

Geographical Information Alignment Boosts Traffic Analysis via Transpose Cross-attention

1st Xiangyu Jiang
Clemson University
 Clemson, SC, USA
 xiangyu@clemson.edu

2nd Xiwen Chen
Clemson University
 Clemson, SC, USA
 xiwenc@g.clemson.edu

3rd Hao Wang
Clemson University
 Clemson, SC, USA
 hao9@g.clemson.edu

4th Abolfazl Razi
Clemson University
 Clemson, SC, USA
 arazi@clemson.edu

Abstract—Traffic accident prediction is crucial for enhancing road safety and mitigating congestion, and recent Graph Neural Networks (GNNs) have shown promise in modeling the inherent graph-based traffic data. However, existing GNN-based approaches often overlook or do not explicitly exploit geographic position information, which often plays a critical role in understanding spatial dependencies. This is also aligned with our observation, where accident locations are often highly relevant. To address this issue, we propose a plug-in-and-play module for common GNN frameworks, termed Geographic Information Alignment (GIA). This module can efficiently fuse the node feature and geographic position information through a novel Transpose Cross-attention mechanism. Due to the large number of nodes for traffic data, the conventional cross-attention mechanism performing the node-wise alignment may be infeasible in computation-limited resources. Instead, we take the transpose operation for Query, Key, and Value in the Cross-attention mechanism, which substantially reduces the computation cost while maintaining sufficient information. Experimental results for both traffic occurrence prediction and severity prediction (severity levels based on the interval of recorded crash counts) on large-scale city-wise datasets confirm the effectiveness of our proposed method. For example, our method can obtain gains ranging from 1.3% to 10.9% in F1 score and 0.3% to 4.8% in AUC¹.

Index Terms—Traffic Accident Analysis, Non-conventional Big data, Graph Data, Transpose Cross-attention, Position Feature Alignment

I. INTRODUCTION

Road traffic accidents pose a significant global health and safety issue, as reported by the World Health Organization (WHO). The 2018 report estimated 1.35 million fatalities per year, and though this number slightly decreased to 1.19 million in the 2023 report, the issue remains critical [1], [2]. To mitigate the risks associated with road traffic incidents, predictive analytics have become a promising approach in modern traffic management systems. The importance of modeling traffic incident risks is well-recognized in the field of urban planning and public safety [3]–[5]. In the machine learning community, the use of traffic data is often large-scale, and it can be inherently possessed to a graph/network structure [6], [7]. For example, nodes represent physical locations, and edges represent the roads connecting them [8]. Therefore,

¹This material is based upon the work supported by the National Science Foundation under Grant Number 2204721 and partially supported by our collaborative project with MIT Lincoln Lab under Grant Number 7000612889.

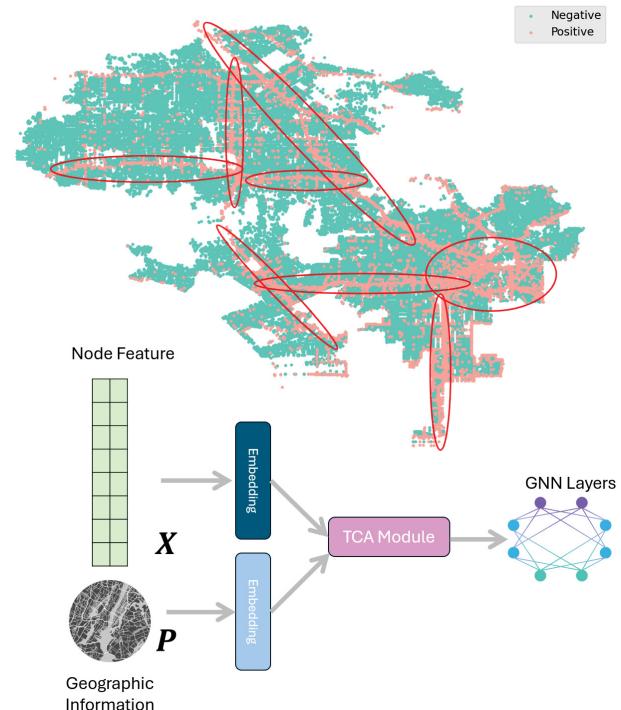


Figure 1: **Top:** Traffic Accident Map in Los Angeles shows a strong correlation between traffic accidents and location information. Positive denotes the crash/accident occurrence. We mark some exemplary accident clusters. **Bottom:** The proposed Geographic Information Alignment (GIA) through the Transpose Cross-attention (TCA) mechanism.

learning graph-based representations of road networks enables understanding traffic flow and identifying high-risk areas for accidents. It is noteworthy that graph-based data is often considered non-conventional due to its irregular structure with dynamics topology. To address this issue, several Graph Neural Networks (GNNs) are proposed [9]–[12]. Subsequently, many of the following works [8], [13]–[16] apply GNN to different traffic tasks.

