsburgh. Each trajectory has a 2-s history and a 3-s future. The scenes in the dataset cover a variety of traffic environments, such as city streets and highways. Each scene contains trajectories of vehicles, pedestrians, and other traffic participants. Each trajectory includes information such as time, location, and speed. In addition, the dataset also includes map data, such as lane and traffic signal information.

Argoverse 2 motion forecasting [44]: This version of the dataset is an expansion and improvement on the 1.1 version. This version contains about 250,000 scenes in six different cities, each scene lasts up to 11 s, greatly increasing the quantity and diversity of the data. Moreover, each scene has a dedicated map, and this version also adds more tags and attributes for more in-depth analysis and research.

The main differences between the two versions lie in the amount of data and the richness of labels. Argoverse 2 has far more data than version 1.1, providing more diverse scenes and interactive behaviors, which can support more complex prediction models and algorithms. In addition, Argoverse 2 also provides more labels and attributes, which can help researchers better understand the behavior and interaction relationships of traffic participants.

5.1.2. Evaluation metrics

The evaluation of the experimental results in this paper uses basic prediction parameters and evaluation metrics, considering the following two situations: unimodal result (k = 1) and multi-modal result (k = 6). In the process of reasoning, this study uses historical data of 2 s and 5 s to predict the future motion trajectories of 3 s and 6 s for the datasets argoverse 1 and argoverse 2, respectively. Common evaluation metrics consist of miss rate (MR_k), minimum average displacement error ($minADE_k$), and minimum final displacement error ($minFDE_k$).

5.1.3. Implementation details

Our model undergoes end-to-end training using the AdamW optimizer set at a learning rate of 0.0001, and a batch size of 4 scenes. The training spans 70 epochs and is conducted on an NVDIA RTX 3090. For our proposed causal spatio-temporal encoder–decoder, we stack different numbers of encoder and decoder layers, depending on the length of the input sequence and the output sequence. Specifically, we define the sequence length processed by each encoder–decoder unit as 1 s. For the Argoverse 1.1 dataset, we stack 2 layers of encoders and 3 layers of decoders. For the Argoverse 2 dataset, the length of the encoder is set to 5, and the number of stacked decoder layers is 6. During the training process, all hidden feature dimensions are set to 128. When capturing the interaction range of vehicles, we define the interaction ranges of vehicle-vehicle, vehicle-lane, and lane-lane as 100 m, 10 m, and 10 m respectively. This is a parameter that is adjusted manually and through experience. Finally, we output multi-modal prediction trajectories through parallel multiple fully connected layers.

5.2. Benchmark results

Table 2 shows the experimental multi-modal results of our model on Argoverse1. The trajectory prediction method based on the perspective of causal inference achieved competitive results. Due to the differences in map scenes and duration between Argoverse1 and Argoverse2, there will be differences in the evaluation metrics on the two datasets, but this is fair for all methods. Our method outperforms most of the methods, which indicates that the method based on causal inference can effectively remove confounding factors that affect the accuracy of the model and improve the accuracy of model prediction. Compared to the Argoverse2 dataset, the Argoverse1 dataset has relatively fewer scene contexts and contains a significant amount of redundant information. However, it can still provide substantial assistance during inference.

We compared our method with methods on the Argoverse1 and Argoverse 2 motion prediction datasets. Since Argoverse 2 has more complex scenes and a longer prediction range, it can better test the model's performance in long-term trajectory prediction. We conducted experiments on Argoverse 2, the results are shown in Table 3. From

Table 2
Argoverse1 test.

