Identifying Critical Congested Roads Based on Traffic Flow-Aware Road Network Embedding

Jing Zhao[1,2], Peng Cheng[3], Qixiang Ge[2], Xun Zhu[2], Lei Chen[4,5], Xi Guo[6], Jinshan Sun[1], and Yangfang Yang[7](✉)

[1] State Key Laboratory of Precision Blasting, Jianghan University, Wuhan, China
{jingzhao,sunjinshan}@jhun.edu.cn
[2] School of Artificial Intelligence, Jianghan University, Wuhan, China
geqixiang@stu.jhun.edu.cn, zhuxun@jhun.edu.cn
[3] East China Normal University, Shanghai, China
pcheng@sei.ecnu.edu.cn
[4] The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China
leichen@cse.ust.hk
[5] The Hong Kong University of Science and Technology, Hong Kong, China
[6] University of Science and Technology Beijing, Beijing, China
xiguo@ustb.edu.cn
[7] China Academy of Transportation Sciences, Beijing, China
yangyf@motcats.ac.cn

Abstract. Traffic congestion occurs frequently and concurrently on urban road networks, and may cause widespread traffic paralysis if not controlled promptly. To relieve traffic congestion and avoid traffic paralysis, it is significant to identify critical congested roads with great propagation influence on others. Existing studies mainly focus on topological measures and statistical approaches to evaluate the criticality of road segments. However, critical congested roads are generated by dynamic changes in traffic flow, so that identifying them involves both the network structure and dynamic traffic flows is required. In this paper, we propose a novel road network embedding model, called *Seg2Vec*, to learn comprehensive features of road segments considering both the road structural information and traffic flow distribution. The Seg2Vec model combines a Markov Chain-based random walk with the *Skip-gram* model. The random walk is conducted on the road network based on the transition probabilities computed from historical trajectory data. Moreover, we define the *propagation influence* of a congested road by a score function based on the learned road representation. The goal is to find the *critical congested roads* with top-K propagation influences. Evaluation experiments are conducted to verify the effectiveness and efficiency of the proposed

Supported by the Project of Hubei Provincial Science and Technology Department(2020BCA084), and Open Foundation of Key Laboratory of Transport Industry of Big Data Application Technologies for Comprehensive Transport (2021B1202).

X. Yang et al. (Eds.): ADMA 2023, LNAI 14176, pp. 598–613, 2023.
https://doi.org/10.1007/978-3-031-46661-8_40

method. A case study of identifying critical congested roads from a congestion cluster is also demonstrated. The identified critical congested roads can facilitate decision-making for traffic management.

Keywords: Road network embedding · Critical congested roads · Congestion propagation · Trajectory data

1 Introduction

Traffic congestion has become a major issue to urban traffic system, since it causes waste of time, energy consumption and air pollution, which further leads to economic cost and damages to citizens' health. In urban road networks, traffic congestion occurs frequently and concurrently, so that it may cause widespread traffic paralysis if not controlled promptly. Therefore, to relieve traffic congestion and avoid traffic paralysis, it is significant to identify *critical congested roads* with great propagation influence on other roads.

Existing studies on identifying critical roads mainly focuses on topological measures and statistical approaches to evaluate the criticality of roads [2,6,9,22]. For instance, a topological graph measure called *betweenness* is defined in [2] to identify critical roads that are located on many shortest paths between other vertices. Guo et al. proposed the weighted degree and the impact distance as the two major measures to identify the most influential locations [6]. Li et al. also studied the percolation process on road networks, and identified bottleneck roads that play a critical role in connecting different functional clusters [9,22].

The above approaches have been proposed to identify fixed critical roads that caused by the essential structure of road networks. However, critical congested roads are generated by dynamic changes of traffic flow or sudden increases of traffic demand, such as traffic accidents, road damage, etc. In order to discover the root cause of traffic congestion, it is necessary to measure the interactions among congested roads and the surrounding roads under real-time conditions. In other words, a congestion occurs on the road segment with great propagation influence on other road segments is considered as the critical congested road. The propagation influence of a road segment is determined by many factors, such as time, location, and properties of road. As a result, identifying critical congested roads involves both the network structure and dynamic traffic flows. How to obtain comprehensive road network features that consider both sides and quantitatively analyze the congestion propagation characteristics of road segments are significant issues studied in this paper.

In order to learn comprehensive features of road segments, road network embedding is extended from graph embedding methods [1]. Graph embedding methods use truncated random walks to map a network to low-dimensional representations, while preserving most of the network information. The learned feature representations are applied widely for network analysis, such as node classification, link prediction and community detection [21]. It has been verified that graph embedding methods are able to capture structural information of road networks [8]. To quantify the traffic interaction between road segments, Road2Vec [10] learns the feature representation of road segments using travel routes. However, straightforwardly using real trajectories to train the model makes the result bias to main roads with high frequencies, and the local structure of road network is ignored. Therefore, training the sequences of graph nodes in an appropriate manner which considers both the road structural information and traffic flow distribution is required.