However, a general issue with GNNs is their permutation-invariant nature, which means they do not consider the order of nodes. In the context of traffic analysis, these models may neglect or do not explicitly exploit geographic position information. This can be problematic in traffic accident

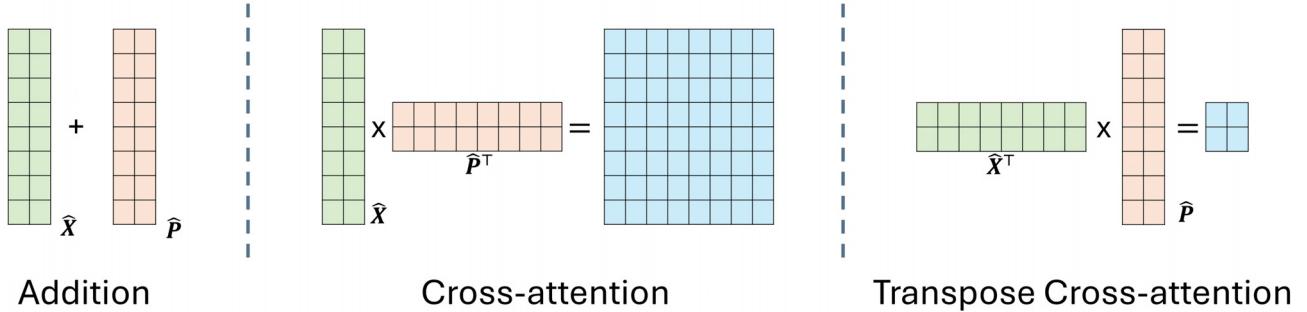


Figure 2: Comparison of different fusion strategies. **Left:** Simply take addition, **Middle:** The conventional Cross-attention, and **Right:** Our proposed Transpose Cross-attention (TCA) for feature-alignment.

prediction because geographical context and the precise location of nodes are highly relevant. This is also aligned with our observation of real-world datasets [8]. In Fig. 1(**Left**), we can observe the accident distribution is often grouped according to some geographical patterns. Areas with dense traffic networks or critical intersections tend to experience higher accident frequencies, highlighting the importance of geographic context in traffic incident analysis.

Therefore, we aim to explicitly harness this geographical information to GNN, and to this end, we propose a plug-in-and-play technique termed *Geographical Information Alignment* (GIA) as shown in Fig. 1(**Right**), which can be integrated into general GNN frameworks. Intuitively, we can employ the Cross-attention mechanism to fuse the node feature and the geographical information. We note that the number of nodes N often is substantial (e.g., over 10k), and directly applying Cross-attention requires $\mathcal{O}(N^2)$ complexity in memory and time, which can challenge the computation resources (In fact, we encountered out-of-memory issues when attempting this approach using one NVIDIA V100 GPU). Instead, we propose a **Transpose** Cross-attention mechanism, where we construct attention between each dimension and attempt to align features from edges and geographical information. This substantially reduces the complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(d^2)$, where d denotes the number of dimensions and is much smaller than N (e.g., $d = 16$ in our experiments). Fig. 2 illustrates the different fusion strategies. The left one simply takes the addition of node features and position representation, which tends to limited learning performance. The middle one shows the conventional Cross-attention to fuse two features, which has a high computational cost, and the right one shows our proposed Transpose Cross-attention, which performs feature-wise alignment.

Our analysis is based on a database of national traffic accident reports provided by [8]. We select some representative cities and states in our experiments, including Dallas, Houston, Miami, Orlando, Los Angeles, and New York. We consider two important tasks involving accident occurrence prediction and severity prediction. The empirical results confirm the effectiveness of our enhanced GNN model, showing significant improvements in predictive performance across multiple state-of-the-art GNN models.

II. RELATED WORK

Graph Neural Networks in Traffic Analysis. In current mainstream research, network, and graph data are widely used to deal with problems related to traffic flow [17], [18]. Therefore, many existing studies focus on network analysis [19]–[21]. However, compared to common data types in the field of machine learning (such as images) [18], graph data are more difficult to process because their topological structure is often dynamic and large in scale [22]. To tackle this issue, the community developed and improved many methods, and the most popular paradigm is GNN. At a very early stage, spectral methods were introduced to perform convolution operations using the graph Laplacian operator [23]. Afterward, Graph Convolutional Networks (GCN) [24] significantly reduce the computational complexity to enhance its generalizability. Subsequently, many variants of GNN are developed with different modules [10], [11], [25], [26], such as attention mechanism [27] or Transformer [12].

Graph Neural Networks (GNNs) have become increasingly important in traffic analysis due to their ability to model traffic networks as graphs, where nodes represent traffic entities like vehicles or road intersections, and edges represent interactions such as traffic flow or road connections. GNNs are particularly effective for tasks like traffic flow prediction, route optimization, and accident risk assessment. For example, recent models such as FPTN [13] enhance traffic forecasting by leveraging sensor-based data division and a Transformer encoder, which reduces computational demands while improving accuracy. Similarly, DyHSL [14] utilizes a hypergraph neural network (HGNN) to capture dynamic and interactive spatio-temporal relationships, significantly improving forecasting performance across multiple datasets. DSTAGNN [15] focuses on dynamically modeling spatial-temporal interactions in road networks, employing enhanced multi-head attention mechanisms and multi-scale gated convolutions for more precise traffic predictions. Moreover, Bi-STAT [16] advances traffic forecasting by using adaptive spatial-temporal transformers, effectively handling diverse tasks and leveraging historical data for more accurate predictions. Finally, STFGNN [28] integrates data-driven temporal and spatial graphs and gated convolutions, successfully managing long sequences of traffic data, making it a robust model for traffic forecasting.

