# TAP: Traffic Accident Profiling via Multi-Task Spatio-Temporal Graph Representation Learning

ZHI LIU and YANG CHEN, Zhejiang University of Technology
FENG XIA, Federation University Australia
JIXIN BIAN, BING ZHU, GUOJIANG SHEN, and XIANGJIE KONG, Zhejiang University of Technology

56

Predicting traffic accidents can help traffic management departments respond to sudden traffic situations promptly, improve drivers' vigilance, and reduce losses caused by traffic accidents. However, the causality of traffic accidents is complex and difficult to analyze. Most existing traffic accident prediction methods do not consider the dynamic spatio-temporal correlation of traffic data, which leads to unsatisfactory prediction accuracy. To address this issue, we propose a multi-task learning framework (TAP) based on the Spatio-temporal Variational Graph Auto-Encoders (ST-VGAE) for traffic accident profiling. We firstly capture the dynamic spatio-temporal correlation of traffic conditions through a spatio-temporal graph convolutional encoder and embed it as a low-latitude vector. Then, we use a multi-task learning scheme to combine external factors to generate the traffic accident profiling. Furthermore, we propose a traffic accident profiling application framework based on edge computing. This method increases the speed of calculation by offloading the calculation of traffic accident profiling to edge nodes. Finally, the experimental results on real datasets demonstrate that TAP outperforms other state-of-the-art baselines.

CCS Concepts: • **Information systems** → *Data mining*; • **Human-centered computing** → *Empirical studies in ubiquitous and mobile computing;*

Additional Key Words and Phrases: Traffic accident profiling, spatio-temporal data, graph convolutional network, graph representation learning

**ACM Reference format:**
Zhi Liu, Yang Chen, Feng Xia, Jixin Bian, Bing Zhu, Guojiang Shen, and Xiangjie Kong. 2023. TAP: Traffic Accident Profiling via Multi-Task Spatio-Temporal Graph Representation Learning. *ACM Trans. Knowl. Discov. Data* 17, 4, Article 56 (February 2023), 25 pages.
https://doi.org/10.1145/3564594

This work is supported by the National Natural Science Foundation of China (62072409 and 62073295), Zhejiang Province Basic Public Welfare Research Project (LGG20F030008), Zhejiang Provincial Natural Science Foundation (LR21F020003), and Fundamental Research Funds for the Provincial Universities of Zhejiang (RF-B2020001).
Authors' addresses: Z. Liu, Y. Chen, J. Bian, B. Zhu, G. Shen, and X. Kong (corresponding author), College of Computer Science and Technology, Zhejiang University of Technology, 288 Liuhe Road, Hangzhou 310023, Zhejiang, China; emails: lzhi@zjut.edu.cn, {yangchen.cn, jixinbian, BingZhu97}@outlook.com, gjshen1975@zjut.edu.cn, xjkong@acm.org; F. Xia, Institute of Innovation, Science and Sustainability, Federation University Australia, PO Box 663, Ballarat, VIC 3353, Australia; email: f.xia@ieee.org.

1556-4681/2023/02-ART56 $15.00
https://doi.org/10.1145/3564594

## 1 INTRODUCTION

According to the **World Health Organization** (**WHO**, 2017) report, about 1.3 million people die every year in the traffic accident, and it on highways often cause more serious injuries. Statistics from the U.S. **National Highway Traffic Safety Administration** (**NHTSA**), the economic losses caused by motor vehicle crashes about 242 billion dollars, which is equivalent to 1.6% of the U.S. actual **gross domestic product** (**GDP**) in 2010. Adler et al. [1] proved that the continuous impact of accidents every minute causes a loss of 57 euros, which will cause higher losses on roads with high road occupancy. Traffic accident prediction can help people forecast the dangerous state of the road, thereby improve road safety, reduce economic losses caused by accidents, and reduce the number of fatal accidents.

The causes of accidents are diverse and the causality is complex. On one hand, some studies are devoted to mining the correlation between the related factors of traffic accidents (such as weather, driver attributes, traffic features, driving behavior, vehicle type, etc.) and the type or severity of traffic accidents [15, 30, 44]. The limitation of these methods is that they cannot directly predict traffic accidents. On the other hand, some studies are devoted to predicting when regional accidents will occur and predicting the number of accidents. The former regards accidents as a classification problem, and the latter regards accidents as a regression problem. Early, based on traditional machine learning methods such as Regression models [4], Bayesian networks [48], and **Decision Trees** (**DTs**) [29] were applied to traffic accident prediction. With the availability of traffic data and the development of deep learning methods such as **long short-term memory** (**LSTM**) network [36], **convolutional neural network** (**CNN**) [33], **Convolutional LSTM** (**ConvLSTM**) Network [49], and graph neural network [51] has attracted more and more attention. These methods do not have comprehensive traffic accident predictions, resulting in unsatisfactory practical applicability.

As we know, the accident is a special traffic event. Traffic data is not only temporal-related but also spatial-related. The spatio-temporal distribution of traffic flow results in different traffic states such as congestion and traffic accident [14]. Previous methods usually model the traffic network as grids or matrix cells [6, 32, 44]. These methods ignore the spatial heterogeneity of traffic. The Graph structure is a powerful mathematic model that can fit the traffic and is widely used in the field of traffic flow prediction. However, due to the complexity of traffic accidents, few studies are devoted to the analysis and prediction of accidents on the graph. Therefore, there is an urgent need for a traffic accident prediction method that captures the temporal and spatial dynamics of traffic data.

In response to the above problems, in our article, we proposed the spatio-temporal graph representation framework, which contains four layers, to generate the traffic accident profiling of the future time slice. Firstly, we collect and preprocess historical traffic data and historical traffic accident data of the area. Then, we construct a traffic spatio-temporal graph network based on the regional traffic network and historical traffic data. After that, we use these data as the input of the model to learn the spatio-temporal correlation of traffic data and embed it as the low-dimensional vector-traffic state. Finally, we generate traffic accident profiling through a multi-task scheme. In addition, we designed a traffic accident profiling application framework based on the **Internet of Things** (**IoT**). The experimental results under real traffic data show that our method has better performance than traditional methods and deep learning methods. In summary, our contributions are as follows:

- We propose a novel multi-task spatio-temporal graph representation learning framework (TAP) to generate the traffic accident profiling. In general, we embed traffic data as the traffic state, and then generate traffic accident profiling through a multi-task scheme.
- A novel **Spatio-temporal Variational Graph Auto-Encoders** (**ST-VGAE**) is designed to model the dynamic spatio-temporal correlation of traffic data. The spatio-temporal graph

convolution block in ST-VGAE captures the spatio-temporal correlation in the traffic data, and then obtains the traffic state through the encoder, and reduces the calculation time of the sub-tasks model.
— We design a framework to generate the traffic accident profiling and distribute it to drivers and traffic management departments through the IoT and edge computing, which provides a more feasible and appropriate method for studying traffic accidents in reality.
— We evaluate our method under real data, and the experimental results prove that our method is better than existing methods.

In the following, we first introduce the related work about traffic accident prediction and graph representation learning in Section 2. Then, we present some preliminary concepts and an overview of our framework TAP in Section 3. Furthermore, we introduce our method in detail in Section 4. Section 5 introduces our experimental setup and verifies the effectiveness of the proposed framework. Finally, we conclude our article in Section 6.

## 2 RELATED WORK

In addition to human factors such as drunkenness and drug abuse, we believe that the causes of accidents are mainly affected by two aspects, (1) spatial influence, that is, by the traffic of adjacent roads, and (2) temporal influence, that is, by the traffic of the period before the accident influences. In other words, we can analyze the causes of accidents by analyzing the spatial and temporal correlation of accident traffic data. Li et al. [26] first introduced the structure of graph to the field of transportation. Because most of the traffic network is a natural graph structure, recent work has shown that the GNN-based architecture can achieve better performance than the traditional CNN-based architecture, especially in terms of spatial correlation [45]. Hamilton et al. [12] pointed out that the central problem of graph representation learning is to find a way to integrate graph structure information into downstream models. Therefore, we review the related work of traffic accident prediction and graph representation learning.

### 2.1 Traffic Accident Prediction

For traffic accident prediction, we divide it into two categories:

**Statistics-based methods.** A traffic accident is usually regarded as a random event, containing various uncertain factors, and these factors are sometimes even conflicting. Therefore, use statistical models, such as Poisson or Negative binomial regression models to predict accidents. Most of these models rely on prior knowledge. Chang et al. [4] used a non-parametric tree-based model **Classification and Regression Tree** (**CART**) to establish the empirical relationship between traffic accidents and highway geometric variables, traffic features and environmental factors. Yu et al. [48] proposed a summary and analysis of single-vehicle accidents and multi-vehicle accidents were carried out. In the summary analysis, the hierarchical Poisson model and the Bayesian binary Poisson lognormal model and related random effects are used to simulate the collapse. In the classification analysis, real-time traffic data, weather information, and geometric features are combined, and a multi-level Bayesian **logistic regression** (**LR**) model is used to evaluate the real-time collision risk. Lin et al. [29] proposed a variable selection algorithm based on an FP tree for real-time traffic accident risk prediction in the context of traffic big data. Hu et al. [16] proposed a traffic outlier events detection method with missing data. A three-way tensor is used to encode the spatio-temporal relationships of traffic, and tensor decomposition and tensor completion models are established through the **alternate direction method of multipliers** (**ADMM**) the framework to detect traffic events and recover missing traffic data.

**Deep learning-based methods.** Due to the availability of traffic data and the effectiveness of deep learning algorithms, some deep learning methods have recently tried to solve the problem of accident prediction. Ren et al. [36] pointed out the temporal correlation features of traffic accidents, and predicted the accident based on LSTM combined with spatio-temporal data. Najjar et al. [33] used CNNs to learn from original satellite images to predict city-level road safety maps. Yuan et al. [49] believed that previous studies either ignored temporal information, or only used data from a small and homogeneous study area, and did not deal with the spatial heterogeneity and temporal correlation of traffic at the same time, and then proposed Hetero-ConvLSTM model, through the method of dividing regions, ConvLSTM uses a large amount of collected spatio-temporal heterogeneous data (such as weather, environment, road conditions, and traffic volume) to predict traffic accidents in different regions. Zhou et al. [51] proposed a framework RiskOracle that improves the prediction granularity to the minute level. The multi-task graph neural network (DTGN) captures the real-time changes of the traffic state and the high-level relationship between the dynamic partitions and the accident to predict the accident. In view of the sparse traffic accident data, which is easily affected by factors such as weather and POI, the complex spatio-temporal relationship of traffic accidents in different regions are difficult to model, the influence of factors such as weather and POI on the occurrence of traffic accidents. Most methods divide cities into grids to model, but this method ignores the spatial spread of traffic, resulting in unsatisfactory predictions.