In this paper, we propose a traffic flow-aware road network embedding model, called *Seg2Vec*, which leverages trajectory data to enhance graph embedding for road representation. The Seg2Vec model combines a Markov Chain-based random walk with the Skip-gram model [14]. The random walk is conducted on the road network based on the transition probabilities computed from historical trajectories. The learned road representation is used to measure the traffic similarity between road segments, which is further used to evaluate the propagation influences of congested roads. Moreover, a critical congested road identification algorithm is proposed to identify critical roads with top-K propagation influences. By monitoring the travel speeds of vehicles, it is able to detect traffic congestion and identify critical congested roads in real time.

Our contributions can be summarized as follows:

1. We propose a road network embedding model, Seg2Vec, to measure the traffic similarity between road segments. The Seg2Vec model learns feature representation of road segments considering both the road network structure and traffic flow distribution.
2. We define the congestion propagation influence of a congested road by a score function based on the learned feature representation of road segments. To the best of our knowledge, it is the first work to measure congestion propagation influence by road embedding.
3. We propose a critical congested road identification algorithm to identify critical roads with top-K propagation influences in real time.
4. We conduct extensive experiments to verify the effectiveness and efficiency of the proposed method using real taxis' trajectories. A case study of identifying critical congested roads for a congestion cluster is also demonstrated.

The rest of this paper is organized as follows. In Sect. 2, we introduce the preliminaries and the problem definition of this paper. The proposed road network embedding model and critical congested roads identification method are introduced in Sect. 3 and Sect. 4, respectively. Experimental results with a case study are shown in Sect. 5. Related work is introduced in Sect. 6. Finally, we conclude the paper in Sect. 7.

2 Problem Statement

2.1 Preliminaries

Definition 1 ***Road network:*** *A road network is defined as a directed graph $G = (V, E)$, where V is a set of nodes and E is a set of directed edges on the* *road network. Each edge* $e_{ij} \in E$ *is determined by a source node* v_i *and a target node* v_j*, i.e.,* $e_{ij} = (v_i \rightarrow v_j)$*,* $v_i, v_j \in V$*.*

Definition 2 ***n-hop neighborhood:*** *Given a (un)directed graph* $G = (V, E)$*, the n-hop neighborhood of a node* $u \in V$*, represented as* $N_n(u)$*, is defined as nodes within n-hop distance from u on the graph, which means the shortest path from u to* $v \in N_n(u)$ *or v to u is not large than n, where n is a given parameter.*

Note that, we consider both of the upstream and downstream neighborhood of a node as its neighborhood. This is because traffic congestion can propagate both backward and forward, i.e., *bi-directional*.

Definition 3 ***Trajectory:*** *A trajectory s consists of a sequence of location points* $\{l_1, l_2, \ldots, l_{|s|}\}$*, where each location point* $l_i = (x_i, y_i, v_i, t_i)$ *corresponds to a location coordinate* (x_i, y_i) *with a velocity* v_i *at a time stamp* t_i*, where* $i \in [1, |s|]$*.*

Traffic congestion generally causes a slowdown in traffic speed on specific roads, which lasts for a short time but happens frequently. Here, we define the congested road by comparing vehicles' speeds with the free-flow speed. The free-flow speed is the average speed that a driver would travel if there is no congestion or other adverse conditions. We use the F percentile of all valid speeds on each road segment as its free-flow speed, as in [20], with a default value of 85. By monitoring vehicles' speeds, traffic congestion is detected rapidly and accurately.

Definition 4 ***Road speed:*** *Given an edge e on a road network, a set of sub trajectories* $S_{e,t}$ *that matched to edge e during time period t, such that* $|S_{e,t}| \geq min\_sup$*, where* min_sup *is a confidence threshold. The road speed* $v_{e,t}$ *is calculated as follows:*

$$v_{e,t} = \frac{1}{|S_{e,t}|} \sum_{s \in S_{e,t}} \sum_{l_i \in s} \frac{v_i}{|s|}, \tag{1}$$

i.e., the average of the speeds of all trajectories in $S_{e,t}$*, where the speed of a trajectory* $s \in S_{e,t}$ *is the average travel speed of all location points* $l_i \in s$*.*

Definition 5 ***Congested road:*** *Given an edge* $e \in E$ *with road speed* $v_{e,t}$*, and a free-flow speed* v_f*. The edge e is defined as a congested road, if* $v_{e,t}$ *is less than C percentage of* v_f*, where C is a default parameter.*

2.2 Problem Definition

To evaluate the propagation influence of a congested road, we construct a road graph which considers each road segment as a node, and learn feature representation of road segments. Then, we generate congestion clusters from a set of congested roads, and identify critical congested roads for each congestion cluster.