Method	$b-minFDE_6$ ↓	$minFDE_6$	$minADE_6$	MR_6
Wayformer [31]	1.7408	1.1616	0.7676	0.1186
GANet [16]	1.7899	1.1605	0.8060	0.1179
Ours	**1.7917**	**1.1975**	**0.8037**	**0.1068**
multipath++[45]	1.7932	1.2144	0.7897	0.1324
TPCN [15]	1.7963	1.1675	0.7797	0.1163
HiVT [32]	1.8171	1.1460	0.7673	0.1221
DenseTNT [46]	1.9759	1.2815	0.8817	0.1258
GOHOME [18]	1.9834	1.4503	0.9425	0.1048
mmTransformer [25]	2.0328	1.3383	0.8436	0.1540
LaneRCNN [22]	2.1470	1.4526	0.9038	0.1232

the results, we observed that using module integration and predicting multi-modal trajectories can increase the diversity of predictions, thereby improving the performance of the brier-minFDE and miss rate metrics. Although our model did not achieve SOTA performance, its performance on the validation and test sets is already competitive. The performance metric of our model, $minFDE_6$, has reached a level comparable with GoRela [28], HPTR [33], and SIMPL [47]. This reflects the effectiveness and rationality of implementing causal inference in trajectory prediction tasks.

5.3. Ablation experiment

To more effectively demonstrate the effectiveness of various modules in the model, we separately remove different model components and structures, train with 30% of the training set, and test the performance of the model on the validation set as shown in the Table 4. In this table, F represents the feature fusion network, D_G represents the dynamic causal graph, and C represents the counterfactual reasoning component.

Importance of Feature Enhancement Network: In the model, M1 removes the Feature Fusion component of the feature enhancement network, which means that the agent features are directly fed into the autoencoder. Due to the lack of feature enhancement and relying solely on trajectory features without fusing scene context features, the model performance is severely impaired, with the BrierMinFDE metric being 29% higher, which is quite substantial. This further confirms the importance of scene context for trajectory prediction.

Importance of Dynamic Spatio-Temporal Causal Graph: M2 eliminates the process of constructing a dynamic spatio-temporal causal graph, and graph convolution is only performed on the dynamic interaction graph. The dynamic interaction graph only considers the positional relationship between vehicles at the current moment and does not utilize the temporal causality of the agent from the past to the present. The loss of this part of the causal relationship resulted in a 15% increase in the BrierMinFDE metric of the model performance.

Importance of Counterfactual Module: M3 removes the counterfactual inference module of the model, and the embedded hidden spatio-temporal features are directly input into the decoder for trajectory prediction. Compared to the first two modules, the counterfactual module has the least impact on the model, but the BrierMinFDE metric is still 9% higher. This is because the model still considers the future factual scene context in the decoding stage to improve the quality of trajectory decoding. Finally, the result with the complete component modules shows that the model's performance is at its best.

5.4. Qualitative results

We randomly selected a portion of scenes from the Argoverse 2 dataset to visualize the predictive performance of our model, as shown in Fig. 6. In the figure, the pale blue lines represent the agent's historical motion, the crimson lines indicate the ground truth, and the dark teal lines represent the multi-modal predictions.

From Fig. 6(a), we can observe that the prediction scheme based on causal inference and counterfactuals provides more reasonable choices, such as turning right or executing, rather than going against the flow. In Fig. 6(b), the trajectory display also provides more reasonable and complete possibilities. In Fig. 6(c), we show a scenario where the model might fail. Thanks to the application of the counterfactual inference module, the model can effectively avoid abnormal trajectories, even though it still does not achieve the optimal result. However, the predicted direction is still reasonable, which is encouraging to us. The failure cases can be addressed by using the target area scene context information to enhance the diversity of model predictions. This performance can be further improved in the future.

