Electronic Research Archive

http://www.aimspress.com/journal/ERA

ERA, 31(7): 3776−3790.
DOI: 10.3934/era.2023192
Received: 31 December 2022
Revised: 16 March 2023
Accepted: 17 March 2023
Published: 28 April 2023

Research article

A construction method of urban road risky vehicles based on dynamic knowledge graph

Yongmei Zhang[1], **Zhirong Du**[1,2,*] **and Lei Hu**[3]

[1] School of Computer Science and Technology, North China University of Technology, Beijing 100144, China
[2] School of Electrical and Control Engineering, North China University of Technology, Beijing 100144, China
[3] School of Computer Information Engineering, Jiangxi Normal University, Jiangxi 330022, China

* **Correspondence:** Email: 2237zhirong@sina.com.

Abstract: The growth of the Internet of Things makes it possible to share information on risky vehicles openly and freely. How to create dynamic knowledge graphs of continually changing risky vehicles has emerged as a crucial technology for identifying risky vehicles, as well as a research hotspot in both artificial intelligence and field knowledge graphs. The node information of the risky vehicle knowledge graph is not rich, and the graph structure plays a major role in its dynamic changes. The paper presents a fusion algorithm based on relational graph convolutional network (R-GCN) and Long Short-Term Memory (LSTM) to build the dynamic knowledge graph of risky vehicles and conducts a comparative experiment on the link prediction task. The results showed that the fusion algorithm based on R-GCN and LSTM had better performance than the other methods such as GCN, DynGEM, ROLAND, and RE-GCN, with the MAP value of 0.2746 and the MRR value of 0.1075. To further verify the proposed algorithm, classification experiments are carried out on the risky vehicle dataset. Accuracy, precision, recall, and F-values were used as heat-tolerance evaluation indexes in classification experiments, the values were 0.667, 0.034, 0.422, and 0.52 respectively.

Keywords: dynamic knowledge graph; risky vehicles; R-GCN; LSTM; fusion algorithm

1. Introduction

Vehicle recognition has increasingly risen at present. One crucial step in resolving the traffic issues on the roads is the risky vehicle identification [1]. The punishment paperwork for transportation infractions has been available on the official websites of the transportation bureaus for the provinces or the public security traffic management bureaus of the public security. The probable connection between the car and the risk can be discovered from the driver's name, the type of vehicle, the license plate, the time, the location, and information on criminal behaviors. Finding the potential relation between vehicles and risks and creating a dynamic knowledge system to swiftly detect risky vehicles are critical problems at the moment.

The growth of transportation, healthcare, and other industries has been impacted by the rise of the knowledge graph in both positive and negative ways. The Shenzhen Traffic Planning and Design Research Center has been doing the majority of the research on the traffic knowledge graph by employing knowledge graph to mine the bus scenes. It has established a knowledge graph with public transport vehicles, bus routes, bus stops, card swiping records, and IC cards as entities, and the belonging, passing, neighboring, getting on, getting off, and getting out as a relationship.

In addition, Zou developed a traffic knowledge graph using the fusion of data from many sources. He fully considers the time element by using the dynamic part connected to time as attribute storage and the static part such as the road in traffic as entity storage [2]. To encourage the ongoing development and advancement of safety management level methods, Li created the urban rail transit knowledge graph based on the safety management of urban rail transit construction [3]. By discretizing and sermonizing the multi-source heterogeneous urban traffic big data, Zhou constructed the urban knowledge graph and introduced the graph convolutional network to further extract the features of the urban knowledge graph, which were processed as the input of the spatial-temporal convolutional neural network [4]. These studies show how the knowledge graph can be used in transportation. Transportation is fundamentally a complex knowledge network since it is a moving object made up of massive individuals in time and space dimensions. Digging out the cross relationships among the entire transport link, the road environment, and events beneath the urban space is the foundation of good urban traffic management. Additionally, it is crucial to the security of urban transportation.