Definition 6 ***Road graph:*** *Given a directed road network $G = (V, E)$, a road graph is defined as an undirected graph $G_r = (V_r, E_r)$, where V_r contains all road segments in E, i.e., a node $v_i \in V_r$ corresponds to an edge $e_i \in E$. An edge $e_{ij} \in E_r$ between v_i and v_j on G_r exists if there is a path from edge e_i to e_j or from e_j to e_i on G.*

Definition 7 ***Road representation:*** *Given a road graph $G_r = (V_r, E_r)$, the road representation aims to learn a feature representation of road segments, denoted as $f : V_r \rightarrow \mathbb{R}^d$, which projects each road to a d-dimensional feature vector.*

Definition 8 ***Congestion cluster:*** *Given a set of congested roads E_t during time period t, a congestion cluster Ec is defined as a subset of E_t, such that each congested road $e \in Ec$ has at least one n-hop neighborhood that is congested, and all of the congested n-hop neighborhood of e belongs to Ec, i.e., $|N_n(e) \cap E_t| \geq 1$ and $N_n(e) \cap E_t \subseteq Ec$. In addition, $|Ec| \geq c$, where c is a default parameter.*

Definition 9 ***Propagation influence:*** *Given a congestion cluster Ec, and the road representation f, each congested road $u \in Ec$ has a propagation influence with respect to its n-hop neighborhood in Ec, denoted as $Nc(u)$. The propagation influence of u is defined as follows:*

$$PI(u \mid Nc(u)) = w_u \sum_{e \in Nc(u)} Sim(f(u), f(e)), \tag{2}$$

where, $Sim()$ is a predefined similarity metric, e.g., cosine similarity, and w_u is a normalized occurrence probability of road segment u.

The value of w_u is calculated by historical trajectory data. The intuition is that, the larger the occurrence probability is, the more likely the road segment affects other road segments.

Definition 10 ***Critical Congested Roads:*** *Given a congestion cluster Ec, and an integer $K \leq |Ec|$. The critical congested roads in Ec are defined as a set of congested roads $E_K \subset Ec$ with maximum size K, such that the sum of propagation influence of each road in E_K is maximal.*

3 Seg2Vec: Traffic Flow-Aware Road Network Embedding

As described above, the propagation influence of a congested road is calculated by the feature representation of road segments. In this paper, we propose Seg2Vec, a traffic flow-aware road network embedding model that learns the structural information of road segments by simulating trajectories using a Markov chain-based random walk. The intuition is that traffic flows on road networks follow certain spatiotemporal distributions, and road segments with high co-occurrences along trajectories indicate high traffic similarities among them. Therefore, we use historical trajectories to capture the transition probability between road segments, then generate neighbor nodes by a sampling strategy based on the precomputed transition matrix, with details as follows.

3.1 Markov Chain-Based Random Walks

The Markov chain is a stochastic process that satisfies the Markov property [4]. The process describes a sequence of possible events in which the probability of each event only depends on the state of the previous event. In this paper, we consider each road segment as a state, the probability of state change between road segments is called transition probability, calculated by historical data.

Given a set of trajectories S mapped to the road graph G_r, each trajectory consists of a sequence of nodes (i.e., road segments). The transition probability distribution can be represented by a transition matrix P, and each element p_{ij}^T represents the probability of changing the state from v_i to v_j during time period T, which is evaluated by the following equation:

$$p_{ij}^T = Pr^T(s_{n+1} = v_j \mid s_n = v_i) = \frac{|S^T(v_i \to v_j)|}{|S^T(v_i)|}, \tag{3}$$

where $|S^T(v_i)|$ is the outflow of road v_i during time period T, and $|S^T(v_i \to v_j)|$ is the traffic flow from road v_i to road v_j during time period T. In addition, the variable $n = 0, 1, ..., l-1$, where l is the walk length. Note that, the transition probability of current node only depends on the previous node, so the random walk is the first-order Markov chain.

Benefits of Random Walks. The benefits of random walks are reflected in both effectiveness and efficiency. For effectiveness, the traffic distribution of real trajectories essentially has a bias to the main roads with high frequencies, which leads to the learned features bias to these roads. The proposed Seg2Vec model simulates equal number of trajectories for each road using a Markov chain-based random walk, so that it can preserve local network structure with less bias. Moreover, training random walks is more efficient, since the number of walks and walk length of the simulated walks are optional. In addition, random walks can provide sample reuse. For instance, a trajectory with length l will generate k samples for $l-k$ nodes at once, where k is the context size, and $l > k$. Then, for N source nodes each with r random walks, there are $Nrk(l-k)$ samples generated in total.