Our work is based on [8] for the large-scale traffic accident analysis with GNNs. In this work, the authors investigate a series of recent state-of-the-art GNN models (such as [9]–[12]) and further propose a new GNN-based method to analyze the traffic accident occurrence and severity. We reiterate that a significant limitation of these approaches is their reliance on topological features of the graph while often ignoring or not explicitly employing the absolute spatial coordinates of the nodes, making it difficult to fully exploit prior to the geographic information. Accordingly, this may limit the models' predictive power for tasks such as traffic accident prediction. Hence, in this work, we propose a method to align position information with node features to mitigate this concern.

Positional Encoding. The proposed method is relevant to positional encoding techniques; however, we realize that there exists an essential difference between them. Position encoding has been extensively studied in Natural Language Processing (NLP), particularly in the context of the Transformer architecture [29], where sinusoidal position encoding is used to help models capture the sequential order of words. This technique is unlearnable, which often leads to limited performance. In the domain of GNNs, researchers have started to investigate methods to encode positional information within graphs. A known work presented in [30] introduced positional encoding for GNN. However, similar to sinusoidal position encoding, they directly add position information and latent features, which apparently degenerate learning ability. Instead, we propose the Transpose Cross-attention to better and efficiently integrate position and node features. In traffic analysis, although some recent studies have experimented with incorporating node coordinates into GNNs by injecting the coordinate information as node features [31], these efforts often lack a systematic approach to position encoding. In contrast, our method offers a universal strategy to elegantly impose the position information to GNNs.

III. METHOD

In this section, we present the problem formulation for traffic accident prediction, followed by our proposed method, the *Geographical Information Alignment* (GIA) through a novel Transpose Cross-attention mechanism.

A. Problem Formulation

Following [8], the task of traffic accident prediction is formulated as a node-wise classification problem in a monolithic graph under the popular transductive setup [32]. In this context, each dataset (i.e., a city) is presented as a large graph with N nodes and E edges. Subsequently, this graph can be presented by three matrices $\mathbf{A} \in \mathbb{R}^{N \times N}$, $\mathbf{X} \in \mathbb{R}^{N \times D_1}$, and $\mathbf{E} \in \mathbb{R}^{E \times D_2}$, denoting the adjacency matrix, node embedding, and edge embedding, respectively. Here, D_1 and D_2 denote the number of dimensions of node and edge features, respectively. We also have partial node labels, with the index set of these labeled nodes denoted as $\mathcal{V}_{\text{train}}$. The set of nodes whose labels are unknown and need to be predicted is denoted as $\mathcal{V}_{\text{test}}$. The goal is to classify nodes based on accident

occurrence or severity. Accident occurrence prediction is a binary classification, indicating whether an accident has occurred at a node or not. In contrast, the severity prediction involves eight categories, where each category is defined based on an interval of recorded crash counts. Please refer to [8] for more detail.

B. Proposed Method

Here, we delve into the proposed *Geographical Information Alignment* and demonstrate how to incorporate it with general GNN models.

1) Position Encoding: The goal is to encode the semantic information of a node's absolute positions into its feature representation. First, we can obtain the embedding of the node features as $\hat{\mathbf{X}} \leftarrow f_{\text{embed}}(\mathbf{X}) \in \mathbb{R}^{N \times D_n}$. Here N and D_n denote the number of nodes and the dimensions of latent space, respectively. This is because, in the raw data space, different features represent different physical meanings (e.g., junction types, speed limit, etc.). Directly imposing position information in that space may constrain the learning ability. We also want to mention that the embedding layer $f_{\text{embed}}(\cdot)$ comes from the original architecture of a GNN, and we do not intend to modify it. Given a geographical position $\mathbf{P} \in \mathbb{R}^{N \times 2}$, we then map it to a latent space via a linear layer to match the dimensions of the input edge feature, which is denoted as $\hat{\mathbf{P}} \leftarrow f_{\text{linear}}(\mathbf{P}) \in \mathbb{R}^{N \times D_n}$. Subsequently, we consider to fuse it with $\hat{\mathbf{X}} \in \mathbb{R}^{N \times D_n}$ and $\hat{\mathbf{P}} \in \mathbb{R}^{N \times D_n}$. An intuitive solution is employing the cross-attention mechanism to perform node-wise alignment:

$$\mathbf{H} = \text{softmax} \left(\underbrace{\frac{\hat{\mathbf{X}} \hat{\mathbf{P}}^\top}{\sqrt{D_n}}}_{N \times N} \right) \hat{\mathbf{P}}. \quad (1)$$

Following the common operation, we can apply the respective linear layers for the Query, Key, and Value. However, the main concern here is that in our graph-based problem, the number of nodes is often very large, and we have $N \gg D_n$, which leads to the computation and memory cost for the attention matrix requiring an $\mathcal{O}(n^2)$. This may be unacceptable for a resource-constrained platform.