Building a dynamic knowledge graph of urban road risky vehicles, on one hand, it facilitates the query and statistics of relevant data for urban road risky vehicles, on the other hand, it can provide rich knowledge and multiple information for urban road traffic situation analysis and prediction. Moreover, the knowledge graph of urban road risky vehicles is a knowledge graph of urban traffic containing time and space information. Compared with the general static knowledge graph, dynamic knowledge contained the temporal knowledge graph, which will change with time and space. The knowledge graph of urban road risky vehicles can provide knowledge with spatial-temporal information for subsequent tasks. Therefore, it is important to study the construction method of urban road risky vehicles dynamic knowledge graph for strengthening and improving the relevant fields of urban traffic knowledge graph.

Numerous types of trucks, different passenger automobiles, small cars, motorcycles, and bicycles are primarily explored as urban road vehicles. But there are fewer data on motorcycles and bicycles, mainly on trucks and buses.

This paper builds a dynamic knowledge graph model combining R-GCN and LSTM to address the issue of poor node information. Specifically, the whole model consists of several layers of the R-GCN network. The input of the model is the static urban road risky vehicle knowledge graph, and the output is the new urban road risky vehicle knowledge graph. The first layer operation of R-GCN is that the input graph data composed of nodes and edges, which changes the characteristics on node of the graph data from X to Z through the several hidden layers of the GCN, the relation between nodes remains unchanged during the transformation process, and updating the parameters of the inter-layer network is completed by LSTM. The LSTM is used to realize the dynamics of the knowledge graph and ensure the previous learning long-distance dependence, while the R-GCN focuses on resolving the directional influence of the edges in the knowledge graph. The main contributions include the following:

1) Create the static risky vehicles knowledge network using text data, and consider time as an entity attribute. This paper employed the R-GCN to realize the embedding representation of entity relations since the edges in the static risky vehicles knowledge graph are directional. In contrast to other networks, the R-GCN can completely consider relations in the knowledge graph.

2) The LSTM is utilized to update the R-GCN weight for the dynamics of the knowledge graph. This paper conducted experiments on link prediction and classification. The experiment results demonstrate that the presented method performs admirably on the link prediction compared to GCN, DynGEM, ROLAND, and RE-GCN. To further verify the proposed method, classification experiments are carried out on the risky vehicle dataset.

The remainder of this paper is structured as follows: Section 2 shows the related work. The proposed method for constructing the dynamic knowledge graph of risky vehicles is provided in Section 3. Experiments and results are described in Section 4, while Section 5 concludes this paper.

2. Related work

Domain information is constantly updated over time in many fields, such as risky vehicles in the transportation industry. A large number of studies have focused on various types of risky driving behaviors [5] in the transportation field such as speeding [6,7] and tailgating [8,9], risk assessment of road transport vehicles for dangerous goods based on the hierarchical fuzzy network model [10], lane change risk analysis methods of expressway vehicles [11], the violation behavior video detection methods of driving the wrong lane [12], etc. The studies are mainly for the risk type of a certain vehicle. Time is a crucial factor in the creation of knowledge graphs when research item changes with time. A time-aware knowledge representation learning method was presented by Cui et al. [13] to address the problem of learning representations for vast knowledge graphs with time labels. The related method of dynamic knowledge graph is often an extension of the static graph.

The initial correlation method to the dynamic knowledge graph is based on the matrix factorization method [14,15], whose nodes are represented by the eigenvectors of the Laplacian matrix of the graph. Li et al. update the feature vectors utilizing the previous feature vectors rather than calculating the feature vectors from the beginning for each new graph, this method is highly computationally efficient [16]. Then researchers proposed the methods based on random wandering, for example, Nyuyen et al. extended the random wandering method by specifying the step size [17], and Yu et al. used resampling of several steps in a continuous time step when the structure of the graph did not change substantially [18], and reduced the computational cost.

A popular method of the dynamic knowledge graph is a continuous point process in time. Trivedi et al. [19] took the embedded representation of nodes as input, adopted neural networks to parameterize the intensity function, and modeled the appearance of edges as a point process. Zuo et al. [20] also used a Hawks process to model the dynamics and added an attention mechanism to assess the influence of nodes' past neighbors on their present neighbors. These methods favor time-of-event prediction because the process is continuous.