### 2.2 Graph Representation Learning

In recent years, with the rise of graph structure in various fields (transportation, chemistry, social science, etc.), the core problem of graph representation learning is to encode the high-latitude, non-Euclidean information of the graph structure into the feature vector and as much as possible the topology information of the graph. The graph representation learning can be divided into two categories:

**Node embeddings-based method.** Perozzi et al. [35] proposed a method for learning the latent representation of vertices in the network (DeepWalk), which uses random walk to sample nodes to obtain local information. Tang et al. [39] proposed a network embedding model (LINE), which is designed to retain first- and second-order neighboring nodes as adjacent nodes and sample the nodes. However, these methods usually only perform a single representation of the graph structure and lack the representation of node features.

**Graph neural networks-based method.** Kipf et al. [22] based on CNN, through the first-order approximation of graph convolution, a semi-supervised classification method (GCN) is proposed on graph structure data to learn the representation of local graph structure and node features. Hamilton et al. [11] generated embeddings by designing a learning function (GraphSAGE) to sample and aggregate the features of the local neighbors of a node, without requiring all nodes to exist.

There have been some studies that have combined graph representation learning with transportation. Wang et al. [42] by modeling the driving state sequence into a driving state transition diagram, they proposed a framework based on **peer-to-peer and time-aware representation learning** (**PTARL**) to describe driving behaviors, based on which they scored driving behaviors and dangerous areas perform testing. Guo et al. [10] constructed the road network into a series of spatio-temporal graph sequences according to three temporal attributes (recent dependency, daily cycle dependency, and weekly cycle dependency), and captured by the spatio-temporal graph convolution method of the attention mechanism. The temporal and spatial correlation of traffic status changes, predict traffic. Song et al. [38] connected a single spatial graph of adjacent timesteps into a localized spatio-temporal graph and proposed a spatio-temporal synchronization graph convolution network to predict spatio-temporal traffic data. Shen et al. [37] combined taxi trajectory data, road network data, and POI data to propose a data-driven business district discovery framework, which uses GCNs to aggregate human mobility features based on geographic similarity.

## 3 OVERVIEW

### 3.1 Preliminary

This subsection briefly introduces the definition of traffic accident profiling problems.

*Definition 3.1 (Congestion Index).* For traffic data, it is difficult to directly obtain traffic congestion situations. The **congestion index** (**CI**) expresses the traffic congestion in an exponential way.

*Definition 3.2 (Rush Hour).* Rush hour is a period of time when there is a large amount of traffic in a day, and it is an important manifestation of traffic periodicity, including morning rush hour and evening rush hour.

*Definition 3.3 (Weather Event).* Extreme weather conditions have a significant influence on traffic. Weather event is a two-dimensional array $(type, severity)$, including the type and severity of the weather event.

*Definition 3.4 (Traffic Accident Warning).* Traffic accident warning is a scalar used to warn the probability of traffic accidents in the future.

*Definition 3.5 (Traffic Accident Classification).* We classify accidents into four categories according to the dissipating time: (1) minor accidents, (2) general accidents, (3) serious accidents, and (4) major accidents.

*Definition 3.6 (Traffic Accident Profiling).* Traffic accident profiling is an indicator to measure road safety in the future and is a two-dimensional array, which including the traffic accident warning and the traffic accident classification.

**Problem Definition.** We can define the problem as Given the $\boldsymbol{X}$, data recorded on the time period $T$ by all nodes on the traffic network, the purpose is to find a mapping function: $\boldsymbol{X} \rightarrow Z$, output a temporal-varying embedding vector $Z$, which we call traffic state. We regard this series of problems as a spatio-temporal graph representation learning task.

### 3.2 Framework

We propose a framework that uses spatio-temporal data to generate traffic accident profiling. It is mainly divided into five steps, data preparation, feature extraction, modeling spatio-temporal correlation, traffic accident profiling, and the application framework. Firstly, we collect and preprocess historical traffic data and historical traffic accident data provided by PeMS. Then, build a traffic spatio-temporal graph network based on the historical traffic accident data deployed by sensors in the traffic road network. After that, through ST-VGAE modeling dynamic spatio-temporal correlation to get the traffic state. Then, we generate traffic accident profiling through a multi-task scheme. Finally, we used an application framework based on edge computing to push the profiling information. More details are shown in Figure 1.

## 4 METHODOLOGY

In this section, we elaborate feature extraction, spatio-temporal graph representation learning, and application framework based on edge computing.

### 4.1 Data Preparation

To generate the traffic accident profiling, in this study, large-scale data related to traffic accidents were collected from Caltrans **Performance Measurement System** (**PeMS**). According to Chen et al. [7] PeMS is a centralized repository of all Caltrans real-time traffic data. It collects highway traffic data by collecting traffic data every 30 seconds, collecting traffic data every 5 minutes, and collecting Incident data from California Highway Patrol.

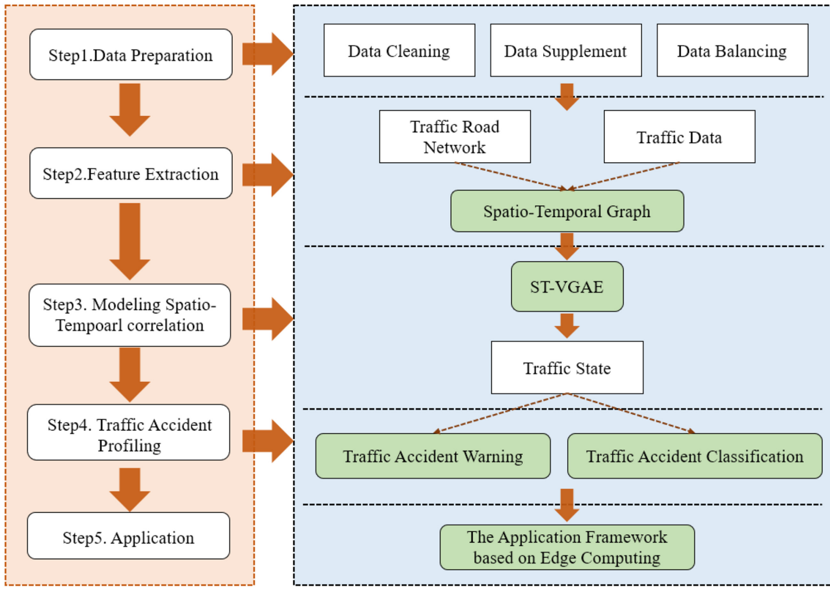


Fig. 1. Overview of the proposed framework.

We selected two areas, PeMS District4 (PeMSD4) and PeMS District8 (PeMSD8) as the research area. The traffic data and accident data are provided by PeMS, and the data are cleaned to remove some dirty and wrong data to ensure the integrity of the data. Besides, we add weather, the CI, and whether it is a Rush Hour to evaluate traffic. In this article, the calculation formula of the CI is defined as

$$CI = \begin{cases} \frac{speedlimit - acutalspeed}{speedlimit} & when\ CI > 0 \\ 0 & when\ CI \leq 0 \end{cases} \tag{1}$$

where *speedlimit* is 70 (mph), *acutalspeed* is the current speed, *Rush Hour* is defined as

$$Rush\ Hour = \begin{cases} 0 & when\ t_a\ in\ P \\ 1 & when\ t_a\ \text{not}\ in\ P \end{cases} \tag{2}$$

where $t_a$ is the time when the accident $a$ occurred, and $P$ represents morning rush hour and evening rush hour. In this article, it refers to 8 am to 9 am and 5 pm to 7 pm. Then, we undersample the accident data to balance the positive and negative samples. The process of data preparation is shown in Figure 2.

### 4.2 Feature Extraction

**Traffic Road Network.** We define the traffic road network as an undirected graph structure, $G = (V, E, A)$. $|V| = N$ is the total number of nodes in the graph. As shown in Figure 3, we regard sensors as nodes in the graph. If the road lacks sensors or the sensors are faulty, we use the data of adjacent sensors for interpolation. $E$ are the edges in the graph, representing the connections between nodes (sensors). $A \in \Re^{N \times N}$ is the adjacency matrix that contains the topological information of the transportation network, $A_{i,j} = 1$ means the node $i$ and node $j$ are connected, and 0 means disconnected.

**Traffic Data.** Each accident contains traffic features $(i.e., speed, occupancy, flow)$ and accident data $(i.e., time,\ location, description, duration)$. The sensors collect traffic data of the

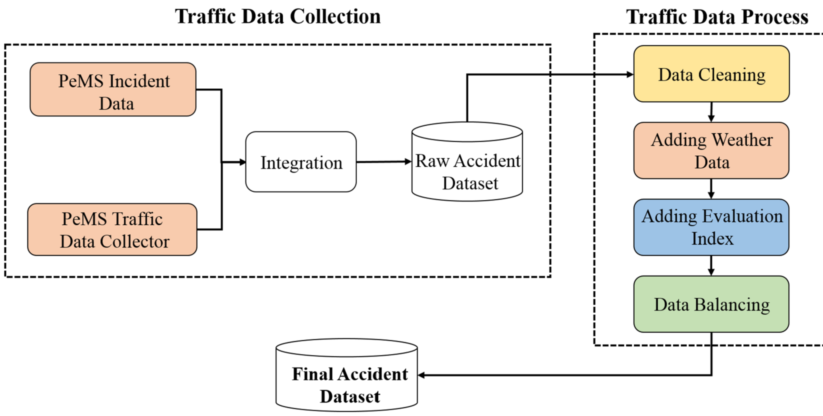


Fig. 2. Data preparation process.