To address the concern, we propose the *Transpose Cross-attention* mechanism. Instead of performing the node-wise alignment, we prefer feature-wise alignment. Mathematically,

$$\mathbf{H} = \text{softmax} \left(\underbrace{\frac{\hat{\mathbf{X}}^\top \hat{\mathbf{P}}}{\sqrt{N}}}_{D_n \times D_n} \right) \hat{\mathbf{P}}^\top. \quad (2)$$

This substantially reduces the complexity from $\mathcal{O}(n^2)$ to $\mathcal{O}(D_n^2)$. Another justification is this operation can also contain sufficient information from the perspective of linear algebra. This is because $\text{rank}(\hat{\mathbf{X}} \hat{\mathbf{P}}^\top) = \text{rank}(\hat{\mathbf{X}}^\top \hat{\mathbf{P}}) =$

$\min(N, D_n) = D_n$. We also added a residual term to stabilize the training as,

$$\hat{\mathbf{X}} \leftarrow f_{\text{linear}}(\mathbf{X}) + \text{softmax} \left(\frac{\hat{\mathbf{X}}^T \hat{\mathbf{P}}}{\sqrt{N}} \right) \hat{\mathbf{P}}^T. \quad (3)$$

The overall architecture is presented in Fig. 1.

IV. EXPERIMENTS

Setup. In this section, we evaluate the performance of the proposed GNN-based model variants for traffic incident prediction. We employ multiple model baselines and analyze their performance on large-scale traffic datasets provided by [8]. We select representative cities and states, such as Miami, Los Angeles and New York to ensure geographical diversity and capture a wide range of traffic patterns. These datasets vary in size and structure, providing a comprehensive testbed for traffic analysis. Detailed information on the number of nodes and edges for each dataset can be found in Table I. It is noteworthy to mention that the datasets are very unbalanced, which challenges most deep learning or machine learning models.

Dataset	Miami (FL)	Los Angeles (CA)	Orlando (FL)	Dallas (TX)	Houston (TX)	New York (NY)
# of Nodes	8461	49251	7513	36150	59711	55404
# of Edges	22648	135547	18216	92348	148937	140005

Table I: Summary of the datasets used in the experiments.

The dataset is stratified into training (60%), validation (20%), and test (20%) sets, with stratification based on the incident labels. All features are normalized using the MinMaxScaler for each split.

Baselines. To validate that our proposed method is model-agnostic, we choose 12 different state-of-the-art GNN models, including (1) **GCN**: Graph Convolutional Networks [9], (2) **ChebNet**: graph convolutional networks with spectral filtering [10], (3) **ARMANet**: Graph neural networks with convolutional auto-regressive moving average (ARMA) filters [25]. (4) **GraphSAGE**: inductive representation learning [11], (5) **TAGCN**: Topology adaptive graph convolutional networks [26], (6) **GIN**: Graph Isomorphism Networks [33], (7) **GAT**: Graph attention networks [27], (8) **MPNN**: Message Passing Neural Network [34], (9) **CGC**: Crystal graph convolutional neural network [35], (10) **GEN**: GENeralized graph convolutional neural networks [36], (11) **Graphformer**: Graph transformers [12], and (12) **TRAVEL**: the recent GNN model proposed for our problem [8]. For the sake of fair comparison, Each model, including the baselines and our proposed variants, applies a unified training regimen, where we adjust the training epochs of 500 to make the model fully converge. The other hyperparameters follow the default setting in [8].

Main Results. The main results are shown in Table II. The primary observation is that by the proposed GIA, all GNN models demonstrate significant improvements across different cities. Specifically, in the occurrence prediction task (Table II), GNN models achieve a 1.53% to 10.9% gain in F1 score and a 1.26% to 4.8% gain in AUC when incorporating our proposed

GIA. For instance, models like GCN-GIA, ChebNet-GIA, and GIN-GIA demonstrate F1 score improvements of 1.53%, 2.6%, and 4.51%, respectively. The model GIN-GIA shows an exceptional improvement in the city of Miami, achieving a remarkable 10.9% increase in F1 score (see Fig. 3) and a 4.0% increase in AUC.

More importantly, the TRAVEL-GIA model, which previously had the best performance, achieved an additional 1.74% gain in F1 score and a 1.26% gain in AUC compared to its non-PE variant. Likewise, in GEN-GIA, the F1 score shows a 4.69% improvement, with notable enhancements in cities like Los Angeles and Dallas. These results underscore the critical role of incorporating geographic positional information to boost GNN performance on traffic prediction tasks.

Similarly, in severity prediction tasks (Table III), all GNN models experience consistent improvements. In this task, due to the already high performance of the original models (around 85%), the improvements are relatively modest but still significant. Notably, Transformer-GIA achieves the highest F1 score of 86.59%, with an improvement of 0.79%. The CGC-GIA model also exhibits strong performance, reaching an F1 score of 86.48% with a 0.68% improvement. TRAVEL-GIA shows the largest relative improvement of 1.12%, reaching an F1 score of 85.82%. These results confirm that integrating GIA leads to measurable performance gains, even in tasks where baseline models already perform well.

In conclusion, the addition of GIA provides a consistent performance boost across multiple models in the severity prediction task despite the already high baseline performance. This demonstrates that GIA is an effective enhancement for traffic incident severity prediction, capable of improving model performance even when the original models are already achieving strong results. The observed improvements, while modest in absolute terms, are significant given the high initial performance levels and highlight the potential of GIA in refining state-of-the-art models for this critical task.