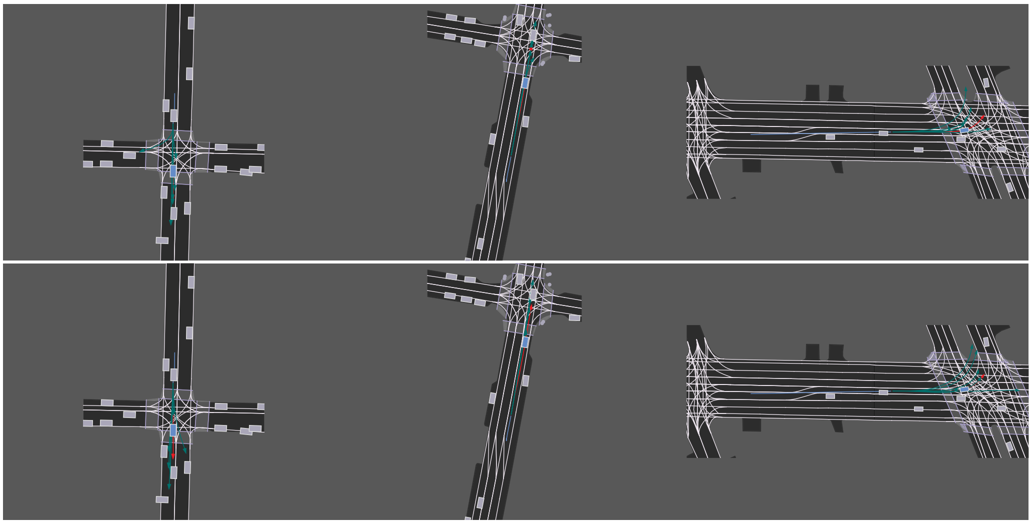
In addition, we performed trajectory prediction in open-world scenarios to evaluate the model's performance in interacting with other agents in multi-agent trajectory prediction. As shown in Fig. 7, it is worth noting that we not only show the vehicle trajectories of the focus agents, but also include the motion trajectories of other autonomous vehicles, represented by orange lines in the figure. Specifically, we plotted the motion trajectories of three different types of traffic participants: vehicles, pedestrians, and cyclist or motorcyclist. For each type of participant, we illustrated two qualitative results: one success case (top) and the other indicating shortcomings(bottom). In Fig. 7(a), we can observe that the model can accurately predict the turning movements of vehicles in unique scenarios. In another scenario below in Fig. 7(a), the model considers the interaction between vehicles. Despite most trajectories being relatively normal, there is still one with a substantial error. In Fig. 7(b), which predicts the trajectories of cyclist or motorcyclist, trajectories that comply with the scenario and social traffic rules can be accurately predicted. However, incorrect predictions exist for behaviors such as crossing the road. This poses a challenge for the model. Similarly, in Fig. 7(c), predicting the movement trajectories of pedestrians, we also exhibit two situations, success and failure. Although the failed cases highlight the limitations of our model, it is encouraging that it is still reasonable. These situations can be further optimized by increasing the diversity of predictions and evaluating their probabilities.

In summary, we can observe that the vehicle trajectories predicted by the model can take into account the interactions with other vehicles, such as reducing speed to avoid collision resulting in shorter trajectory length, and so on. What we aim to achieve is that in complex scenarios, the model can effectively explore the interactions in the scene to predict trajectories. From the visualized results, our model has demonstrated a good understanding of the scene context and dynamic interactions between vehicles, and provides multi-modal prediction results.

5.5. Efficiency analysis

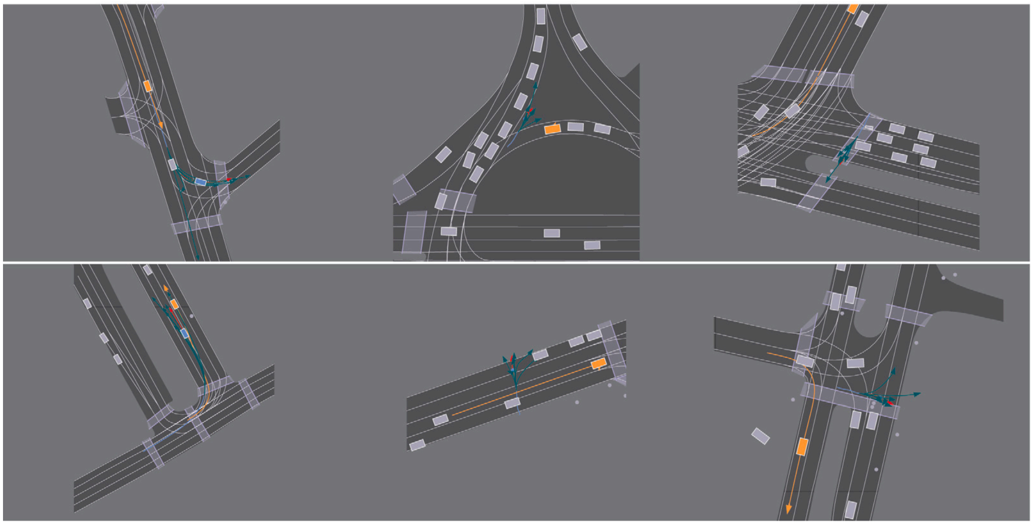
Most models are not open source, but HPTR [33] is. CrossAtt [4] is our previous work, which uses a cross-attention mechanism to enhance the model's perception of interactive behaviors and further strengthens scene-context information through a gating mechanism. Additionally, we manually implemented the baseline of Wayformer [31] for comparison. Thus, we compared the efficiency of our approach with the work mentioned above. The Fig. 8 shows the performance comparison of these models.