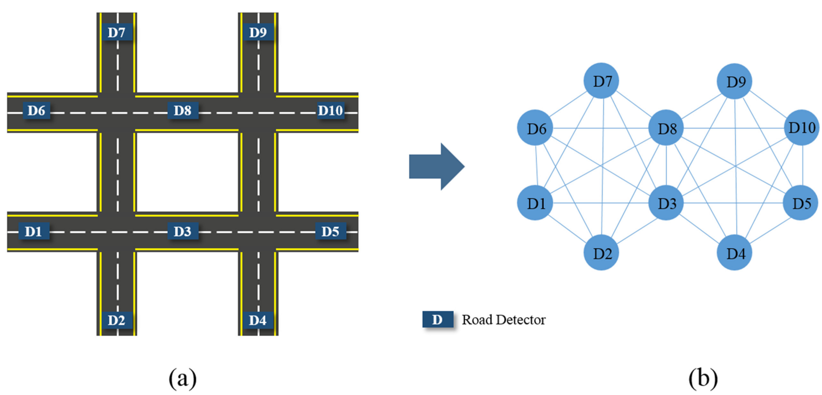


Fig. 3. The graph topology structure is constructed by traffic network sensors. (a) visualize the distribution of sensors in the traffic road network, (b) construct a graph by sensors. The connection of nodes represents the connection of roads, and what we construct is an undirected graph. If the road lacks sensors, or the sensors are faulty, we use the data of neighboring sensors for interpolation.

current state at regular intervals of time, including traffic data ($i.e.$, $speed$, $occupancy$, $flow$). $\boldsymbol{X} = (X_1, X_2, \ldots, X_T) \in \Re^{N \times F \times T}$ is all the features under the time $T$ period. $x_n^t$ is node $i$ features at time $t$. $X^t = (x_1^t, x_2^t, \ldots x_n^t) \in \Re^{N \times T}$ represents the features of all nodes at the time $t$. We collect the features of each accident two hours before the occurrence for research, that is, $T = 24$.

### 4.3 Modeling Spatio-Temporal Correlation

Some existing works have achieved good performance in modeling traffic spatio-temporal correlation. For example, Wang et al. [41] model periodic behavior in crowd flow by taking the deviation between previous and future periods. A lightweight spatial channel augmentation encoder is proposed, which augments standard CNNs to capture global spatial correlations and temporal dependencies to build more robust region representations. This approach makes the network more efficient and robust in long-term predictions. Liang et al. [28] use SENet to perform a local feature extraction module to learn representations for each region, and then combine global and sampled features to model spatio-temporal correlations in traffic. Finally, region-specific predictors based on tensor decomposition provide customized predictions for each region. This method excels on crowd flow analysis tasks. Huang et al. [17] encode latent semantic signals into regional representations and tracks the dynamic interdependencies between traffic accident data and external factors of urban events (such as POIs, emergencies, etc.) through a hierarchical fusion network. This method excels at modeling complex sequential transitions of traffic accidents occurrences. Huang et al. [18] divide the city into grids to model the spatio-temporal correlation of traffic accidents in terms of temporal, space, and event categories. An attention-based model is proposed to fuse per-view information and learn feature latent representations via LSTM to model dynamic patterns of city-wide anomalous events from the perspective of spatio-temporal classification. Liang et al. [27] divide cities into grids to model global spatial dependencies. Then, the grid space is transformed into the region space and the region correlations are inferred globally via message passing. Finally, the features are projected back into grid space and global perceptual features are obtained. This method can obtain the optimal model effect with fewer parameters in traffic flow prediction. Wang et al. [43] divide cities into coarse-grained and fine-grained grids, using channel-level CNNs and multi-view GCNs to capture local geographic dependencies and global semantic dependencies at different granularities. Furthermore, a feature fusion module is proposed to simulate the influence of external factors such as weather and points of interest on traffic accidents. This method more effectively predicts both fine-grained and coarse-grained citywide traffic accident risk. Inspired by the above papers, this article further considers the impact of dynamic spatio-temporal correlation in traffic flow on traffic accident prediction and proposes ST-VGAE.

**ST-VGAE.** The AutoEncoder is an unsupervised learning neural network model, which can learn the embedding vector of the input data (encode) and reconstruct the embedding vector into the original input data (decode). Variational Graph Auto-Encoders are based on AutoEncoders and learn the interpretable potential embedding of undirected graph structures from the perspective of data distribution through GCN [20, 21]. This article proposes a novel Variational Graph Auto-Encoders that uses spatio-temporal graph convolution blocks instead of GCN to represent complex traffic information. As shown in Figure 4, we stack multiple spatio-temporal convolution blocks, and each module includes a layer of spatial convolution and a layer of temporal convolution. Specifically, we first use the GCN model to model the spatial relationship and aggregate the traffic features in the spatial and then use the CNN model to model the temporal correlation and aggregate the traffic features in the time series. In addition, we added a residual network to each spatio-temporal convolution block to prevent overfitting.

**Encoder.** We collect 2 hours of data before accidents for embedding [32]. Given the adjacency matrix $A$ and the feature matrix $\boldsymbol{X}$, the spatio-temporal convolution block constructs a filter on the graph, captures node features through edge features, models the spatial correlation in traffic, and obtains a new embedding of the graph. To reduce the computational complexity, the Chebyshev polynomial approximate graph convolution can be rewritten as [13]:

$$g_\theta * x = \sum_{k=0}^{K} \theta_k T_k(\tilde{L})x, \tag{3}$$

$$\tilde{L} = \frac{2L}{\lambda_{\max}} - I, \tag{4}$$

where $g_\theta$ is the convolution kernel, $\theta$ is a vector of polynomial coefficients, $*$ is the graph convolution operation, $L = I - D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$ is the identity matrix, $D = \Sigma_j A_{ij}$ Representation degree matrix, $T_k$ is the Chebyshev polynomials of order $k$. After graph convolution models the spatial correlation of traffic by aggregating the neighboring information of each node of the graph, we use standard convolution to model the temporal correlation in traffic by aggregating features on

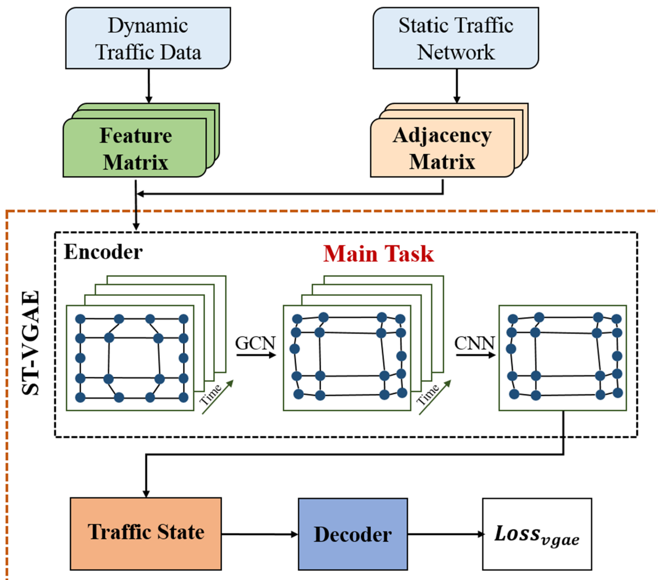


Fig. 4. The architecture of ST-VGAE.

adjacent time slices. The output of the first spatio-temporal convolution block can be defined as

$$h^l = \mathrm{ReLU}(\Phi * (\mathrm{ReLU}(w^T \cdot g_\theta * x) + b)), \tag{5}$$

where $h^l$ is the output of the $l$ spatio-temporal convolution block, $\Phi$ is the standard convolution kernel parameters, and $\Phi(*)$ is the standard convolution operation. Inspired by the work of [10], we designed a two-layer spatio-temporal graph convolution block. The second layer can be written as

$$h^{l+1} = \mathrm{ReLU}(\Phi * (\mathrm{ReLU}(w^T \cdot g_\theta * h^l) + b)). \tag{6}$$

We use the spatio-temporal graph convolution block to model the dynamic spatio-temporal correlation in traffic, and fit the distribution of the aggregated data through graph convolution, which could be denoted as

$$\mu = \mathrm{GCN}_u(X, A), \tag{7}$$

$$\log_\sigma = \mathrm{GCN}_\sigma(X, A), \tag{8}$$

where $\mu$ is the mean value of the distribution and $\sigma$ is the variance of the distribution. The traffic state $Z$ can be described as below:

$$Z \sim N(\mu, \sigma^2), \tag{9}$$

the encoder can also be denoted as follows:

$$q(Z|X, A) = \prod_{i=1}^{N} q(z_i|X, A), \tag{10}$$

where $q(z_i|X, A) = N(z_i|\mu_i, diag(\sigma^2))$.

**Decoder.** The decoder is defined as the inner product of the embedding vector and outputs the reconstructed adjacency matrix as shown below:

$$\hat{A} = Sigmoid(ZZ^T), \tag{11}$$

where $Z^T$ is the transposed matrix of $Z$, $\hat{A}$ is the adjacency matrix reconstructed by the decoder. The decoder can also be defined as

$$p(A|Z) = \prod_{i=1}^{N} \prod_{j=1}^{N} p(A_{i,j}|z_i, z_j), \tag{12}$$

where $p(A_{i,j} = 1|z_i, z_j) = Sigmoid(ZZ^T)$. The loss function of ST-VGAE is divided into two parts. The first part is to calculate the binary cross entropy between the original adjacency matrix $A$ and the reconstructed adjacency matrix $\hat{A}$. The second part is to calculate the KL-divergence between $q(Z|X, A)$ and $p(Z)$, as shown below:

$$L_{vgae} = \mathrm{E}_{q(Z|X,A)}[logp(A|Z)] - KL[q(Z|X, A)||p(Z)]. \tag{13}$$

**Traffic Inference with Normalizing Flows.** The posterior approximation of variational inference is usually a simple Gaussian distribution, which is difficult to fit complex traffic flow conditions. Normalizing flows push a simple density through a series of transformations to produce a richer multi-modal distribution-like a fluid flowing through a set of tubes [34]. Inspired by [34, 40, 50], to improve the posterior distribution in ST-VGAE, we introduced standardization to transform the simple Gaussian distribution into a more accurate traffic distribution and proposed TAP*, which is based on the improved ST-VGAE*.