Ablation Study. Table IV presents the results of our ablation study, where we compare the performance of different variations of GNN models across three methods: ARMANet, GIN, and GEN. The first row corresponds to the original model without any positional encoding. The second row shows we only keep the linear residual term (i.e., the first term presented in Eq.3). This can also be viewed as a simple way for linear positional encoding. Finally, we will add the most popular Sinusoidal PE for comparison. As shown in Table IV, both linear and Sinusoidal positional encodings provide improvements over the original models, with Linear Encoding leading to a slightly higher improvement than Sinusoidal Encoding across all three models. This may be because it is learnable. Finally, we show the full model, demonstrating the performance achieved when Geographic Interaction Attention (GIA) is integrated into the system. This yields the highest improvements, with F1 scores increasing by 3.26%, 3.77%, and 4.9% for ARMANet, GIN, and GEN, respectively, demonstrating the effectiveness of each component in our design for these traffic analysis tasks.

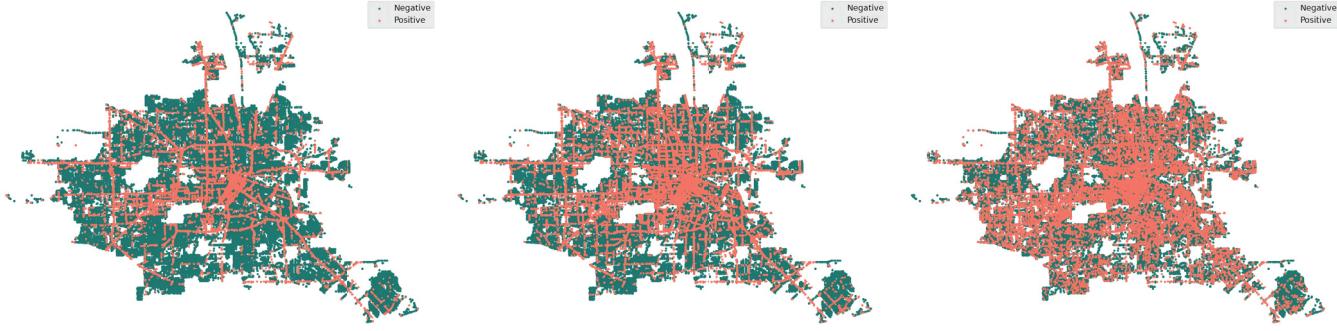


Figure 3: Comparison of traffic accident predictions in the Houston area using the GEN model(left), GEN-GIA model(middle), and the ground truth(right). The GEN-GIA model shows improved accuracy in predicting high-accident areas by integrating positional encoding.