In the left of Fig. 8, we compared the performance of different models when handling varying numbers of intelligent agents. Wayformer, based on the Transformer architecture, requires substantial computational resources during inference. On an NVIDIA RTX 3090, it can handle a maximum of 48 agents before encountering a memory overflow error. Our two methods and HPTR performed similarly, but the approach proposed in this paper used fewer parameters and occupied the least GPU memory during training. In the right of Fig. 8, we demonstrated the BrierFDE performance of the four models when trained with different sample sizes. Since we did not use the official Wayformer code, its performance was mediocre. We found that our



(a) (b) (c)

Fig. 6. Qualitative results comparison. (The top results are derived from causal inference and counterfactuals. The below are achieved by eliminating the spatio-temporal causal diagram.)



(a) Vehicle (b) Cyclist or Motorcyclist (c) Pedestrian

Fig. 7. Qualitative results of different agents. For each type of agent, we illustrated two qualitative results: one success case (top) and the other indicating shortcomings (bottom).

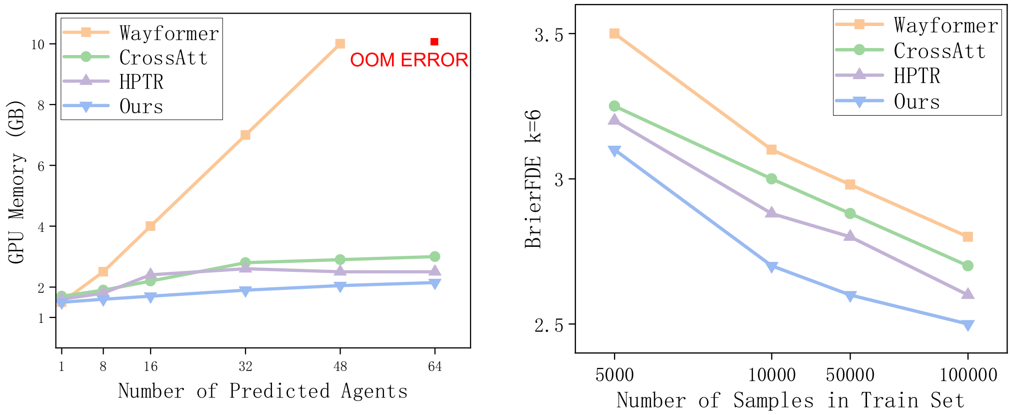


Fig. 8. Efficiency comparison. (Experiments were conducted on an Ubuntu host equipped with an NVIDIA RTX 3090 and Pytorch 2.0 to compare the GPU memory usage of different models and their conver'gence speed under varying sample conditions.)

Table 3
Argoverse2 test.

Method	$b-minFDE_6 \downarrow$	$minFDE_6$	$minFDE_1$	$minADE_6$	$minADE_1$	MR_6	MR_1
GANet [16]	1.96	1.34	4.48	0.72	1.77	0.17	0.59
GoRela [28]	2.01	1.48	4.62	0.76	1.82	0.22	0.61
HPTR [33]	2.03	1.43	4.61	0.73	1.84	0.19	0.61
SIMPL [47]	2.05	1.43	5.50	0.72	2.03	0.19	0.65
Ours	2.07	1.46	4.81	0.74	1.91	0.19	0.67
THOMAS [19]	2.16	1.51	4.71	0.88	1.95	0.20	0.64
FRM [34]	2.47	1.81	5.93	0.89	2.37	0.29	0.71
CrosAtt [4]	3.58	2.72	12.44	1.18	4.58	0.41	0.92
CratPred [48]	3.68	2.82	13.30	1.21	4.93	0.42	0.93

Table 4
Ablation study on Argoverse2 (Validation set) training on 30% of training dataset.