3.2 Feature Learning Model

Given a road graph $G_r = (V_r, E_r)$, and a set of simulated trajectories S generated by the random walks introduced above. The objective function of the Seg2Vec model is to maximize the log-probability of observing the neighborhood $N(u)$ for a source node $u \in V_r$ conditioned on its feature representation, given by f:

$$\max_f \sum_{u \in V} \log Pr(N(u) \mid f(u)), \tag{4}$$

where, the neighborhood $N(u)$ of road segment u is determined by a sliding window of size k over consecutive road segments of trajectories.

Based on the conditional independence, the probability of observing a neighborhood $n_i \in N(u)$, given node u, is calculated by the softmax function:

$$Pr(n_i \mid f(u)) = \frac{\exp(f(n_i) \cdot f(u))}{\sum_{v \in V_r} \exp(f(v) \cdot f(u))}. \tag{5}$$

Based on the above assumptions, the feature learning model is trained by optimizing the objective function using stochastic gradient ascent (SGD) and back propagation [17].

3.3 The Seg2Vec Algorithm

The pseudo-code of Seg2Vec is given in Algorithm 1. The algorithm consists of three phases, i.e., computing the transition matrix (Line 1), Markov chain-based random walks (Line 2–14) and optimization using SGD (Line 15). The first phase is constructing a transition matrix P by computing the transition probabilities between road segments based on historical trajectories. Since we aim to learn representations for all nodes, we simulate r walks per node with fixed length l starting from each node. The sampling strategy for the random walk is the alias sampling, which is also used in [5,18]. The alias sampling can be done efficiently in $O(1)$ time complexity, with the precomputed transition matrix for the first-order Markov chain. Finally, the generated walks are selected as the input of the training model, and the feature representations of road segments are optimized using SGD. Note that, each phase described above can be conducted in parallel, which contributes to the scalability of the proposed model.

Algorithm 1 The Seg2Vec algorithm.

Input: G, trajectories S, dimensions d, walks per node r, length l, context size k.
Output: Feature representation f.

1: P =ComputeTransitionPr(G, S) ▷ Preprocess the transition matrix
2: $walks \leftarrow \emptyset$ ▷ Initialize the walks to empty
3: **for** $iter \leftarrow 1$ to r **do**
4:     **for all** nodes $u \in V$ **do**
5:         $walk \leftarrow [u]$ ▷ Initialize the walk list as the source node u
6:         **for** $n \leftarrow 1$ to $l-1$ **do**
7:             $v_{cur} = walk[n-1]$
8:             $N(v_{cur})$ = GetNeighbors(v_{cur}, G) ▷ Get the neighborhoods of v_{cur}
9:             v_{next} =AliasSample($v_{cur}, N(v_{cur}), P$)
10:             Append v_{next} to $walk$
11:         **end for**
12:         Append $walk$ to $walks$
13:     **end for**
14: **end for**
15: f = StochasticGradientDescent(k,d,$walks$)
16: **return** f

4 Critical Congested Road Identification

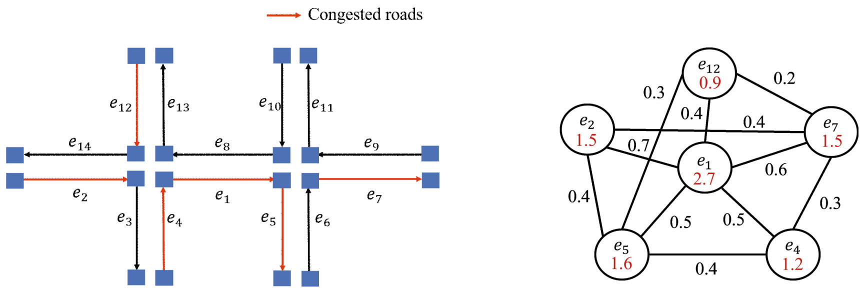
Consider a real-time traffic monitoring system that consecutively collects vehicles' GPS trajectory data, and detects congested roads based on the collected trajectories. In order to identify critical congested roads among a series of detected congested roads, we propose a critical congested roads identification method as shown in Algorithm 2. We first generate several congestion clusters based on the n-hop neighborhoods for each congested road (Line 1). Then, the propagation influence of each congested road is computed by the feature representation of road segments (Line 6). Finally, we select the critical roads with top-K propagation influences in each cluster (Line 9-10).

Algorithm 2 The CriticalRoadIdentification algorithm.

Input: Congested roads E_t, order n, K, feature representation f.
Output: Overall top-K critical roads R.