The waves of deep learning have driven many supervised and unsupervised approaches. As the predecessor of deep learning, current research on neural networks pays more attention to the balance of efficiency and performance [21,22]. Currently, the most effective combination is the graph neural network and recurrent structure, the graph neural network obtains graph information and recurrent structure processes dynamics. For supervised methods, Graph Convolutional Network (GCN) [23] has been most studied, for example, modeling without any time effect, using a single model for all time steps and loss functions accumulated along the time axis. In 2022, based on the structure of GCNs (extremely high sparsity and unbalanced non-zero data distribution) and the neuromorphic characteristics of memristive crossbar circuit, Lyu et al. proposed the acceleration method including Sparse Laplace Matrix Reordering and Diagonal Block Matrix Multiplication [24]. In 2023, to balance resource cost and performance, Lyu et al. designed the multiobjective reinforcement learning (RL)-based neural architecture search (NAS) scheme, which comprehensively balances the accuracy, parameters, FLOPs, and inference latency [25].

Graph convolution network is a method that aggregates the node information by using edge information to generate a new node representation, and it can execute different learning tasks on the graph. Processing static knowledge graphs with GCN has already been very successful [26−29]. However, the processing of the dynamic knowledge graph by GCN is less studied [30−32], and the orientation of edges was not considered in the studies.

A typical unsupervised method is the Deep embedding method for dynamic graphs (DynGEM). This is a self-encoding method proposed by Goyal et al. [33] to minimize the reconstruction loss and the distance between the connected nodes in the embedding space. The depth of the architecture is commensurate with the size of the graph, and the past learned auto-encoder is used to initialize the next time of the auto-encoder training for faster learning.

In addition, Li et al. [34] introduced the Recurrent Evolution network based on Graph Convolution Network (RE-GCN) in 2021. This network learned the evolutional representation of entities and relations on each timestamp by cyclic modeling of knowledge graph sequences. A relation-aware GCN was specifically used for evolutionary units to capture the structural dependencies within the knowledge graph in each timestamp. To capture the sequence pattern of all information in parallel, the traditional knowledge graph sequence is automatically regressed and modeled by the gate loop component.

Graph neural networks (GNNs) are widely used in dynamic knowledge graphs currently [35−39]. However, these methods have limitations in model design, evaluation set, and training strategy. In 2022, You et al. [40] proposed a graph learning framework for the dynamic graph in view of the limitations, transformed static GNN into dynamic GNN, treated nodes embedding on different GNN layers as a hierarchical node state, and then updated it repeatedly over time.

These methods can create dynamic knowledge graphs, but they necessitate node information for the entire time period (including train and test sets), which are inapplicable to frequent change node sets, and do not consider the directionality of edges. Therefore, directional dynamic knowledge graph construction methods have become a research hotspot.

3. A dynamic knowledge graph construction method of risky vehicles

This section describes the dynamics of the risky vehicles triples, risk types, and the risky vehicle knowledge graph. The concept of the relational graph convolutional network is presented in Section 3.2. The model of the risky vehicle knowledge graph is introduced in Section 3.3.

3.1. Problem description

Risky vehicles triples. Assume that information about risky vehicles is recorded as triples $D^{+}=\{(h,r,t)\,|\,h\in E,r\in R,t\in E\}$ in a knowledge graph comprising *n* entities and *m* relations. Each triple is made up of the head entity *h* and tail entity *t* as well as the relation *r* between them, where *E* and *R* represent the relation set and the entity set, respectively. For instance, there are (Yue K72586, belong to, yangmou), (Yue K72586, vehicle type, a large truck) and (Yue K72586, risk type, illegal over-limit transportation more than 3 times in 1 year) and so on.