Model	F	D_G	C	$b-minFDE_6$	$minFDE_6$	$minFDE_1$
M_1	×	✓	✓	3.06	1.86	4.02
M_2	✓	×	✓	2.73	1.61	3.23
M_3	✓	✓	×	2.59	1.35	2.98
Ours	✓	✓	✓	2.37	1.14	2.69

method converged faster than the others, indicating that it has a faster inference speed under the same conditions. We did not use all training samples for testing due to the significant time cost of training. Both experiments demonstrated that the method proposed in this paper has lower memory consumption and faster inference capability, indicating its advancement.

6. Conclusion

In this paper, we re-examine the task of trajectory prediction from the perspective of causal inference, and provide a reasonable causal graph for explanation. Through the methods of causal intervention and counterfactual representation inference, we demonstrate efficient information fusion of scene context and trajectory data. Specifically, using the front-door criterion, we decompose the process of spatio-temporal feature extraction into several sub-problems and propose specific modules to solve these sub-problems individually. Importantly, we embed the dynamic spatio-temporal causal effects in trajectories into the causal graph to enhance the representation ability of the causal autoencoder, and then use counterfactual representation inference to improve the inferential performance of the model. Through experiments and analysis on two real-world datasets, we demonstrate that the suggested approach attains competitive performance.

CRediT authorship contribution statement

Jianmin Liu: Writing – original draft, Formal analysis, Conceptualization. **Hui Lin:** Writing – original draft, Validation, Methodology. **Xiaoding Wang:** Validation, Investigation, Formal analysis. **Lizhao Wu:** Validation, Supervision, Resources. **Sahil Garg:** Writing – review & editing, Project administration. **Mohammad Mehedi Hassan:** Writing – review & editing, Supervision, Resources, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The authors are grateful to King Saud University, Riyadh, Saudi Arabia for funding this work through Researchers Supporting Project Number (RSP2024R18).

References

[1] G. Li, Z. Zhao, X. Guo, L. Tang, H. Zhang, J. Wang, Towards integrated and fine-grained traffic forecasting: A spatio-temporal heterogeneous graph transformer approach, Inf. Fusion 102 (2024) 102063.

[2] M. Liang, B. Yang, R. Hu, Y. Chen, R. Liao, S. Feng, R. Urtasun, Learning lane graph representations for motion forecasting, in: A. Vedaldi, H. Bischof, T. Brox, J.-M. Frahm (Eds.), Computer Vision – ECCV 2020, in: Lecture Notes in Computer Science, Springer International Publishing, Cham, 2020, pp. 541–556.

[3] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, C. Schmid, Vectornet: Encoding hd maps and agent dynamics from vectorized representation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11525–11533.

[4] X. Wang, J. Liu, H. Lin, S. Garg, M. Alrashoud, A multi-modal spatial– temporal model for accurate motion forecasting with visual fusion, Inf. Fusion 102 (2024) 102046.

[5] G. Xie, H. Gao, B. Huang, L. Qian, J. Wang, A driving behavior awareness model based on a dynamic bayesian network and distributed genetic algorithm, Int. J. Comput. Intell. Syst. 11 (1) (2018) 469–482.

[6] M. Hu, Y. Liao, W. Wang, G. Li, B. Cheng, F. Chen, et al., Decision tree-based maneuver prediction for driver rear-end risk-avoidance behaviors in cut-in scenarios, J. Adv. Transp. 2017 (2017).

[7] G. Xie, H. Gao, L. Qian, B. Huang, K. Li, J. Wang, Vehicle trajectory prediction by integrating physics-and maneuver-based approaches using interactive multiple models, IEEE Trans. Ind. Electron. 65 (7) (2017) 5999–6008.

[8] K. Min, D. Kim, J. Park, K. Huh, Rnn-based path prediction of obstacle vehicles with deep ensemble, IEEE Trans. Veh. Technol. 68 (10) (2019) 10252–10256.