Dataset	Mean		Miami (FL)		Los Angeles(CA)		Orlando (FL)		Dallas (TX)		Houston (TX)		New York (NY)	
	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC
Method														
GCN	34.53	72.88	20.0±3.3	68.5±3.3	40.2±1.1	80.4±0.3	51.6±0.8	73.1±1.2	39.8±1.9	73.1±0.4	16.4±1.3	66.7±0.2	39.2±3.7	75.5±0.4
GCN-GIA	36.06	75.54	21.73±2.77	72.69±0.18	40.74±0.47	82.79±0.15	52.32±0.18	76.65±0.16	41.16±0.80	74.00±0.11	18.83±1.47	67.57±0.20	41.57±0.91	79.52±0.13
Δ	1.53	2.66	1.7	4.1	0.5	2.3	0.7	3.5	1.3	0.9	2.4	0.8	2.3	4
ChebNet	36.72	75.45	20.7±2.9	71.3±3.6	39.8±1.8	81.0±0.3	53.1±0.6	76.7±1.6	42.0±0.5	75.8±0.4	23.8±0.5	69.6±0.5	40.9±4.3	78.3±1.1
ChebNet-GIA	39.32	77.29	25.80±0.83	73.16±0.87	42.43±1.35	83.86±0.33	55.52±1.50	78.16±0.63	43.46±0.07	76.64±0.08	25.30±0.96	70.26±0.27	43.41±0.74	81.68±0.16
Δ	2.6	1.84	5.1	1.8	2.6	2.8	2.4	1.4	1.4	0.8	1.5	0.6	2.5	3.3
ARMANet	36.37	74.77	19.2±3.3	69.5±3.5	40.8±1.0	80.9±0.4	51.5±1.3	75.7±1.4	41.2±0.5	75.6±0.2	23.1±0.4	69.2±0.7	42.4±1.1	77.7±0.6
ARMANet-GIA	39.51	77.25	25.69±0.29	74.32±0.06	43.70±1.30	83.55±0.35	57.43±0.20	78.44±0.45	43.66±1.13	76.23±0.14	24.53±0.69	69.90±0.34	42.03±0.78	81.04±0.42
Δ	3.14	2.48	6.4	4.8	2.9	2.6	5.9	2.7	2.4	0.6	1.4	0.7	-0.4	3.3
GraphSAGE	37.55	73.57	20.7±2.4	67.6±2.8	41.6±0.5	80.5±0.3	52.6±1.3	74.1±1.2	44.2±0.5	74.4±0.3	23.7±0.4	68.5±0.4	42.5±1.1	76.3±0.1
GraphSAGE-GIA	38.94	75.98	25.44±3.05	71.41±1.65	44.05±1.31	82.86±0.68	54.51±1.00	77.00±0.64	42.49±0.66	75.16±0.12	24.50±0.41	69.74±0.53	42.68±0.94	79.68±0.01
Δ	1.39	2.41	4.7	3.8	2.4	2.3	1.9	2.9	-1.8	0.7	0.8	1.2	0.1	3.3
TAGCN	39.85	77.40	25.2±1.1	73.5±2.4	49.5±0.7	84.7±0.2	53.3±2.5	77.2±1.2	45.4±0.4	77.0±0.5	23.7±0.6	70.5±0.3	42.0±1.1	81.5±0.2
TAGCN-GIA	42.77	79.11	30.43±1.68	75.64±0.75	53.35±1.78	86.24±0.21	57.87±0.62	80.13±0.46	47.38±0.65	77.66±0.01	25.20±0.92	70.95±0.26	42.40±0.60	84.06±0.30
Δ	2.92	1.71	5.2	2.1	3.8	1.5	4.5	2.9	1.9	0.6	1.5	0.4	0.4	2.5
GIN	37.17	75.57	22.8±1.2	72.7±2.6	41.6±0.7	81.8±0.2	54.7±1.4	76.6±1.1	41.3±2.0	75.2±0.3	20.9±1.0	68.0±0.3	41.7±2.1	79.1±0.5
GIN-GIA	41.68	77.94	33.77±0.15	76.71±3.35	46.51±0.54	84.37±0.49	57.81±1.27	78.94±0.08	43.81±0.61	76.93±0.35	24.50±1.49	68.78±0.07	43.66±0.47	81.92±0.28
Δ	4.51	2.37	10.9	4	4.9	2.5	3.1	2.3	2.5	1.7	3.6	0.7	1.9	2.8
GAT	36.93	73.47	22.6±1.5	68.3±3.0	41.6±0.4	80.9±0.2	55.3±1.3	74.1±1.0	42.1±1.5	73.6±0.3	17.8±0.8	67.3±0.3	42.2±0.5	76.6±0.4
GAT-GIA	39.13	76.11	27.69±2.23	72.03±2.31	45.28±1.24	83.92±0.49	56.55±1.11	77.59±0.08	42.01±0.05	74.59±0.10	20.91±1.15	68.28±0.33	42.37±0.03	80.24±0.03
Δ	2.2	2.64	5	3.7	3.6	3	1.2	3.4	-0.1	0.9	3.1	0.9	0.1	3.6
MPNN	44.63	81.32	38.8±2.1	82.4±1.0	46.0±1.6	83.9±0.2	61.4±2.5	81.84±0.7	48.5±1.9	79.44±0.4	28.2±1.7	73.5±0.5	44.9±0.8	86.94±0.4
MPNN-GIA	47.01	82.96	43.03±1.44	84.72±0.38	48.88±0.98	86.91±0.51	63.72±1.05	82.45±0.95	50.23±0.73	81.26±0.60	30.72±0.43	74.61±0.26	45.49±0.08	87.84±0.19
Δ	2.38	1.64	4.2	2.3	2.8	3	2.3	0.6	1.7	1.8	2.5	1.1	0.5	0.9
CGC	42.47	79.83	34.4±2.7	79.5±1.5	45.0±1.2	81.5±0.2	59.0±2.1	81.1±0.8	48.5±0.5	79.2±0.7	27.3±1.9	72.3±0.1	40.6±1.2	85.4±0.8
CGC-GIA	46.08	82.29	38.33±3.06	84.16±0.26	47.98±1.94	85.88±0.57	63.12±0.60	82.74±0.47	51.20±0.50	80.37±0.33	33.02±2.86	73.13±0.13	42.83±0.15	87.47±0.25
Δ	3.61	2.46	3.9	4.6	2.9	4.3	4.1	1.6	2.7	1.1	5.7	0.8	2.2	2
Graphformer	45.13	81.32	37.7±3.3	81.0±1.9	48.9±0.3	83.8±0.3	62.9±1.6	82.0±0.7	49.8±0.7	80.0±0.7	28.4±0.7	73.9±0.4	43.1±0.7	87.2±0.4
Transformer-GIA	48.02	83.17	42.04±1.35	85.04±0.47	51.74±1.45	86.59±0.51	64.52±0.83	83.02±0.63	51.33±0.80	80.96±0.21	33.22±0.58	74.83±0.24	45.26±0.74	88.56±0.16
Δ	2.89	1.85	4.3	4	2.8	2.7	1.6	1	1.5	0.9	4.8	0.9	2.1	1.3
GEN	49.07	80.97	44.9±3.1	81.0±2.4	48.6±6.2	82.7±0.9	63.0±1.1	81.2±0.9	56.5±1.7	79.5±0.1	34.1±6.0	73.7±0.4	47.3±1.4	87.7±0.9
GEN-GIA	53.76	83.62	51.96±0.66	86.14±0.55	55.44±0.17	87.67±0.22	65.42±1.04	82.79±1.08	56.14±0.99	81.06±0.49	42.95±0.94	75.35±0.89	50.65±2.37	88.70±0.07
Δ	4.69	2.65	7	5.1	6.8	4.9	2.4	1.5	-0.4	1.5	8.8	1.6	3.3	1
TRAVEL	54.62	82.77	51.9±1.0	84.9±0.9	55.3±0.9	85.9±0.5	65.0±0.4	82.3±0.4	58.0±0.9	80.8±0.7	46.4±0.7	74.5±0.3	51.1±0.9	88.2±0.2
TRAVEL-GIA	56.36	84.03	54.91±1.04	86.44±0.41	58.07±0.47	87.99±0.29	65.71±0.21	83.58±0.13	58.29±0.12	81.59±0.22	47.34±0.66	75.79±0.58	53.83±0.95	88.76±0.05
Δ	1.74	1.26	3	1.5	2.7	2	0.7	1.2	0.2	0.7	0.9	1.2	2.7	0.5