The target vector $z_k$ is a latent vector $z_0$ sampled from a simple distribution such as the Gaussian distribution, which is obtained by a series of $k$ changes $f_k$:

$$z_k = f_k(f_{k-1}(\cdots f_1(z_0))). \tag{14}$$

$f(\cdot) = f_k \circ f_{k-1} \circ \cdots \circ f_1(z_0)$ is called flow, which is composed of a series of invertible functions and differentiable functions, $z_0 = f^{-1}(z_k)$, the process of forming the target distribution $q_k$ is called normalized flow. Given a set of variables $z = [z_0 \cdots z_n]$, the probability distribution of random variable $z_n$ is

$$q_k(z_n) = q_0(z_0) \left|\det \frac{dz_0}{dz_n}\right| = q_0(z_0) \left|\det \frac{\partial f^{-1}(z_n)}{\partial z_n}\right| = q_0(z_0) \prod_{k=0}^{k} \left|\det \frac{\partial f^{-1}(z_k)}{\partial z_k}\right|^{-1}. \tag{15}$$

$z_0 = f^{-1}(z_k)$ brings into the formula, det $\frac{\partial f^{-1}(z_n)}{\partial z_n} = J_{f^{-1}}(z_n)$ is the Jacobian determinant, because of the nature of the determinant, $\det J_{f^{-1}}(z_n) = \det J_f(z_0)^{-1}$. Inspired by [34], we use plane flow parameterization to approximate the posterior $q_k$:

$$f_i(z) = z + u_i h(w_i^{\mathrm{T}} z + b_i), \tag{16}$$

where $u_i, w_i \in R^d$ is the learnable parameter, $d$ is the characteristic dimension of the characterizing vector, and $h(\cdot)$ is the smooth element-wise non-linear activation function tanh. Therefore, the Jacobian determinant is expressed as

$$\left|\det \frac{\partial f}{\partial z}\right| = \left|\det(\mathrm{I} + u\left[h'(w^{\mathrm{T}} z + b)w\right]^{\mathrm{T}}\right| = \left|1 + u^{\mathrm{T}} h'(w^{\mathrm{T}} z + b)w\right|, \tag{17}$$

where $h'$ is the derivative of $h$ and can be calculated in $O(d)$ time complexity. We modify the ELBO of variational inference as

$$L_{vage*} = \mathrm{E}_{q_0(z_0)}[\ln q_0(z_0)] - \mathrm{E}_{q_0(z_0)}[\log p(A, z_k)] - \mathrm{E}_{q_0(z_0)}\left[\sum_{k=1}^{K} \ln \left|1 + u^{\mathrm{T}} h'(w^{\mathrm{T}} z + b)w\right|\right]. \tag{18}$$

Algorithm 1 detailed introduces Modeling Spatio-Temporal correlation process.

**ALGORITHM 1:** Modeling Spatio-Temporal Correlation

**Input**: Adjacency matrix $A$, Features matrix $\boldsymbol{X}$;
**Output**: Traffic state $Z$;
Initialize the GCNs and CNNs parameters
Initialize $Z = \emptyset, \hat{A} = \emptyset$
Initialize $\mu = \emptyset, \sigma = \emptyset$
**for** $\{A, \boldsymbol{X}\}$ *in* $\{Traffic\ Data\}$ **do**
  initialize *spatio_conv* and *temporal_conv*
  initialize $h^l = \boldsymbol{X}$
  **for** *block in* $\{spatio - temporal\ blocks\}$ **do**
    $spatio\_conv$ = GCNs($A$, spatio_conv)
    $temporal\_conv$ = CNNs($h^l$)
    $h^l$ = ResNet($temporal\_conv$, $\boldsymbol{X}$)
  **end**
  $\mu = \text{GCN}_u(X, A)$
  $\log_\sigma = \text{GCN}_\sigma(X, A)$
  $Z = N(\mu, \sigma^2)$
  **if** $TAP^*$ **then**
    *update* $Z$ *through Normalizing Flows.*
  **end**
  $\hat{A} = \text{Sigmoid}(ZZ^T)$
**end**

### 4.4 Traffic Accident Profiling

User profiles are a typical application of feature engineering. Common features are extracted through data mining and analysis of various types of data. The essence of user profiles is to accurately describe any object. Drawing on the idea of user profiles, this article uses graph representation learning, mining the hidden variables profiling of accidents in the road network, and combining the displayed variables to construct accident profiling. Specifically, our profiling includes the probability and level of accidents occurring in the future. Inspired by [25], we applied the idea of cascade network to traffic accident profiling, and established the model structure shown in Figure 5.

**Traffic Accident Warning.** As Chen et al. [8] said, it is difficult for us to directly predict whether a traffic accident will occur, because the factors that affect traffic accidents are usually related to people, such as driver distraction, driver drinking, or drug use, these factors often unobservable and difficult to collect. Therefore, we try to predict the risk of an accident, that is, the probability of an accident in the future (5 minutes, 30 minutes, 60 minutes, etc.). The method we propose can learn the embedding of each accident, and predict the accident risk of the future time slice. In this article, we collect the data of each accident two hours before the occurrence for research, namely $T = 24$. In order to use the embedding vector for the traffic accident warning, we designed a simple application-layer externally to learn the mapping of the embedding vector to the accident risk. Essentially, the application layer is an FC layer. We input the embedding vector into the application layer to predict the traffic accident risk for the next time slice, as shown below:

$$\tilde{y} = Sigmoid(w^T \cdot Z + b), \tag{19}$$

where $\tilde{y}$ is the probability of an accident. Therefore, the loss function of accident warning is

$$Loss_{warn} = Loss_{vgae} + \lambda_1 BCE(y, \tilde{y}). \tag{20}$$

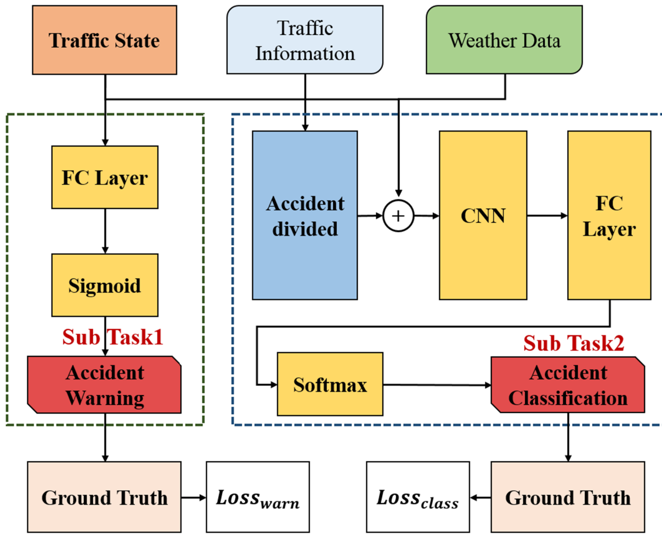


Fig. 5. The flow chart of multi-task scheme.

**Traffic Accident Classification.** Foreseeing the severity of the accident will help government departments to take appropriate and timely measures. Therefore, we studied the severity of the accident and the spatio-temporal dynamics of regional traffic to predict the severity of accidents. Specifically, we first classify accidents into four categories according to the dissipating time: (1) minor accidents, (2) general accidents, (3) serious accidents, and (4) major accidents. After getting the traffic state through the model, we combine the current time CI and peak time indicators to predict the accident level. Since the traffic accident classification does not update the main model parameters through backpropagation, we added a layer of CNN to obtain a better model effect, as shown below:

$$\hat{Z} = Z \oplus T_{data}, \tag{21}$$

$$\hat{y} = Softmax(\Phi * (w^T \cdot \hat{Z} + b)), \tag{22}$$

where $\hat{Z}$ is the vector after the traffic state and traffic data are aggregated, $\oplus$ is the aggregation operation, $\hat{y}$ is the accident level, $\Phi$ is the standard convolution kernel parameters, and $\Phi\bullet$ is the one-dimensional convolution operation. Therefore, the loss function of accident classification is

$$Loss_{class} = Loss_{vgae} + \lambda_2 CE(y, \hat{y}). \tag{23}$$

Algorithm 2 detailed introduces the traffic accident profiling process.

## 4.5 Application

One of the main causes of casualties in traffic accidents is the lack of timely rescue. Whether the ambulance arrives at the scene in time determines the survival rate of the wounded [2]. Since highways are generally far away from urban areas, it is very important to rescue accidents in advance [9]. In the past 20 years, the IoT technology has developed in many industries such as smart transportation, smart industry, and smart health, making cities smarter and playing an important role in traffic emergency management [24]. Edge computing offloads the storage, calculation, and propagation that previously relied on cloud servers to **Road Side Units** (**RSU**) or **Base Stations** (**BS**), reducing transmission delays through distributed parties and improving the real-time nature

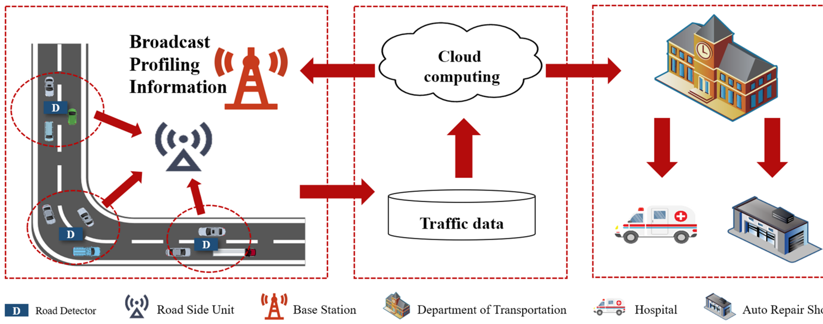


Fig. 6. The application of the traffic accident profiling.

**ALGORITHM 2:** Traffic Accident Profiling

**Input**: Traffic data $T_{data}$;
**Output**: Traffic state $Z$, the traffic accident profiling $T_{profiling}$;
Initialize the GCNs and CNNs parameters
Initialize $T_{warn} = \varnothing$, $T_{class} = \varnothing$
**for** $\{Z, T_{data}\}$ *in* {*Traffic Data*} **do**
    $T_{warn}$ = FC($Z$)
    Back propagation update the parameters of ST-VGAE and FC layers.
    **if** *the label of embedding vector* $T_{label}$ == 1 **then**
        Concentrate $Z$ and $T_{data}$ as $Variable$
        $Variable$ = CNNs($Variable$)
        $T_{class}$ = FC($Variable$)
        Back propagation update the parameters of CNNs and FC layers.
    **end**
**end**
$T_{profiling}$ is composed of $T_{warn}$ and $T_{class}$.

of information [23]. Therefore, we propose a traffic accident profiling framework based on edge computing as shown in Figure 6, which is inspired by [3] distributed architectures. Specifically, we firstly collect and upload the traffic data (such as speed, capacity, occupancy, etc.) of the loop detector through RSU/BS. On the cloud server, ST-VGAE embeds the traffic data and returns it, and then RSU/BS obtains the traffic accident profiling through edge computing. Finally, we regularly broadcast profiling information through the push message dissemination mechanism through IoT [9]. If the level of the accident is relatively low, only notify the driver and the auto repair shop, otherwise, send the relevant information about the location of the accident to the transportation department, hospital, and auto repair shop.

Algorithm 3 detailed introduces the traffic accident profiling application based on the edge computing process.

## 5 EXPERIMENTS

In this section, we first introduce the dataset and experimental settings. Then, we conducted extensive experiments to evaluate the effectiveness of our proposed framework.