Risk types. To maintain road traffic order, prevent and reduce traffic accidents, protect personal and property safety, and legitimate rights and interests of citizens, legal persons, and other organizations, and improve traffic efficiency [41], the Road Traffic Safety Law of the People's Republic of China stipulates road vehicles. After statistical analysis, the risk types are summarized as the risks of the vehicle itself and the vehicle risks caused by the drivers, the latter is divided into direct and indirect types, in accordance with this regulation and the data related to the risk vehicles obtained from the Beijing transportation website.

The first type of risk is caused by the vehicle itself, specifically as follows. 1) Over-limit transportation. 2) Heavy goods vehicles are released after exceeding the standard loading. 3) No necessary measures were taken to prevent the goods from falling off and spreading them. 4) Driving a vehicle that has met the scrap standards on the road. 5) Speeding. 6) Driving a motor vehicle whose parts do not meet the technical standards shall leave the scene after a traffic accident. 7) The vehicle appears to be overloaded with driving. 8) Passenger vehicles other than highway passenger vehicles carry goods in violation of regulations.

The second type of vehicle risk is directly caused by the driver, listed as follows. 1) Driving over the limit without authorization. 2) Driving a motor vehicle while intoxicated. 3) Driving a motor vehicle after drinking alcohol. 4) Illegal passenger transport operation. 5) Motor vehicle that affects normal driving when changing lanes. 6) Parking in violation of prohibited line marking. 7) Unregistered motor vehicles on the road, driving three-wheeled motorcycles when the driver does not wear a safety helmet as required. 8) Driving a vehicle that is not compatible with the type of driving permit stated in the driver's license.

The third type of vehicle risk is indirectly caused by the driver, specifically as follows. 1) Drivers do not obtain the appropriate qualification documents, driving road freight transport vehicles. 2) Road transport employees do not carry the qualification documents. 3) The driver cannot provide a valid charter contract. 4) Drivers use canceled road transport operating permits to engage in road cargo transport operations. 5) Driving a motor vehicle caused a traffic accident and then fleeing, does not constitute a crime, the circumstances are less serious. 6) In violation of the provisions of road traffic safety laws and regulations, a major accident shall constitute a criminal act. 7) The driver caused a traffic accident and then fled, constituting a criminal offense. 8) Drivers do not hang number plates on the road or fail to install a motor vehicle number plate as required.

Dynamics of the knowledge graph for risky vehicles. The risk analysis of individual automobiles is not highly dynamic, and the type of risk mentioned above is typically static for a certain vehicle. For instance, the Huangzhuang Brigade of the Haidian Traffic Detachment in Beijing discovered the person Jia, whose license plate number is Beijing QXXXXX, was operating a road freight transport vehicle without the required qualification certificate at 15:30 on August 26, 2022. The knowledge graph created in this way is static, and time is assumed to be a risky vehicle attribute.

However, each entity represented by a risky vehicle has a time attribute from the entire risky vehicle dataset that indicates the moment the vehicle caused the risk (also called the timestamp). Based on the time range (April 2011−August 2022) of the risky vehicle dataset on urban roads in Beijing collected in this paper, it can be divided into 49 different time steps with an average interval of about 80 days, each contains a separately connected risky vehicles graph, and there are no edge connections between the risky vehicle graphs in different time steps. Obviously, the nodes in a given time step are associated with each other with very close timestamps, so that each node can be effectively considered as an instantaneous snapshot in time. The growth in overall information and the alteration in overall structure over time are dynamic reflections of the risky vehicle knowledge graph. The dynamics of the overall system can be converted into a dynamic knowledge graph using the knowledge graph.

For a dynamic knowledge graph *G*, at each time point *t*, *G* can be expressed as (A_t, X_t), where A_t and X_t represent the adjacency matrix and feature matrix, respectively, ultimately to learn is the node representation of each node at each time point in *G*.

3.2. The relational graph convolutional network

Kipf and Welling proposed the relational graph convolutional network in 2017. It is a unique method of graph representation that is frequently employed in the categorization of graph nodes, prediction of graph relations, social discovery, and network similarity [23]. The GCN consists of multilayer graph convolution. Although it is similar to the perceptron, it also needs an additional neighbor aggregation step activation convolution.