Table II: City-wise accident occurrence prediction results in terms of F1 score and AUC. Δ denotes the gain obtained by imposing our proposed Geographical Information Alignment(with the suffix "-GIA") in the neural networks.

Model	F1 (%)	ΔF1%
Transformer-GIA	86.59	+0.79
CGC-GIA	86.48	+0.68
TRAVEL-GIA	85.82	+1.12

Table III: Comparison of different models on Severity Prediction tasks.

For the severity prediction tasks (Table V), we observe a similar trend, with GIA once again providing the largest performance boost. Although the baseline models already perform well, GIA enhances F1 scores by 0.95% for ARMANet, 0.64% for GIN, and 0.75% for GEN, confirming its positive impact even in high-performing models.

Model	ARMANet		GIN		GEN	
	F1	Δ F1	F1	Δ F1	F1	Δ F1
Baseline	36.37	-	49.07	-	49.07	-
+GIA w/o TCA	38.81	+2.44	38.49	+1.32	51.59	+2.52
+GIA	39.63	+3.26	40.94	+3.77	53.97	+4.9
Sinusoidal	38.28	+1.91	38.33	+1.16	51.05	+1.98

Table IV: Ablation study comparing the performance of different GNN variants on accident occurrence prediction tasks with and without positional encoding techniques. The methods include the original GNN models (baseline), models without TCA (only linear residual term), models with Sinusoidal Encoding, and the full model.

Model	ARMANet		GIN		GEN	
	F1	Δ F1	F1	Δ F1	F1	Δ F1
Baseline	81.4	-	81.3	-	83.6	-
+GIA w/o TCA	81.42	+0.02	81.58	+0.28	83.67	+0.07
+GIA	82.35	+0.95	81.94	+0.64	84.35	+0.75
Sinusoidal	81.5	+0.1	81.41	+0.11	83.89	+0.29

Table V: Ablation study comparing the performance of different GNN variants on severity prediction tasks with and without positional encoding techniques. The methods include the original GNN models (baseline), models without TCA (only linear residual term), models with Sinusoidal Encoding, and the full GIA model.

V. CONCLUSION

In this paper, we aim to address the common limitation of GNNs, which often fail or explicitly incorporate spatial information in traffic prediction tasks. To this end, we propose a Geographical Information Alignment (GIA) module to enhance GNNs for addressing this constraint. Specifically, the GIA effectively integrates geographic positional information with node features using the proposed Transpose Cross-attention mechanism with a substantially low computation overhead. Our extensive experiments, conducted on large-scale traffic datasets from various cities, demonstrate significant performance improvements across multiple state-of-the-art GNN models, both in accident occurrence and severity prediction tasks. For example, our method can obtain up to 10.9% and 4.8% gain in F1 score and AUC, respectively.