### 5.1 Data Description

The experimental data includes traffic data, accident data, and weather data. More details are shown in the Table 1.

Table 1. Variable Description

| Datasets | Variables | Desciption |
|---|---|---|
| Traffic Data | Traffic_Flow | Sum of flows over the 5-minute period across all lanes. |
| | Avg_Speed | Average occupancy across all lanes over the 5-minute period. |
| | Avg_ Occupancy | Flow-weighted average speed over the 5-minute period across all lanes. |
| | CongestionIndex | Measure Lane congestion index. |
| Accident Data | Time | Date and time of the accident. |
| | Location | The accident occurred place. |
| | Description | A brief description of the accident. |
| | Rush_Hour | Whether it is the rush hour in the morning or evening |
| | Duration | The duration time of the accident. |
| Weather Data | Type | The type of an event. |
| | Severity | The severity of an event. |

**ALGORITHM 3:** Application Based on Edge Computing

```
Initialize traffic data T_data;
Initialize embedding vector Z,
Initialize the traffic accident profiling T_profiling;
for timestamp = 1, 2, ... N do
    Initial traffic data s
    RSU/BS collect T_data aggregation it into s and send it to cloud
end
for s_0, s_1 ... s_n in s do
    The cloud server calculate Z through ST-VGAE and return it
    BS/RSU calculate T_profiling and push this information at a fixed frequency
end
```

**Traffic Data.** The traffic data comes from the two regions of California in America provided by PeMS. PeMSD4 is the traffic data of the San Francisco Bay Area in January 2018. PeMSD8 is the traffic data of San Bernardino Area from July to August 2016. PeMSD4 screened 307 sensors from 3,848 sensors on 29 roads, and PeMSD8 screened 170 sensors from 1,979 sensors on 8 roads [10]. In the experiment, the three features of total traffic flow, average speed, average occupancy, and the CI indices were used.

**Accident Data.** The experiment used PeMS to provide accident data, including accident time, duration, and so on. According to the time of the accident, we can calculate the rush hour index. Among them, District 4 had a total of 2,551 accidents in January 2018. A total of 1,853 incidents occurred in District 8 from July to August 2016. It is worth noting that we have cleaned the accident data. We believe that accidents that last less than 15 minutes will hardly have a significant impact on regional traffic, so we have eliminated this part of the data. In addition, when the time and location of the two accidents are very close, we consider them to be cascading accidents and treat them as the same positive sample.

**Weather Data.** We used the weather event datasets of PeMSD4 in January 2018 and PeMSD8 in July and August 2016 to study the impact of weather events on traffic accidents. Specifically, the weather event dataset includes the type of weather event (such as fog, hail, rain, etc.) and the severity of the weather event (light, moderate, severe) [31].

### 5.2 Experiment Settings

**Implementation Details.** We first balance the data so that the positive and negative samples are close to 1:1. Then, we divide the dataset into 60%, 20%, and 20%, respectively, as the training dataset, validation dataset, and test dataset. We set the weight of the loss function $\lambda_1 = 0.8, \lambda_2 = 0.8$.

The TAP was optimized by backpropagation. For all models, we set the learning rate to 0.001 and apply the Adam optimizer to optimize the model. In the process of training accident warning, we select the model with the smallest loss each time to save and update the ST-VGAE parameters through back propagation. We do not update the ST-VGAE parameters when training the accident classification auxiliary task and obtain better results through the CNN. In the test, we extract the required data and sent it to the model, and output the traffic accident profiling through the main task and auxiliary tasks. In addition, we use the label smoothing method to avoid the polarization of accident warning results.

**Evaluation Metrics.** To be more intuitive and realistic, we set thresholds to classify traffic accident warnings into positive and negative samples and use classification metrics to verify the performance of the model. Inspired by [47], in this article, we choose the threshold as 0.5, if the road with traffic accident warning higher than 0.5 is regarded as a positive sample, otherwise it is regarded as a negative sample. We use Precision, Recall, F1-Score, Accurary@, AUC@ to measure the performance of different methods. Among them, Precision, Recall, F1-Score are used to measure the effect of accident warning sub-task, and Accuracy@, AUC@ are used to measure accident classification sub-task:

- **Precision.** Calculate the proportion of correctly predicted samples among the positive samples.
- **Recall.** Calculate the proportion of correct prediction positive samples in the total number of positive samples.
- **F1-Score.** Fully measured precision and recall.
- **Accuracy.** Calculate the proportion of correct samples among various samples.
- **AUC.** Evaluating the performance of a classification model reflects the model's ability to distinguish between different categories.

**Embedding feature dimension analysis.** To investigate the effect of different embedding dimensions on model performance, we conduct experiments on different embedding dimensions. Figure 7(a) and (b) shows that in the areas of PeMSD4 and PeMSD8, the model performance of the embedded dimension 4 in the accident warning subtask is better than other dimensions. It can be seen from Figure 7(c) and (d) that in the traffic accident classification subtask, when the embedding size is 1, PeMSD4 and PeMSD8 have the best performance, and when the embedding dimension is 16, the model performance is the worst. We believe that both too large and too small embedding dimensions will cause model performance degradation. If the embedding size is too small, it will be difficult for the model to completely preserve the traffic state. Models with too large embedding dimensions will have redundant traffic states. Considering the model effect, parameter amount, and training time, the embedding dimension of PeMSD4 is set to 4, and the embedding size of PeMSD8 is set to 1.

**Time-Slice length analysis.** To study the effect of the length of time-slice on model performance, we conducted experiments on the different lengths of time-slice. The scale of the abscissa represents time-slices, and each time-slice is 5 minutes. As shown in Figure 8, as the length of time increases, the effect of the model first increases and then decreases. Figure 8(a) and (b) shows that in the accident warning subtask in the PeMSD4 area, the 30-minute model performance is higher than other time-slice, and the worst result occurs in 60-minute, as is the PeMSD8 area. Figure 8(c) and (d) shows PeMSD4 and PeMSD8, the 30-minute accident classification subtask is the best. We

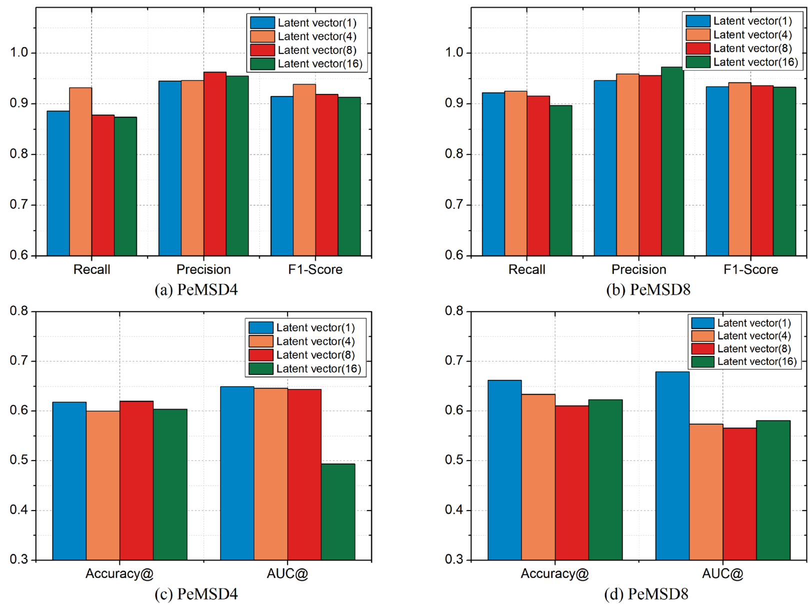


Fig. 7. Comparison of different dimensions.

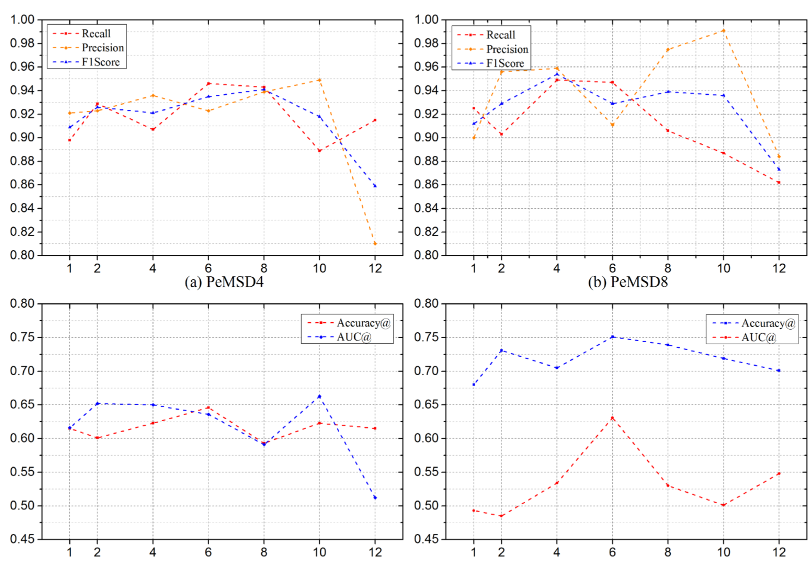


Fig. 8. Comparison of different time-slice.

believe that too fine or too coarse time granularity will reduce the model's ability to capture the features of traffic conditions, resulting in a slight decrease in performance. Comprehensive consideration, we set the accident profiling time-slice to 30-minute.