1: E_c =GenerateCongestCluster(E_t, n) ▷ Generate a set of congestion clusters for E_t
2: $R \leftarrow \emptyset$ ▷ Initialize the top-K critical roads for all congestion clusters
3: **for all** cluster $c \in E_c$ **do**
4:     $r_c \leftarrow \emptyset$ ▷ Initialize the critical road list for cluster c
5:     **for all** road $e \in c$ **do**
6:         $PI(e) =$ ComputePI(e, c, f) ▷ Compute the propagation influence of road e
7:         Append $(e, PI(e))$ to r_c
8:     **end for**
9:     Sort the propagation influences of roads in r_c in descending order
10:     Select the top-K critical roads in r_c and add to result R
11: **end for**
12: **return** R

Figure 1 illustrates an instance of identifying critical congested roads. While Fig. 1a shows a set of congested roads in red, it is difficult to explain the congestion or decide how to relieve them. In this work, we provide an alternative solution to explain the root cause of congestion. Figure 1b shows a congestion cluster, in which each node represents a congested road in Fig. 1a. The value on a node (in red) represents its propagation influence, while the value on an edge (in black) represents the similarity between two roads. The propagation influence is calculated by Eq. 2, assuming the occurrence probability of each node is 1. Here, we consider both the upstream and downstream roads of a congested road as its neighborhoods, e.g., e_2 and e_7 are both e_1's neighborhoods. Moreover, we consider the relationship between any 2-hop neighborhoods, e.g., e_2 and e_7 are mutual neighborhoods. By ranking the propagation influences of congested roads, it is more intuitive to express the criticality of each one. For instance, it is evident that the propagation influence of e_1 is the greatest one, which has strong influence on other roads, followed by e_5, e_2 and e_7.



(a) Detected congested roads (b) The congestion cluster in (a)

Fig. 1. An instance of identifying critical congested roads

5 Experiments

5.1 Datasets and Preprocessing

We use real GPS trajectories generated by taxicabs in Chengdu from October 1st to October 20th, 2016, provided by Didi Chuxing[1] There are about 3,503,276 records of trip orders with 671 millions trajectory points, each of which contains a time stamp, a longitude and latitude, an encrypted driver ID and order ID. We use trajectory data of two weeks for training, and the rest for testing. The road network of Chengdu is obtained from Open Street Map (OSM)[2] with about 3151 nodes and 7336 edges on the road network. Figure 2a shows the original distribution of trajectory data (in red) generated on October 1st, 2016, inside

(a) The raw trajectories (b) The preprocessed trajectories

Fig. 2. The distribution of trajectory data on the road network

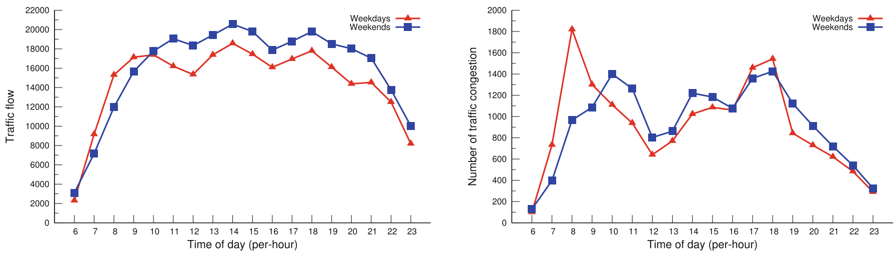
[1] Didi chuxing, https://outreach.didichuxing.com/app-vue/dataList.
[2] OSM, https://www.openstreetmap.org/,

the Second Ring Road of Chengdu's road network. Since the raw trajectory data is unordered and full of noise, data preprocessing is necessary. The preprocessing mainly contains three steps, i.e., coordinates transformation, data cleaning and map matching. We utilize an efficient map matching algorithm, called ST-Matching algorithm [12], to map the trajectories to the road network. After data preprocessing phase, the raw trajectory data is transformed into 3,689,336 records of time-sorted trajectories, with about 657 millions of location points mapped to the road network, as shown in Fig. 2b.

5.2 Experimental Setup

We perform extensive experiments and a case study to evaluate the quality and efficiency of the proposed method. The quality of Seg2Vec is evaluated by cosine similarities of the learned road representation, as well as the effectiveness of critical congested road identification, comparing with Road2Vec [10]. The efficiency of our methods is evaluated by the execution time of offline training and online process. In addition, a case study is also demonstrated to show the effectiveness of our proposed method. Our experiments are performed on a 64-bit server running Ubuntu 20.04.4 (OS) with an Intel Xeon Gold 6226R CPU @ 2.90 GHz $\times$ 32 and a 256GB RAM.