REFERENCES

- [1] WorldHealthOrganization, “Global status report on road safety 2018,” 2018.
- [2] ——, “Global status report on road safety 2023,” 2023.
- [3] M. A. Fadhel, A. M. Duham, A. Saihood, A. Sewify, M. N. Al-Hamadani, A. Albahri, L. Alzubaidi, A. Gupta, S. Mirjalili, and Y. Gu, “Comprehensive systematic review of information fusion methods in smart cities and urban environments,” *Information Fusion*, p. 102317, 2024.
- [4] A. Razi, X. Chen, H. Li, H. Wang, B. Russo, Y. Chen, and H. Yu, “Deep learning serves traffic safety analysis: A forward-looking review,” *IET Intelligent Transport Systems*, vol. 17, no. 1, pp. 22–71, 2023.
- [5] A. Bastola, J. Brinkley, H. Wang, and A. Razi, “Driving towards inclusion: Revisiting in-vehicle interaction in autonomous vehicles,” *arXiv preprint arXiv:2401.14571*, 2024.
- [6] S. Rahmani, A. Baghbani, N. Bouguila, and Z. Patterson, “Graph neural networks for intelligent transportation systems: A survey,” *IEEE Transactions on Intelligent Transportation Systems*, 2023.
- [7] X. Chen, H. Wang, A. Razi, B. Russo, J. Pacheco, J. Roberts, J. Wishart, L. Head, and A. G. Baca, “Network-level safety metrics for overall traffic safety assessment: A case study,” *IEEE Access*, vol. 11, pp. 17 755–17 778, 2022.
- [8] B. Huang, B. Hooi, and K. Shu, “Tap: A comprehensive data repository for traffic accident prediction in road networks,” in *Proceedings of the 31st ACM International Conference on Advances in Geographic Information Systems*, 2023, pp. 1–4.
- [9] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” *arXiv preprint arXiv:1609.02907*, 2016.
- [10] M. Defferrard, X. Bresson, and P. Vandergheynst, “Convolutional neural networks on graphs with fast localized spectral filtering,” *Advances in neural information processing systems*, vol. 29, pp. 3844–3852, 2016.
- [11] W. L. Hamilton, R. Ying, and J. Leskovec, “Inductive representation learning on large graphs,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 1025–1035.
- [12] Y. Shi, Z. Huang, S. Feng, H. Zhong, W. Wang, and Y. Sun, “Masked label prediction: Unified message passing model for semi-supervised classification,” 2021.
- [13] J. Zhang, J. Jin, J. Tang, and Z. Qu, “Fptn: Fast pure transformer network for traffic flow forecasting,” in *International Conference on Artificial Neural Networks*. Springer, 2023, pp. 382–393.
- [14] Y. Zhao, X. Luo, W. Ju, C. Chen, X.-S. Hua, and M. Zhang, “Dynamic hypergraph structure learning for traffic flow forecasting,” in *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2023, pp. 2303–2316.
- [15] S. Lan, Y. Ma, W. Huang, W. Wang, H. Yang, and P. Li, “Dstagnn: Dynamic spatial-temporal aware graph neural network for traffic flow forecasting,” in *International conference on machine learning*. PMLR, 2022, pp. 11 906–11 917.
- [16] C. Chen, Y. Liu, L. Chen, and C. Zhang, “Bidirectional spatial-temporal adaptive transformer for urban traffic flow forecasting,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 10, pp. 6913–6925, 2022.
- [17] X. Chen, H. Li, R. Amin, and A. Razi, “Learning on bandwidth constrained multi-source data with mimo-inspired dpp map inference,” *IEEE Transactions on Machine Learning in Communications and Networking*, vol. 2, pp. 1341–1356, 2024.
- [18] A. Bastola, H. Wang, X. Chen, and A. Razi, “FedMIL: Federated-Multiple Instance Learning for Video Analysis with Optimized DPP Scheduling ,” in *2024 20th DCOSS-IoT*, pp. 109–116.
- [19] D. J. Watts and S. H. Strogatz, “Collective dynamics of ‘small-world’networks,” *nature*, vol. 393, no. 6684, pp. 440–442, 1998.
- [20] S. Boccaletti, G. Bianconi, R. Criado, C. I. Del Genio, J. Gómez-Gardenes, M. Romance, I. Sendina-Nadal, Z. Wang, and M. Zanin, “The structure and dynamics of multilayer networks,” *Physics reports*, vol. 544, no. 1, pp. 1–122, 2014.
- [21] A.-L. Barabási and R. Albert, “Emergence of scaling in random networks,” *science*, vol. 286, no. 5439, pp. 509–512, 1999.
- [22] R. Ding, N. Ujang, H. B. Hamid, M. S. Abd Manan, R. Li, S. S. M. Albadareen, A. Nochian, and J. Wu, “Application of complex networks theory in urban traffic network researches,” *Networks and Spatial Economics*, vol. 19, no. 4, pp. 1281–1317, 2019.
- [23] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, “Spectral networks and locally connected networks on graphs,” *arXiv preprint arXiv:1312.6203*, 2013.
- [24] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” *arXiv preprint arXiv:1609.02907*, 2016.
- [25] F. M. Bianchi, D. Grattarola, L. Livi, and C. Alippi, “Graph neural networks with convolutional arma filters,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1–1, 2021.
- [26] J. Du, S. Zhang, G. Wu, J. M. F. Moura, and S. Kar, “Topology adaptive graph convolutional networks,” 2018.
- [27] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, “Graph attention networks,” *arXiv preprint arXiv:1710.10903*, 2017.
- [28] M. Li and Z. Zhu, “Spatial-temporal fusion graph neural networks for traffic flow forecasting,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 5, 2021, pp. 4189–4196.
- [29] A. Vaswani, “Attention is all you need,” *Advances in Neural Information Processing Systems*, 2017.
- [30] V. P. Dwivedi, C. K. Joshi, A. T. Luu, T. Laurent, Y. Bengio, and X. Bresson, “Benchmarking graph neural networks,” *Journal of Machine Learning Research*, vol. 24, no. 43, pp. 1–48, 2023.
- [31] Z. Zhou, Y. Wang, X. Xie, L. Chen, and C. Zhu, “Foresee urban sparse traffic accidents: A spatiotemporal multi-granularity perspective,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 8, pp. 3786–3799, 2022.
- [32] S. J. Prince, *Understanding Deep Learning*. The MIT Press, 2023.
- [33] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, “How powerful are graph neural networks?” *arXiv preprint arXiv:1810.00826*, 2018.
- [34] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, “Neural message passing for quantum chemistry,” in *International conference on machine learning*. PMLR, 2017, pp. 1263–1272.
- [35] T. Xie and J. C. Grossman, “Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties,” *Phys. Rev. Lett.*, vol. 120, p. 145301, Apr 2018.
- [36] G. Li, C. Xiong, A. Thabet, and B. Ghanem, “Deepergcn: All you need to train deeper gcns,” 2020.