The simplest GCN is equivalent to a simple neural network and can be expressed as Eq (1).

$$f(H^{(l)}, A) = \sigma(AH^{(l)}W^{(l)}) \tag{1}$$

Where A is the adjacency matrix, H represents the feature matrix of nodes, W is the parameter matrix and the activation function is σ. The activation function is usually the sigmoid function. Directly employing the adjacency matrix will only calculate the feature-weighted sum of all neighbors for a node while the features of the node itself will be ignored. Generally, a unit matrix will be added. Additionally, if the adjacency matrix is not normalized, multiplying it with the feature matrix will alter the original distribution of the features, leading to unexpected issues. As a result, Eq (2) is the equation for the final layer feature propagation [23].

$$f(H^{(l)}, A) = \sigma(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \tag{2}$$

The type and direction of edges are not considered in the above GCN, but edges are oriented in the domain knowledge graph (such as traffic domain), which is solved by the relational graph convolutional network [42] proposed by Michael Schlichtkrull. The core formula is shown in Eq (3).

$$H_i^{(l+1)} = \sigma(\sum_{r\in R} \sum_{v_j \in N_{v_i}^{(r)}} \frac{1}{|N_{v_i}^{(r)}|} W_r^{(l)} h_j^{(l)} + W_o^{(l)} h_i^{(l)}) \tag{3}$$

Where R denotes the set of all relations in the graph, $N_{v_i}^{(r)}$ denotes the set of neighbors with relations r to the node v_i, W_r is the weight parameter corresponding to the neighbors with relations r, and W_o denotes the weight parameter corresponding to the node itself.

3.3. Dynamic knowledge graph model construction for risky vehicles

Although the weight update and evolution of the convolution cells are the most important in the dynamic knowledge graph construction model for risky vehicles, the premise is to construct the static knowledge graph for risky vehicles. In this paper, R-GCN is used for feature extraction of the risky vehicle knowledge graphs, and R-GCN parameters are updated through LSTM. The model for dynamic knowledge graph construction of risky vehicles is divided into two stages, as shown in Figure 1.

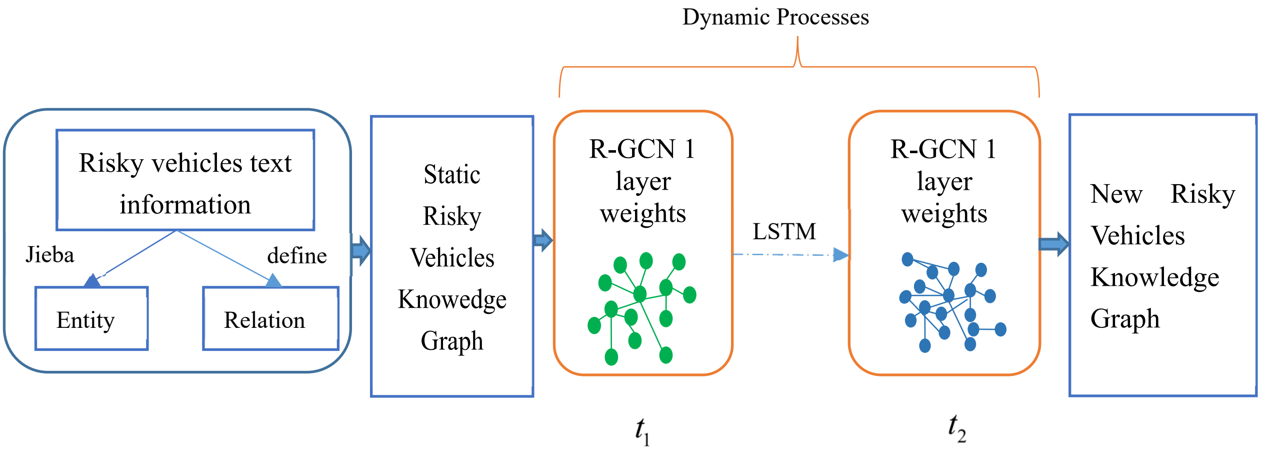
Dynamic Processes
Risky vehicles text information
Jieba
define
Entity
Relation
Static Risky Vehicles Knowedge Graph
R-GCN 1 layer weights
LSTM
R-GCN 1 layer weights
New Risky Vehicles Knowledge Graph
t_1
t_2

Figure 1. Dynamic knowledge graph construction model of risky vehicles.