Time Division. As shown in Fig. 3, the number of traffic flow and traffic congestion on the road network is time-variant. According to the distribution of traffic flow and congestion changing by time, we divide a day into three time periods, i.e., *morning peak*, *normal* and *evening peak*. The morning peak hours of weekdays and weekends are set as 7 am to 10 am and 8 am to 11 am, respectively, while the evening peak hours are set as 5 pm to 8 pm. The remained time periods during daytime are set as normal time.



(a) Change of traffic flow (b) Change of traffic congestion

Fig. 3. Traffic distribution changing over time of day

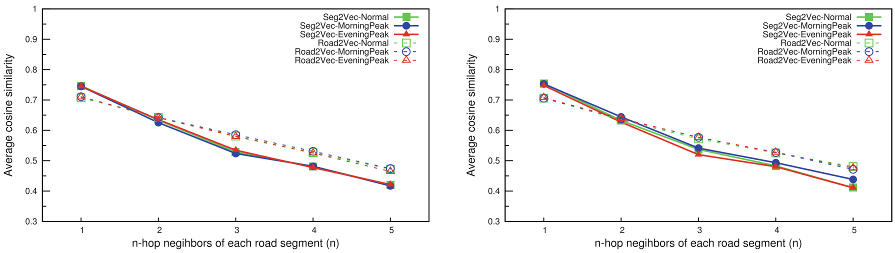
Effectiveness Evaluation. Given a congestion cluster Ec during time period $[t_s, t_e]$, and a set of critical congested roads E_K with top-K propagation influences, the effectiveness of E_K is measured by the average percentage of traffic volume reduction if deleting the trajectories passing through E_K from the congestion cluster Ec, the calculation function is as follows:

$$Effectiveness(E_K) = \frac{1}{|E_c|} \sum_{e \in Ec} \frac{|V(E_K) \cap V(e)|}{|V(e)|}, \quad (6)$$

where, $V(E_K)$ is the vehicles on the critical congested roads E_K during time period $[t_s, t_e]$, and $V(e)$ is the vehicles passing through road $e \in Ec$ during time period $[t_s, t_e + \delta t]$. In the following, we will investigate the effect of time delay on propagation influence by comparing the effectiveness by varying δt.

5.3 Evaluation Results

Traffic Similarity. We first compare the two learning models by showing the traffic similarity with varying n, i.e., the average cosine similarities of n-hop neighbors for each road segment, where $n \in [1, 5]$. In Fig. 4, the traffic similarities of Road2Vec (in hollow) and Seg2Vec (in solid) are decreased by the increase of n, which indicates the two models can capture the spatial proximity of road segments. Moreover, the traffic similarities of Seg2Vec are larger than the ones of Road2Vec when n equals to 1, while n becomes larger than 2, the traffic similarities of Seg2Vec are smaller than Road2Vec. The result indicates that Seg2Vec is more sensitive to both adjacent and distant neighbors than Road2Vec.



(a) Weekdays (b) Weekends

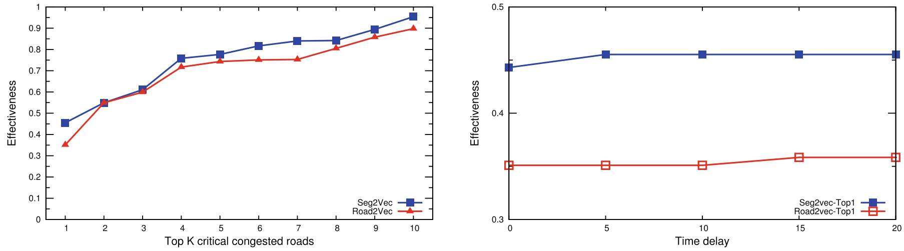
Fig. 4. Traffic similarity with varying n

Case Study. We evaluate the effectiveness of our proposed model through a case study of a congestion cluster around Tianfu Square during 9:10-9:20 on October 19th, 2016. As shown in Fig. 5a, there are 15 congested roads (in red) in the congestion cluster. To identify critical congested roads which have strong influences on others, our proposed road embedding model learns the feature representation of road segments. Based on the learned feature vectors and the proposed propagation influence function, we visualize the embedded road segments in Fig. 5b. Each congested road is represented as a node, the size of which represents its propagation influence on others. An edge between two nodes represents the two road segments that are mutual 5-hop neighborhoods, and the thickness of an edge represents the traffic similarity between them. As Fig. 5 shows, the top-2 critical roads are road-1 and road-2, which are both located in the central position of the congestion cluster and have a series of neighborhood nodes. Even though road-3 locates at the marginal position of the cluster, it has strong similarities with its neighborhoods, e.g., road-6 and road-7.