[9] F. Altché, A. de La Fortelle, An lstm network for highway trajectory prediction, in: 2017 IEEE 20th International Conference on Intelligent Transportation Systems, ITSC, IEEE, 2017, pp. 353–359.

[10] N. Deo, M.M. Trivedi, Convolutional social pooling for vehicle trajectory prediction, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 1468–1476.

[11] P. Pecher, M. Hunter, R. Fujimoto, Data-driven vehicle trajectory prediction, in: Proceedings of the 2016 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation, 2016, pp. 13–22.

[12] S. Casas, W. Luo, R. Urtasun, Intentnet: Learning to predict intention from raw sensor data, in: Conference on Robot Learning, in: PMLR, 2018, pp. 947–956.

[13] H. Cui, V. Radosavljevic, F.-C. Chou, T.-H. Lin, T. Nguyen, T.-K. Huang, J. Schneider, N. Djuric, Multimodal trajectory predictions for autonomous driving using deep convolutional networks, in: 2019 International Conference on Robotics and Automation, ICRA, IEEE, 2019, pp. 2090–2096.

[14] F. Da, Y. Zhang, Path-aware graph attention for hd maps in motion prediction, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 6430–6436.

[15] M. Ye, T. Cao, Q. Chen, Tpcn: Temporal point cloud networks for motion forecasting, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 11318–11327.

[16] M. Wang, X. Zhu, C. Yu, W. Li, Y. Ma, R. Jin, X. Ren, D. Ren, M. Wang, W. Yang, GANet: Goal area network for motion forecasting, 2023, http://dx.doi.org/10.48550/arXiv.2209.09723, arXiv:2209.09723.

[17] Z. Zhou, J. Wang, Y. Li, Y. Huang, Query-centric trajectory prediction, in: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, IEEE, Vancouver, BC, Canada, 2023, pp. 17863–17873.

[18] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, F. Moutarde, Gohome: Graph-oriented heatmap output for future motion estimation, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 9107–9114.

[19] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, F. Moutarde, THOMAS: Trajectory heatmap output with learned multi-agent sampling, 2021, https://arxiv.org/abs/2110.06607v3.

[20] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, S. Savarese, Social lstm: Human trajectory prediction in crowded spaces, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 961–971.

[21] A. Mohamed, K. Qian, M. Elhoseiny, C. Claudel, Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 14424–14432.

[22] W. Zeng, M. Liang, R. Liao, R. Urtasun, Lanercnn: Distributed representations for graph-centric motion forecasting, in: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, IEEE, 2021, pp. 532–539.

[23] Y. Liu, S. Rasouli, M. Wong, T. Feng, T. Huang, Rt-gcn: Gaussian-based spatiotemporal graph convolutional network for robust traffic prediction, Inf. Fusion 102 (2024) 102078.

[24] X. Jia, P. Wu, L. Chen, Y. Liu, H. Li, J. Yan, Hdgt: Heterogeneous driving graph transformer for multi-agent trajectory prediction via scene encoding, IEEE Trans. Pattern Anal. Mach. Intell. (2023).

[25] Y. Liu, J. Zhang, L. Fang, Q. Jiang, B. Zhou, Multimodal motion prediction with stacked transformers, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 7577–7586.

[26] J. Ngiam, B. Caine, V. Vasudevan, Z. Zhang, H.-T.L. Chiang, J. Ling, R. Roelofs, A. Bewley, C. Liu, A. Venugopal, D. Weiss, B. Sapp, Z. Chen, J. Shlens, Scene Transformer: A unified architecture for predicting multiple agent trajectories, 2022, http://dx.doi.org/10.48550/arXiv.2106.08417, arXiv:2106.08417.

[27] J. Liu, X. Wang, H. Lin, F. Yu, Gsaa: A novel graph spatiotemporal attention algorithm for smart city traffic prediction, ACM Trans. Sensor Netw. (2023).

[28] A. Cui, S. Casas, K. Wong, S. Suo, R. Urtasun, GoRela: Go relative for viewpoint-invariant motion forecasting, 2022, https://arxiv.org/abs/2211.02545v2.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).