1) The relevant text information of risky vehicles was obtained from some provinces and cities, and the entities (driver names, license plates, vehicle types, time, places, risk types) were extracted from the text through the jieba algorithm, and six basic relations were defined to build a static risky vehicle knowledge graph.

2) The combination of R-GCN and LSTM can study time-varying data. It is decided to employ a 3-layer network structure to prevent the issue of low accuracy brought on by small samples. Considering that the risky vehicles knowledge graph is mainly the structure of the graph, the LSTM

network was used for parameter update according to the [32].

It is possible to output the weights for each training R-GCN, which allows for observation analysis and a logical interpretation of the model-chosen weights. The memory of LSTM effectively utilizes the weight of the previous moment to update the R-GCN of the subsequent moment and realize the interaction of several R-GCN model moments.

The sigmoid function and tanh function are used throughout the construction to introduce non-linearity and ensure that the data do not diverge in the process of passing. Take the constructed static knowledge graph as input to the R-GCN. The initial R-GCN weights are obtained after the first round of training and updated the parameters of the R-GCN by the LSTM, which is a time-related dynamic process. The fusion algorithm of R-GCN and LSTM is shown in Algorithm 1.

```
Algorithm 1 The fusion algorithm of R-GCN and LSTM
Input: Nodes and edges
Output: Embed new nodes and edges
1: Nodes and edges vector quantization $\rightarrow (N, E)$ .
2: Model initial weight of training R-GCN  $\rightarrow W_{R-GCN}^{initial}$ .
3: The initial weight serves as the input of the LSTM, meter, and calculate the new weight. $\rightarrow W_{LSTM}^{new}$
4: The data is input into the R-GCN composed of new weights, the network, to obtain a new graph representation. $\rightarrow KG = f(W_{LSTM}^{new}, N, E)$
```

The weight update of the R-GCN network in Algorithm 1 is shown in Eq (4).

$$W_t^{(l)} = LSTM(W_{t-1}^{(l)}) \tag{4}$$

The weight of LSTM used to update the R-GCN is included in the equation. The equation was calculated in accordance with the literature [32].

4. Experimental setup and result analysis

This paper adopts the risky vehicle dataset, and SBM and Bitcoin Alpha serving as the comparison datasets. The risky vehicle data mainly come from the Public Security Traffic Management Bureau of Beijing Municipal Public Security Bureau. The risky vehicle dataset is from a real and reliable source, including data from 16 municipal districts such as Haidian District and Chaoyang District in Beijing. This data can be used not only to automatically identify the vehicle’s historical illegal information, but also to determine the potential risk of vehicles and risk level, and play an important role in early warning and control of urban road operation risk. The SBM dataset is synthetic data generated from a commonly used random graph model for simulating community structure and evolution following the literature [33]. Data from various trading platforms are used to create the Bitcoin Alpha dataset, which is a set of data that is traded using Bitcoin.

The data distribution of the SBM dataset and Bitcoin Alpha dataset is similar to that of the presented risky vehicle dataset, and the two datasets also have obvious dynamic features (time change), so the two datasets are selected as comparative datasets. The datasets are summarized in Table 1. The training, validation, and test datasets are divided according to the time steps. The time step depends on the data acquisition time interval, and the interval is a small-time step.

Table 1. Summary of dataset statistics.

	Nodes	Edges	Time step (Training/Verification/Test)
Risky vehicles	203,769	234,355	34/5/10
SBM	1000	4,870,863	35/5/10
Bitcoin Alpha	3777	24,173	95/13/28

In the experiment, the comparison models including GCN, DynGEM, ROLAND, and RE-GCN from the related work were chosen and compared on the link prediction. These methods are applicable to the construction of dynamic knowledge graphs, but they necessitate node information for the entire time period (including the training set and test set), which are not suitable for the frequent changes of node sets without considering the directionality of edges. Instead of just providing an optimal prediction result, the model will be required to use all the entities in the knowledge graph as candidates for the link prediction, so when choosing the evaluation criteria, generally choose the Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR).