**Weight Coefficient Analysis.** We analyzed the impact of multi-task weight coefficients on traffic accidents. We hope that traffic accident warning can be as accurate as possible, so subtask 1

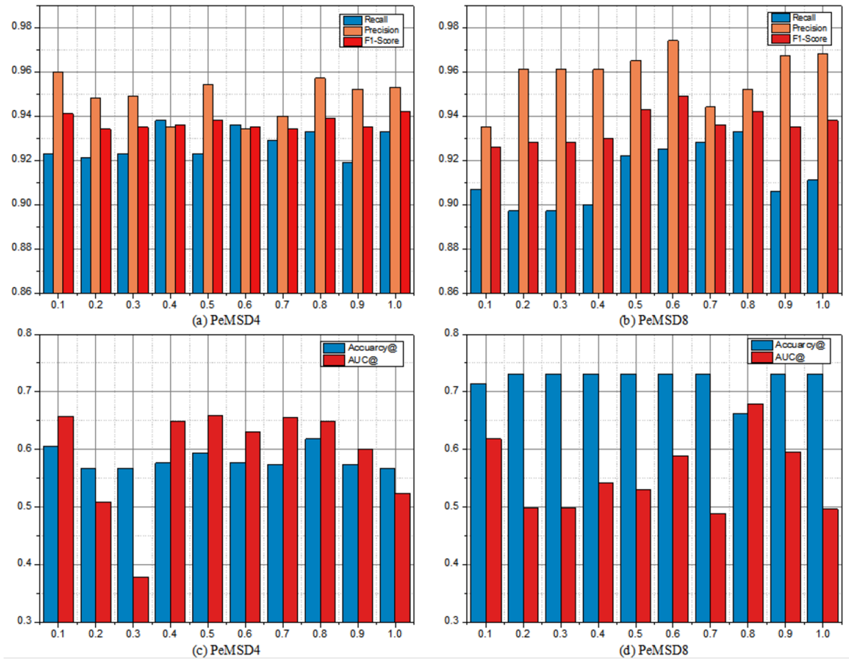


Fig. 9. Comparison of weight coefficient.

will update the model parameters of the main task through backpropagation, and we do not want traffic accident classification to learn from the backpropagation interference model (we confirm in the ablation study). So we only experiment with $\lambda_1$, and the range of $\lambda_1$ is between 0.1 and 1. As shown in Figure 9, we can see that on our dataset, the model performs most stable when the $\lambda_1$ is 0.8. This is because too large will cause the model to be more inclined to optimize sub-tasks, and too small will lose the meaning of multi-task learning.

**Weather Analysis.** We analyzed the impact of weather on traffic accidents. As shown in Figure 10, we can see that on the PeMSD4 dataset, the number of traffic accidents related to light weather events is the largest, and the number of minor accidents is also the largest, while the number of accidents related to moderate and severe weather events is relatively small. This is because the severe weather events themselves are sparse, and on the other hand, people try to reduce the number of trips and be more cautious when encountering severe weather events. On the PeMSD8 dataset, the traffic data related to moderate weather events is the largest. This is because the number of moderate weather events (mainly fog) is far more than other weather events.

**Analysis of experimental results**

**Baselines.** We have studied several common machine learning algorithms (LR, **Support Vector Machines** (**SVM**), DT) and some of the latest deep learning algorithms: LSTM, GRU [19], these two models are processing time-series features. In terms of performance, it is a special RNN model. ConvLSTM [49] adds convolution operation based on LSTM, which performs well in processing time-series data. **Stack Denoising Autoencoder** (**SDAE**) [8] stacks multiple layers of Denoising Autoencoder to show good performance in unsupervised learning. **Spatio-Temporal Graph Convolutional Networks** (**STGCN**) [46] performs well in traffic forecasting. A novel **Stack Denoise Convolutional Auto-Encoder** (**SDCAE**) [5] to predict the risk of traffic accidents at the city

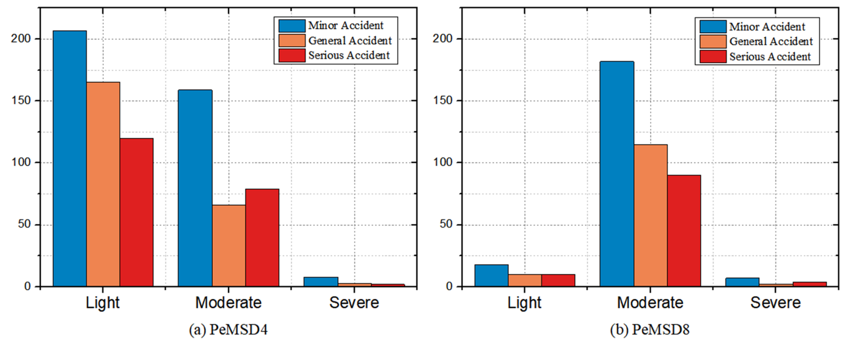


Fig. 10. Comparison of weather condition.

level. The **Deep Spatio-Temporal Graph Convolutional Network** (**DSTGCN**) [47] proposes a graph-based spatio-temporal method to predict the risk of future traffic accidents.

The classical methods (LR, SVM, DT) and some neural network methods (LSTM, GRU, ConvLSTM, SDAE) cannot directly input the graph structure, so we do not retain the graph topology and convert the graph into a sequence as the input of the model. We chose machine learning algorithms because we wanted to use a small number of datasets to obtain excellent results. For a fair comparison, we first initialize the baseline's hyperparameters from the original literature and then fine-tune them to achieve the best performance on our dataset. The detailed hyperparameter settings for the baseline are as follows: For LSTM and GRU, we set the number of features in the hidden state to 32 and the number of stacked layers to 2. For ConvLSTM, we set the kernel size to $3 \times 3$, the number of LSTM layers stacked on each other to 2, and the number of features in the hidden state to 16. For SDAE, we stack three denoising autoencoder layers and set the number of units in each layer to 40, 40, 40. For SDCAE, we stack three DCAE blocks, each DCAE block contains a convolutional layer with a kernel size of $2 \times 1$. The output channels of these three convolutional layers are 6, 9, and 15, respectively. For STGCN, we stack two ST-Conv Blocks with 64, 16, and 64 channels for three layers in each ST-Conv Block. For DSTGCN, we specify the number of spatial and temporal convolution blocks as 3. Furthermore, we set the hidden dimensions of spatial, temporal, and external features to 10, 20, and 10, respectively. For the hyperparameter tuning strategy, we perform a grid search strategy (same as [46, 47]) to locate the best parameters in validation. We do not use machine learning as a comparative experiment for accident classification, because machine learning often trains multiple classifiers when processing multiple classifications, which cannot be used in practice. The experimental results are shown in Tables 2 and 3.

From the results, it can be concluded as follows: First of all, the performance of the DT model is better than other classic machine learning, because DT can better learn features related to traffic accidents. Secondly, LSTM and GRU process time series information through recurrent neural networks, and their performance is close. ConvLSTM uses a convolutional network on the basis of a recurrent neural network, which shows better performance than LSTM and GRU. SDAE cannot capture the temporal and spatial correlation in traffic, so the effect is similar to other deep learning models. SDCAE uses convolution instead of full connection on SDAE to obtain better prediction results. DSTGCN not only considers spatial correlation but also considers the temporal correlation of heterogeneous data, so it performs better than other baseline models. TAP* introduces a generative flow-based method to construct a more accurate distribution of traffic data to solve the pain points of variational inference, so it achieves the best results on the dataset. Thirdly, the deep learning method achieves better performance than the traditional machine learning model, which proves

Table 2. Comparison Results of Various Methods on PeMSD4

| Model | PeMSD4 | | | | |
|---|---|---|---|---|---|
| | Recall | Precision | F1-Score | Acc@ | AUC@ |
| LR | 0.745 | 0.854 | 0.796 | – | – |
| SVM | 0.719 | 0.811 | 0.761 | – | – |
| DT | 0.785 | 0.842 | 0.813 | – | – |
| LSTM | 0.875 | 0.913 | 0.894 | 0.47 | 0.594 |
| GRU | 0.924 | 0.871 | 0.894 | 0.475 | 0.616 |
| ConvLSTM | 0.919 | 0.913 | 0.915 | 0.493 | 0.558 |
| SDAE | 0.879 | 0.942 | 0.909 | 0.473 | 0.538 |
| SDCAE | 0.891 | 0.924 | 0.915 | 0.503 | 0.565 |
| STGCN | 0.895 | 0.902 | 0.899 | 0.592 | 0.471 |
| DSTGCN | 0.901 | 0.917 | 0.909 | 0.615 | 0.616 |
| TAP (ours) | **0.932** | **0.946** | **0.939** | **0.618** | **0.649** |
| TAP* (ours) | **0.929** | **0.954** | **0.941** | **0.600** | **0.640** |

Table 3. Comparison Results of Various Methods on PeMSD8

| Model | PeMSD8 | | | | |
|---|---|---|---|---|---|
| | Recall | Precision | F1-Score | Acc@ | AUC@ |
| LR | 0.768 | 0.849 | 0.807 | – | – |
| SVM | 0.706 | 0.906 | 0.793 | – | – |
| DT | 0.806 | 0.818 | 0.812 | – | – |
| LSTM | 0.878 | 0.896 | 0.887 | 0.644 | 0.581 |
| GRU | 0.895 | 0.902 | 0.898 | 0.649 | 0.608 |
| ConvLSTM | 0.871 | 0.942 | 0.902 | 0.634 | 0.472 |
| SDAE | 0.88 | 0.91 | 0.895 | 0.624 | 0.494 |
| SDCAE | 0.898 | 0.925 | 0.911 | 0.606 | 0.595 |
| STGCN | 0.797 | 0.929 | 0.858 | 0.615 | 0.616 |
| DSTGCN | 0.910 | 0.921 | 0.915 | 0.597 | 0.612 |
| TAP (ours) | **0.925** | **0.959** | **0.942** | **0.662** | **0.679** |
| TAP* (ours) | **0.931** | **0.957** | **0.944** | **0.724** | **0.621** |

that the deep learning model has a stronger ability to model complex traffic relations. Finally, we can see that our framework is superior to other methods in most evaluation indicators. We believe that there are two main reasons: Firstly, we designed ST-VGAE to capture spatio-temporal correlations through multiple spatio-temporal convolution blocks. Secondly, ST-VGAE uses an

Table 4. Ablation Study on PeMSD4

| Model | PeMSD4 | | | | | | |
|---|---|---|---|---|---|---|---|
| | Recall | Precision | F1-Score | Acc@ | AUC@ | Time (s) | Epoch |
| TAP | **0.932** | **0.946** | **0.939** | **0.618** | **0.649** | **75** | **33** |
| w/o weather | 0.923 | 0.949 | 0.936 | 0.597 | 0.644 | 78 | 42 |
| w/o ti | 0.856 | 0.892 | 0.873 | 0.593 | 0.593 | 77 | 44 |
| sub1 w/o bp | 0.857 | 0.915 | 0.885 | 0.595 | 0.658 | – | – |
| sub2 w/o cnn | – | – | – | 0.597 | 0.622 | – | – |
| sub2 bp | 0.913 | 0.945 | 0.929 | 0.594 | 0.637 | 112 | 57 |

Table 5. Ablation Study on PeMSD8

| Model | PeMSD8 | | | | | | |
|---|---|---|---|---|---|---|---|
| | Recall | Precision | F1-Score | Acc@ | AUC@ | Time (s) | Epoch |
| TAP | **0.932** | **0.946** | **0.939** | **0.618** | **0.649** | **79** | **38** |
| w/o weather | 0.903 | 0.953 | 0.927 | 0.634 | 0.566 | 75 | 39 |
| w/o ti | 0.905 | 0.921 | 0.912 | 0.615 | 0.515 | 73 | 41 |
| sub1 w/o bp | 0.822 | 0.885 | 0.852 | 0.624 | 0.508 | – | – |
| sub2 w/o cnn | – | – | – | 0.609 | 0.594 | – | – |
| sub2 bp | 0.922 | 0.949 | 0.935 | 0.632 | 0.523 | 92 | 55 |

embedding layer to abstract the accident instead of directly using the data for downstream models, which makes our model more robust.