[30] R. Girgis, F. Golemo, F. Codevilla, M. Weiss, J.A. D'Souza, S.E. Kahou, F. Heide, C. Pal, Latent variable sequential set transformers for joint multi-agent motion prediction, 2021, arXiv preprint arXiv:2104.00563.

[31] N. Nayakanti, R. Al-Rfou, A. Zhou, K. Goel, K.S. Refaat, B. Sapp, Wayformer: Motion forecasting via simple & efficient attention networks, in: 2023 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2023, pp. 2980–2987.

[32] Z. Zhou, L. Ye, J. Wang, K. Wu, K. Lu, Hivt: Hierarchical vector transformer for multi-agent motion prediction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 8823–8833.

[33] Z. Zhang, A. Liniger, C. Sakaridis, F. Yu, L.V. Gool, Real-time motion prediction via heterogeneous polyline transformer with relative pose encoding, in: Thirty-Seventh Conference on Neural Information Processing Systems, 2023.

[34] D. Park, H. Ryu, Y. Yang, J. Cho, J. Kim, K.-J. Yoon, Leveraging future relationship reasoning for vehicle trajectory prediction, in: The Eleventh International Conference on Learning Representations, 2022.

[35] C. Liu, X. Sun, J. Wang, H. Tang, T. Li, T. Qin, W. Chen, T.-Y. Liu, Learning causal semantic representation for out-of-distribution prediction, Adv. Neural Inf. Process. Syst. 34 (2021) 6155–6170.

[36] K.A. Keith, D. Jensen, B. O'Connor, Text and causal inference: A review of using text to remove confounding from causal estimates, 2020, arXiv preprint arXiv:2005.00649.

[37] K. Tang, Y. Niu, J. Huang, J. Shi, H. Zhang, Unbiased scene graph generation from biased training, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 3716–3725.

[38] D. Zhang, H. Zhang, J. Tang, X.-S. Hua, Q. Sun, Causal intervention for weakly-supervised semantic segmentation, Adv. Neural Inf. Process. Syst. 33 (2020) 655–666.

[39] O. Ahmed, F. Träuble, A. Goyal, A. Neitz, Y. Bengio, B. Schölkopf, M. Wüthrich, S. Bauer, Causalworld: A robotic manipulation benchmark for causal structure and transfer learning, 2020, arXiv preprint arXiv:2010.04296.

[40] X. Lin, Y. Chen, G. Li, Y. Yu, A causal inference look at unsupervised video anomaly detection, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, 2022, pp. 1620–1629.

[41] P. Deng, Y. Zhao, J. Liu, X. Jia, M. Wang, Spatio-temporal neural structural causal models for bike flow prediction, 2023, arXiv preprint arXiv:2301.07843.

[42] J. Pearl, et al., Models, Reasoning and Inference, Vol. 19, CambridgeUniversityPress, Cambridge, UK, 2000, p. 3, (2).

[43] M.-F. Chang, J.W. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, J. Hays, Argoverse: 3d tracking and forecasting with rich maps, in: Conference on Computer Vision and Pattern Recognition, CVPR, 2019.

[44] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J.K. Pontes, D. Ramanan, P. Carr, J. Hays, Argoverse 2: Next generation datasets for self-driving perception and forecasting, in: Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021), 2021.

[45] B. Varadarajan, A. Hefny, A. Srivastava, K.S. Refaat, N. Nayakanti, A. Cornman, K. Chen, B. Douillard, C.P. Lam, D. Anguelov, et al., Multipath++: Efficient information fusion and trajectory aggregation for behavior prediction, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 7814–7821.

[46] J. Gu, C. Sun, H. Zhao, Densetnt: End-to-end trajectory prediction from dense goal sets, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 15303–15312.

[47] ArgoAI-Argoverse, Argoverse 2: Motion forecasting competition, [Online], available: https://eval.ai/web/challenges/challenge-page/1719/overview/.

[48] J. Schmidt, J. Jordan, F. Gritschneder, K. Dietmayer, Crat-pred: Vehicle trajectory prediction with crystal graph convolutional neural networks and multi-head self-attention, in: 2022 International Conference on Robotics and Automation, ICRA, IEEE, 2022, pp. 7799–7805.