(a) Congested roads (in red) (b) Embedded road segments

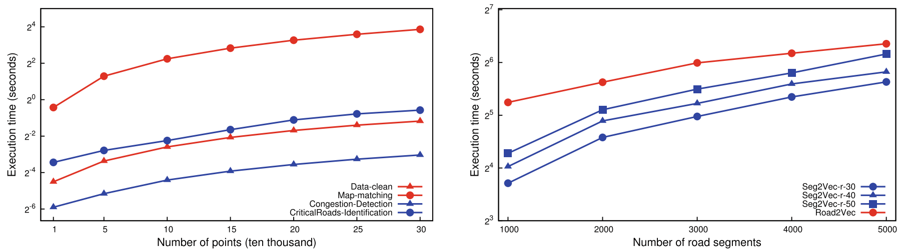
Fig. 5. A case study in Chengdu on weekday morning peak hour

Figure 6 shows the effectiveness comparison of Seg2Vec and Road2Vec. In Fig. 6a, the effectiveness of Seg2Vec for top-K critical roads is better than the ones of Road2Vec, especially for top-1 critical road. This can be explained by the sensitivity of Seg2Vec to adjacent neighbors. The effectiveness increases when K increases, since the more critical congested roads identified, the larger propagation influence on other roads. Figure 6b illustrates that the effectiveness is also affected by time delay δt. As Fig. 6b shows, the effectiveness of Seg2Vec increases when δt changes from 0 to 5 min, while the effectiveness of Road2Vec increases when δt becomes 15 min, which is caused by the time delay of congestion propagation. The result also suggests that the Top-1 critical congested road identified by Seg2Vec spreads more quickly than the one identified by Road2Vec.



(a) Top-K critical congested roads (b) Time delay δt

Fig. 6. Effectiveness with varying K and δt



(a) Online process (b) Model training time

Fig. 7. Efficiency

Efficiency. We evaluate the efficiency of the online process, which consists of data cleaning, map matching, traffic congestion detection and critical road identification. As shown in Fig. 7a, the execution time of the online process is almost within 1 min with reasonable numbers of trajectory points during 10 min, such as 300 thousand. Therefore, the proposed critical congested road identification method is efficient and suitable for online monitoring scenarios. We also compare the training time of Seg2Vec and Road2Vec. In Fig. 7b, the training time of Seg2Vec is less than Road2Vec when the road segments increases from 1000 to 5000. The result indicates the scalability of our proposed model.

Discussions. The experimental results are summarized as follows:

1. Both of Seg2Vec model and Road2Vec model can capture spatial proximity of road segments, while the Seg2Vec model is more expressive to traffic similarities of adjacent and distant neighborhoods.
2. The effectiveness of Seg2Vec is better than Road2Vec model for top-K critical congested road identification. The effectiveness becomes better when time delay increases, which is because of the delay of congestion propagation.
3. For efficiency, the online process of our proposed method is adaptive to online traffic monitoring scenarios, and the training time of Seg2Vec is also scalable.

6 Related Work

Congestion Propagation : The causal interactions among road segments are explored in [11,15] using pattern mining approaches. Liu et al. [11] proposed a Spatio-Temporal Outlier (STO) method, which divides the road network into regions, and uses a recursive approach to find frequent traffic jam propagation trees in the road network. The disadvantage of the STO method is that the road network is oversimplified to regions and the results are probably imprecise. The Spatio-Temporal Congestion (STC) [15] process is further studied to develop the STO method. Instead of regional division, it models the road network as a directed graph. The STCTree algorithm generates the most frequent subtree from all discovered congestion trees. The limitation of frequent pattern mining is that the latent propagation patterns may be infrequent. Visualization methods that analyze traffic congestion propagation are also studied in [3,20]. Wang et al. [20] provided a system for traffic congestion visualization based on GPS trajectories. Deng et al. [3] proposed a visual analytics system called VisCas, which combines a network inference model with interactive visualizations to infer the latent cascading patterns. They model the congestion cascading network based on the spatial and temporal distance between congestion events. While this kind of hand-crafted features is easy to access and understood, they can hardly reserve the complex spatial and temporal correlations of urban data.

Road Network Embedding. In order to obtain comprehensive features of road networks, road network embedding is extended from graph embedding [21]. Graph embedding generally uses truncated random walks to learn network representations while preserving the network structures [5,16,18]. DeepWalk [16] simulates uniform random walks, which is analogical to a depth-first search (DFS). LINE [18] uses a breadth-first search (BFS), and optimizes a carefully designed objective function that preserves both the local and global network structures. Node2vec [5] proposed a flexibility random walk procedure which combines the DFS and BFS by search bias parameters. Node2vec was extended to learn road segment embeddings in [8], and a case study on the Danish road network was conducted. The results suggested that the network embedding model is able to capture structural information of road networks. To quantify the traffic interaction between road segments, Road2vec [10] learns the feature representation of road segments using travel routes based on the Word2Vec model [14]. Wang et al. [19] proposed RN2Vec to learn intersections and road segments jointly by exploring geo-locality and homogeneity of them. The learned feature vectors of road networks are used for downstream tasks such as road classification and traffic flow prediction [7,13].