**Ablation Study.** In order to study the influence of different features and components on the model's performance, we conduct ablation experiments on two datasets. We named the variants of TAP as follows:

— **w/o weather:** This is TAP, which removes weather event features.
— **w/o ti:** This is TAP, which removes traffic indicators, including CI, peak hours, and weather events.
— **sub1 w/o bp:** This is the subtask 1 traffic accident warning that does not update the main task parameters through backpropagation.
— **sub2 w/o cnn:** This is subtask 2 traffic accident classification to remove the CNN model.
— **sub2 bp:** This is the subtask 2 traffic accident classification back propagation update the main task ST-VGAE model parameters.

We repeat each experiment five times and report the average values of Recall, Precision, and F1-Score of accident warning on the test dataset and the average values of Acc@ and AUC@ of accident classification in Tables 4 and 5. The introduction of weather events and traffic indicators (w/o weather and w/o ti) improves the effectiveness and efficiency of the model because these indicators have a significant impact on traffic travel. The ablation experiment in which traffic accident warning (subtask 1) does not update the main task model parameters through backpropagation proves that our model framework can improve the experimental effect of specific tasks. To achieve better

Table 6. Model Complexity Analysis

| Models | PeMSD4 | | PeMSD8 | |
|---|---|---|---|---|
| | Training Time (s) | Training Epoch | Training Time (s) | Training Epoch |
| LSTM | 78 | About 35 | 73 | About 32 |
| GRU | 76 | About 34 | 72 | About 32 |
| ConvLSTM | 52 | About 30 | 49 | About 28 |
| SDAE | 101 | About 39 | 98 | About 39 |
| SDCAE | 83 | About 39 | 98 | About 39 |
| STGCN | 85 | About 41 | 81 | About 40 |
| DSTGCN | 81 | About 46 | 79 | About 43 |
| **TAP** | **79** | **About 38** | **75** | **About 33** |
| **TAP*** | **100** | **About 38** | **89** | **About 33** |

results in accident classification, we used a layer of CNN to better model time series features (w/o cnn). In addition, we also conduct ablation experiments on the training strategy of the multi-task model and back-propagate the accident classification to update the model parameters (sub2 bp) of the main task (ST-VGAE). The increase of the model parameters in a single training reduces the efficiency of the model.

In summary, weather events and traffic indicators are essential to the performance of TAP. CNN helps accident classification improve model performance. The multi-task training strategy improves the efficiency of the model during training.

### 5.3 Traffic Accident Profiling

Considering the usability of the model, we analyzed the cost time of each epoch and training epochs of each deep learning model. The experiment was conducted on an Ubuntu machine equipped with 4 Inter(R) Xeon(R) Sliver 4110 CPU @2.10GHz with 8 physical cores, and the GPU is NVIDIA Quadro M4000 equipped with 8G of memory. Machine learning methods have fewer parameters and shorter training time, thus we not compare them. The comparison results are shown in Table 6. Firstly, ConvLSTM combines CNN and LSTM has the shortest training time. Secondly, SDAE learns the hidden embedding of features by optimizing multiple stacked denoising autoencoders with the longest training time and epoch, while SDCAE uses CNN instead of full connection to improve the computational efficiency of the model. Finally, our TAP model uses ST-VGAE to embed spatio-temporal traffic data to reduce the dimensionality of the traffic data and then uses a multi-task scheme to generate traffic accident profiling, so the model complexity is better than STGCN and DSTGCN. TAP* optimizes the effect of the model through normalizing flows, but it also increases the complexity of the model. In practical applications, we deploy the main task model in the cloud, and offload the subtasks that generate profiling to RSU or BS. In this way, we can reduce the calculation time of the model, so that the profiling information is transmitted to the driver and management department more quickly.

We select real data and generate accident profiling through our model. Figure 11 shows our profiling information. We assume that the BS has strong storage and computing capabilities, and ignores the delay in message propagation. In the traffic accident profiling, the intensity of the color indicates the level of accidents in the area, and the risk is the probability of accidents. We found that the probability of an accident and the level of the accident are not linearly related, because the

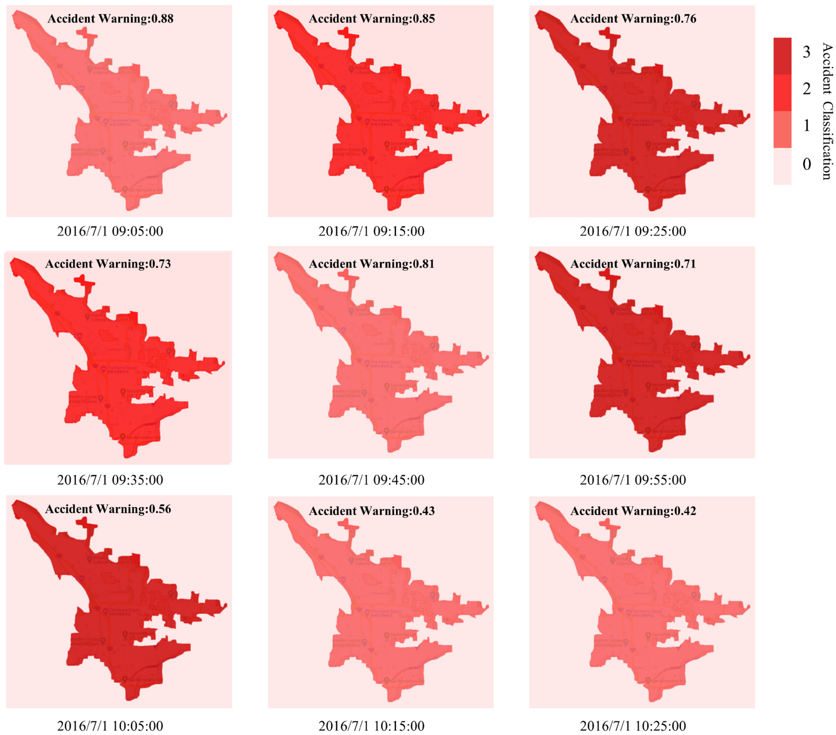


Fig. 11. The traffic accident profiling.

level of the accident is often related to external factors. We currently do not have more accident-related datasets, so we cannot improve our ability to predict accident levels.

## 6 CONCLUSION

Traffic accident prediction is very important and challenging. In this article, we propose a multi-task spatio-temporal graph representation learning framework, which embeds the traffic state through the spatio-temporal graph convolutional encoder and then uses a multi-task scheme to generate traffic accident profiling. In addition, we propose an application framework based on edge computing to reduce model calculation time by offloading calculations to edge nodes.

Although the model proposed in this article has a good effect on traffic accident profiling, there are still many shortcomings, and there are many aspects that need to be improved. On one hand, we can add traffic-related data such as points of interest and accident-prone points to improve the performance of accident classification. On the other hand, we can mine and analyze accident data on the constructed accident profile. These improvements will be studied in future work.

## REFERENCES

[1] Martin W. Adler, Jos van Ommeren, and Piet Rietveld. 2013. Road congestion and incident duration. *Economics of Transportation* 2, 4 (2013), 109–118. DOI: https://doi.org/10.1016/j.ecotra.2013.12.003

[2] Unaiza Alvi, Muazzam A. Khan Khattak, Balawal Shabir, Asad Waqar Malik, and Sher Ramzan Muhammad. 2020. A comprehensive study on IoT based accident detection systems for smart vehicles. *IEEE Access* 8 (2020), 122480–122497. DOI: https://doi.org/10.1109/ACCESS.2020.3006887

[3] Ricardo De Azevedo, Gabriel Resende Machado, Ronaldo Ribeiro Goldschmidt, and Ricardo Choren. 2020. A reduced network traffic method for IoT data clustering. *ACM Transactions on Knowledge Discovery from Data* 15, 1, Article 13 (Dec. 2020), 23 pages. DOI: https://doi.org/10.1145/3423139

[4] Li-Yen Chang and Wen-Chieh Chen. 2005. Data mining of tree-based models to analyze freeway accident frequency. *Journal of Safety Research* 36, 4 (2005), 365–375. DOI: https://doi.org/10.1016/j.jsr.2005.06.013

[5] Chao Chen, Xiaoliang Fan, Chuanpan Zheng, Lujing Xiao, Ming Cheng, and Cheng Wang. 2018. SDCAE: Stack denoising convolutional autoencoder model for accident risk prediction via traffic big data. In *Proceedings of the 2018 6th International Conference on Advanced Cloud and Big Data*. IEEE, 328–333.

[6] Cen Chen, Kenli Li, Sin G. Teo, Xiaofeng Zou, Keqin Li, and Zeng Zeng. 2020. Citywide traffic flow prediction based on multiple gated spatio-temporal convolutional neural networks. *ACM Transactions on Knowledge Discovery from Data* 14, 4, Article 42 (May 2020), 23 pages. DOI: https://doi.org/10.1145/3385414

[7] Chao Chen, Karl Petty, Alexander Skabardonis, Pravin Varaiya, and Zhanfeng Jia. 2001. Freeway performance measurement system: Mining loop detector data. *Transportation Research Record* 1748, 1 (2001), 96–102. DOI: https://doi.org/10.3141/1748-12

[8] Quanjun Chen, Xuan Song, Harutoshi Yamada, and Ryosuke Shibasaki. 2016. Learning deep representation from big and heterogeneous data for traffic accident inference. *Proceedings of the AAAI Conference on Artificial Intelligence* 30, 1 (Feb. 2016), 338–344. DOI: https://doi.org/10.1609/aaai.v30i1.10011

[9] Muhammad Uzair Ghazi, Muazzam A. Khan Khattak, Balawal Shabir, Asad W. Malik, and Muhammad Sher Ramzan. 2020. Emergency message dissemination in vehicular networks: A review. *IEEE Access* 8 (2020), 38606–38621. DOI: https://doi.org/10.1109/ACCESS.2020.2975110

[10] Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. 2019. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence* 33, 1 (Jul. 2019), 922–929. DOI: https://doi.org/10.1609/aaai.v33i01.3301922

[11] William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in Neural Information Processing Systems* 30 (2017), 1024–1034.

[12] William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Representation Learning on Graphs: Methods and Applications. *IEEE Data Engineering Bulletin* 40, 3 (2017), 52–74.

[13] David K. Hammond, Pierre Vandergheynst, and Rémi Gribonval. 2011. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis* 30, 2 (2011), 129–150. DOI: https://doi.org/10.1016/j.acha.2010.04.005

[14] Xiao Han, Guojiang Shen, Xi Yang, and Xiangjie Kong. 2020. Congestion recognition for hybrid urban road systems via digraph convolutional network. *Transportation Research Part C: Emerging Technologies* 121 (2020), 102877. DOI: https://doi.org/10.1016/j.trc.2020.102877

[15] Steven Haynes, Prudencia Charles Estin, Sanela Lazarevski, Mekala Soosay, and Ah-Lian Kor. 2019. Data analytics: Factors of traffic accidents in the UK. In *Proceedings of the 2019 10th International Conference on Dependable Systems, Services and Technologies*. 120–126. DOI: https://doi.org/10.1109/DESSERT.2019.8770021

[16] Yue Hu and Daniel B. Work. 2021. Robust tensor recovery with fiber outliers for traffic events. *ACM Transactions on Knowledge Discovery from Data* 15, 1, Article 6 (Dec. 2021), 27 pages. DOI: https://doi.org/10.1145/3417337

[17] Chao Huang, Chuxu Zhang, Peng Dai, and Liefeng Bo. 2019. Deep dynamic fusion network for traffic accident forecasting. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2673–2681.

[18] Chao Huang, Chuxu Zhang, Jiashu Zhao, Xian Wu, Dawei Yin, and Nitesh Chawla. 2019. Mist: A multiview and multimodal spatial-temporal learning framework for citywide abnormal event forecasting. In *Proceedings of the World Wide Web Conference*. 717–728.

[19] Feifeng Jiang, Kwok Kit Richard Yuen, and Eric Wai Ming Lee. 2020. A long short-term memory-based framework for crash detection on freeways with traffic data of different temporal resolutions. *Accident Analysis & Prevention* 141 (2020), 105520. DOI: https://doi.org/10.1016/j.aap.2020.105520

[20] Diederik P. Kingma and Max Welling. 2014. Stochastic gradient VB and the variational auto-encoder. In *Proceedings of the 2nd International Conference on Learning Representations*. Vol. 19.

[21] Thomas N. Kipf and Max Welling. 2016. Variational Graph Auto-Encoders. arXiv:1611.07308. Retrieved from https://arxiv.org/abs/1611.07308.

[22] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *Proceedings of the 5th International Conference on Learning Representations (ICLR'17)*. Toulon.

[23] Xiangjie Kong, Shiqin Tong, Haoran Gao, Guojiang Shen, Kailai Wang, Mario Collotta, Ilsun You, and Sajal Das. 2020. Mobile edge cooperation optimization for wearable internet of things: A network representation-based framework. *IEEE Transactions on Industrial Informatics* 17, 7 (2020), 5050–5058. DOI: https://doi.org/10.1109/TII.2020.3016037

[24] Xiangjie Kong, Bing Zhu, Guojiang Shen, Tewabe Chekole Workneh, Zhanhao Ji, Yang Chen, and Zhi Liu. 2021. Spatial-temporal-cost combination based taxi driving fraud detection for collaborative internet of vehicles. *IEEE Transactions on Industrial Informatics* 18, 5 (2021), 3426–3436.

[25] Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua. 2015. A convolutional neural network cascade for face detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

[26] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2018. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *Proceedings of the 6th International Conference on Learning Representations (ICLR'18)*. Vancouver, BC.

[27] Yuxuan Liang, Kun Ouyang, Junkai Sun, Yiwei Wang, Junbo Zhang, Yu Zheng, David Rosenblum, and Roger Zimmermann. 2021. Fine-grained urban flow prediction. In *Proceedings of the Web Conference 2021*. 1833–1845.

[28] Yuxuan Liang, Kun Ouyang, Yiwei Wang, Ye Liu, Junbo Zhang, Yu Zheng, and David S. Rosenblum. 2020. Revisiting convolutional neural networks for citywide crowd flow analytics. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 578–594.

[29] Lei Lin, Qian Wang, and Adel W. Sadek. 2015. A novel variable selection method based on frequent pattern tree for real-time traffic accident risk prediction. *Transportation Research Part C: Emerging Technologies* 55 (2015), 444–459. DOI: https://doi.org/10.1016/j.trc.2015.03.015 Engineering and Applied Sciences Optimization (OPT-i) - Professor Matthew G. Karlaftis Memorial Issue.

[30] Xinhua Mao, Changwei Yuan, Jiahua Gan, and Shiqing Zhang. 2019. Risk factors affecting traffic accidents at urban weaving sections: Evidence from China. *International Journal of Environmental Research and Public Health* 16, 9 (2019), 1542. DOI: https://doi.org/10.3390/ijerph16091542

[31] Sobhan Moosavi, Mohammad Hossein Samavatian, Arnab Nandi, Srinivasan Parthasarathy, and Rajiv Ramnath. 2019. Short and long-term pattern discovery over large-scale geo-spatiotemporal data. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, New York, NY, 2905–2913. DOI: https://doi.org/10.1145/3292500.3330755

[32] Sobhan Moosavi, Mohammad Hossein Samavatian, Srinivasan Parthasarathy, Radu Teodorescu, and Rajiv Ramnath. 2019. Accident risk prediction based on heterogeneous sparse data: New dataset and insights. In *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, New York, NY, 33–42. DOI: https://doi.org/10.1145/3347146.3359078

[33] Alameen Najjar, Shun'ichi Kaneko, and Yoshikazu Miyanaga. 2017. Combining satellite imagery and open data to map road safety. *Proceedings of the AAAI Conference on Artificial Intelligence* 31, 1 (Feb. 2017), 4524–4530. DOI: https://doi.org/10.1609/aaai.v31i1.11168

[34] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. 2021. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research* 22, 57 (2021), 1–64.

[35] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, 701–710. DOI: https://doi.org/10.1145/2623330.2623732

[36] Honglei Ren, You Song, Jingwen Wang, Yucheng Hu, and Jinzhi Lei. 2018. A deep learning approach to the citywide traffic accident risk prediction. In *Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems*. 3346–3351. DOI: https://doi.org/10.1109/ITSC.2018.8569437

[37] Guojiang Shen, Zhenzhen Zhao, and Xiangjie Kong. 2021. GCN2CDD: A commercial district discovery framework via embedding space clustering on graph convolution networks. *IEEE Transactions on Industrial Informatics* 18, 1 (2021), 356–364. DOI: https://doi.org/10.1109/TII.2021.3051934

[38] Chao Song, Youfang Lin, Shengnan Guo, and Huaiyu Wan. 2020. Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 1 (Apr. 2020), 914–921. DOI: https://doi.org/10.1609/aaai.v34i01.5438

[39] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. LINE: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1067–1077. DOI: https://doi.org/10.1145/2736277.2741093

[40] Rianne Van Den Berg, Leonard Hasenclever, Jakub M. Tomczak, and Max Welling. 2018. Sylvester normalizing flows for variational inference. In *Proceedings of the 34th Conference on Uncertainty in Artificial Intelligence 2018*. Association For Uncertainty in Artificial Intelligence (AUAI), 393–402.

[41] Chengxin Wang, Yuxuan Liang, and Gary Tan. 2021. PRNet: A Periodic Residual Learning Network for Crowd Flow Forecasting. arXiv:2112.06132. Retrieved from https://arxiv.org/abs/2112.06132.

[42] Pengyang Wang, Xiaolin Li, Yu Zheng, Charu Aggarwal, and Yanjie Fu. 2021. Spatiotemporal representation learning for driving behavior analysis: A joint perspective of peer and temporal dependencies. *IEEE Transactions on Knowledge and Data Engineering* 33, 2 (2021), 728–741. DOI: https://doi.org/10.1109/TKDE.2019.2935203

[43] Senzhang Wang, Jiaqiang Zhang, Jiyue Li, Hao Miao, and Jiannong Cao. 2021. Traffic accident risk prediction via multi-view multi-task spatio-temporal networks. *IEEE Transactions on Knowledge and Data Engineering* (2021), 1–1. DOI: https://doi.org/10.1109/TKDE.2021.3135621

[44] Wencheng Wang, Zhenzhou Yuan, Yang Yang, Xiaobao Yang, and Yanting Liu. 2019. Factors influencing traffic accident frequencies on urban roads: A spatial panel time-fixed effects error model. *PLOS ONE* 14, 4 (2019), e0214539. DOI: https://doi.org/10.1371/journal.pone.0214539

[45] Jiexia Ye, Juanjuan Zhao, Kejiang Ye, and Chengzhong Xu. 2020. How to build a graph-based deep learning architecture in traffic domain: A survey. *IEEE Transactions on Intelligent Transportation Systems* 23, 5 (2020), 3904–3924. DOI: https://doi.org/10.1109/TITS.2020.3043250

[46] Bing Yu, Haoteng Yin, and Zhanxing Zhu. 2018. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence.* DOI: https://doi.org/10.24963/ijcai.2018/505

[47] Le Yu, Bowen Du, Xiao Hu, Leilei Sun, Liangzhe Han, and Weifeng Lv. 2021. Deep spatio-temporal graph convolutional network for traffic accident prediction. *Neurocomputing* 423 (2021), 135–147.

[48] Rongjie Yu and Mohamed Abdel-Aty. 2013. Multi-level bayesian analyses for single- and multi-vehicle freeway crashes. *Accident Analysis & Prevention* 58 (2013), 97–105. DOI: https://doi.org/10.1016/j.aap.2013.04.025

[49] Zhuoning Yuan, Xun Zhou, and Tianbao Yang. 2018. Hetero-ConvLSTM: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.* ACM, New York, NY, 984–992. DOI: https://doi.org/10.1145/3219819.3219922

[50] Fan Zhou, Qing Yang, Ting Zhong, Dajiang Chen, and Ning Zhang. 2020. Variational graph neural networks for road traffic prediction in intelligent transportation systems. *IEEE Transactions on Industrial Informatics* 17, 4 (2020), 2802–2812.

[51] Zhengyang Zhou, Yang Wang, Xike Xie, Lianliang Chen, and Hengchang Liu. 2020. RiskOracle: A minute-level city-wide traffic accident forecasting framework. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 1 (Apr. 2020), 1258–1265. DOI: https://doi.org/10.1609/aaai.v34i01.5480

Received 30 March 2021; revised 23 July 2022; accepted 14 September 2022