7 Conclusion and Future Work

In this paper, we study the problem of identifying critical congested roads for traffic congestion management. We propose a road network embedding model, Seg2Vec, to obtain comprehensive road features that consider both the network structure and traffic flow distribution. We also define a score function to evaluate the propagation influence of congested roads based on the learned road representation. The effectiveness of Seg2Vec is verified by a case study comparing with Road2vec. The scalability of Seg2Vec and the efficiency of the online process are also demonstrated. This paper focuses on measuring the congestion propagation and discovering the critical cause in the local congestion cluster. In future work, we consider to extend the local critical road identification to global critical road identification. In addition, other downstream tasks such as road classification and congestion prediction are also considered.

References

1. Cai, H., Zheng, V.W., Chang, K.C.C.: A comprehensive survey of graph embedding: Problems, techniques, and applications. IEEE Trans. Knowl. Data Eng. **30**(9), 1616–1637 (2018)
2. Demšar, U., Špatenková, O., Virrantaus, K.: Identifying critical locations in a spatial network with graph theory. Trans. GIS **12**(1), 61–82 (2008)
3. Deng, Z., et al.: Visual cascade analytics of large-scale spatiotemporal data. IEEE Trans. Visual Comput. Graphics **28**(6), 2486–2499 (2021)
4. Gagniuc, P.A.: Markov chains: from theory to implementation and experimentation. John Wiley and Sons (2017)
5. Grover, A., Leskovec, J.: Node2vec: scalable feature learning for networks. In: Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855–864. KDD '16, ACM, New York, NY, USA (2016). https://doi.org/10.1145/2939672.2939754
6. Guo, S., et al.: Identifying the most influential roads based on traffic correlation networks. EPJ Data Sci. **8**(1), 1–17 (2019). https://doi.org/10.1140/epjds/s13688-019-0207-7
7. Hu, S., et al.: Urban function classification at road segment level using taxi trajectory data: a graph convolutional neural network approach. Comput. Environ. Urban Syst. **87**, 101619 (2021)
8. Jepsen, T.S., Jensen, C.S., Nielsen, T.D., Torp, K.: On network embedding for machine learning on road networks: a case study on the Danish road network. In: 2018 IEEE International Conference on Big Data (Big Data), pp. 3422–3431. IEEE (2018)
9. Li, D., et al.: Percolation transition in dynamical traffic network with evolving critical bottlenecks. Proc. Natl. Acad. Sci. **112**(3), 669–672 (2015)
10. Liu, K., Gao, S., Qiu, P., Liu, X., Yan, B., Lu, F.: Road2vec: measuring traffic interactions in urban road system from massive travel routes. ISPRS Int. J. Geo Inf. **6**(11), 321 (2017)
11. Liu, W., Zheng, Y., Chawla, S., Yuan, J., Xing, X.: Discovering spatio-temporal causal interactions in traffic data streams. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1010–1018 (2011)

12. Lou, Y., Zhang, C., Zheng, Y., Xie, X., Wang, W., Huang, Y.: Map-matching for low-sampling-rate gps trajectories. In: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 352–361 (2009)
13. Medina-Salgado, B., Sanchez-DelaCruz, E., Pozos-Parra, P., Sierra, J.E.: Urban traffic flow prediction techniques: a review. Sustain. Comput.: Inform. Syst. **35** 100739 (2022)
14. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR abs/1301.3781 (2013)
15. Nguyen, H., Liu, W., Chen, F.: Discovering congestion propagation patterns in spatio-temporal traffic data. IEEE Trans. Big Data **3**(2), 169–180 (2016)
16. Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: Online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 701–710. KDD '14, ACM, New York, USA (2014). https://doi.org/10.1145/2623330.2623732
17. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**, 533–536 (1986)
18. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: Large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web, pp. 1067–1077 (2015)
19. Wang, M.X., Lee, W.C., Fu, T.Y., Yu, G.: On representation learning for road networks. ACM Trans. Intell. Syst. Technol. (TIST) **12**(1), 1–27 (2020)
20. Wang, Z., Lu, M., Yuan, X., Zhang, J., Van De Wetering, H.: Visual traffic jam analysis based on trajectory data. IEEE Trans. Visual Comput. Graphics **19**(12), 2159–2168 (2013). https://doi.org/10.1109/TVCG.2013.228
21. Xia, F., et al.: Graph learning: a survey. IEEE Trans. Artif. Intell. **2**(2), 109–127 (2021)
22. Zeng, G., et al.: Switch between critical percolation modes in city traffic dynamics. Proc. Natl. Acad. Sci. **116**(1), 23–28 (2019)