The *MAP* value is the mean average precision, which is the average accuracy value for multiple validation sets, and the calculation equation is shown in Eq (5).

$$MAP=\frac{1}{|Q|}\sum_{q\in Q} AveP(q) \tag{5}$$

In the equation, $AveP$ is the average accuracy, Q represents the number of validation sets, MAP is the value required in this paper.

MRR is to evaluate the performance of the link prediction by ranking the correctly predicted result values in the predicted results, and the calculation equation is shown in Eq (6).

$$MRR=\frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \tag{6}$$

Where Q is the number of validation sets, $rank_i$ indicates the rank at the i .

This paper evaluates the results on the link prediction task, uses the information before it at time *t* to predict whether an edge exists at time *t* + 1. Since the historical information has been encoded in the R-GCN parameters, this paper performs edge prediction based on the head-to-tail entities. The results are shown in Table 2.

Table 2. MAP and MRR values of different models.

	MAP Risky vehicles	SBM	Bitcoin Alpha	MRR Risky vehicles	SBM	Bitcoin Alpha
GCN	0.0724	0.1987	0.0003	0.0017	0.0138	0.0031
DynGEM	0.0948	0.1680	0.0525	0.0103	0.0139	0.1287
ROLAND	0.1218	0.0012	**0.0962**	0.1036	0.0011	**0.2887**
RE-GCN	0.11	0.1873	0.0931	0.009	0.0014	0.0756
R-GCN + LSTM (this paper)	**0.2746**	**0.2019**	0.0023	**0.1075**	**0.0146**	0.0864

Table 2 gives the results of the present method compared with GCN, DynGEM, ROLAND, RE-GCN. On the risky vehicle datasets, the current method outperforms the other four methods, improving 0.2022, 0.1798, 0.1528 and 0.1646. It is mainly because the comparison methods are not conducive to processing the directional data. On the SBM dataset, the R-GCN+LSTM method is still better than the other four methods, mainly due to enough edge information. However, on this dataset, the ROLAND method performs extremely poorly, mainly because the method repeatedly updates the node embedding representation over time and does not take time as the entity's attribute value. However, on the Bitcoin Alpha dataset, the method is not dominant, and the accuracy is lower, but higher than the GCN method alone, mainly because the edges are not informative and not directional.

From the perspective of time complexity, the time complexity of the presented method is determined by R-GCN and LSTM, which is $O(n^2)$, where n is the size of the input layer. GCN needs to decompose the Laplace matrix and calculates the matrix multiplication in each forward propagation process. When the graph is large, the time complexity is $O(n^2)$, and n represents the number of nodes in the knowledge graph, which is very time-consuming. DynGEM uses a dynamically expandable self-coder to maintain the network structure characteristics and handle the changing network scale, so the time complexity of DynGEM is $O(n^2)$ the same as that of the self-coder, and n represents the number of nodes in the knowledge graph. ROLAND is to reuse static GNN for dynamic graph settings, and its time complexity is mainly affected by GNN time complexity, which is $O(m)$. RE-GCN learns the evolutionary representation of entities and relationships in each time stamp by modeling the knowledge graph sequence circularly. For each evolution unit, the GCN of relationship perception is used to capture the structural dependency in the knowledge graph in each time stamp. Therefore, the time complexity is $O(m(|E|\omega+|R|D)+|E^s|)$, where $|E|$ represents the number of entities in the time stamp m, $|E^s|$ is the number of entities in the static knowledge graph, $|R|$ represents the number of relations, and ω represents the number of layers.

To further validate the effectiveness of the present method, by performing classification experiments in the risk vehicle dataset, the paper evaluated them using accuracy, precision, recall, and F value. The results of accuracy, precision, recall, and F values are shown in Figure 2.

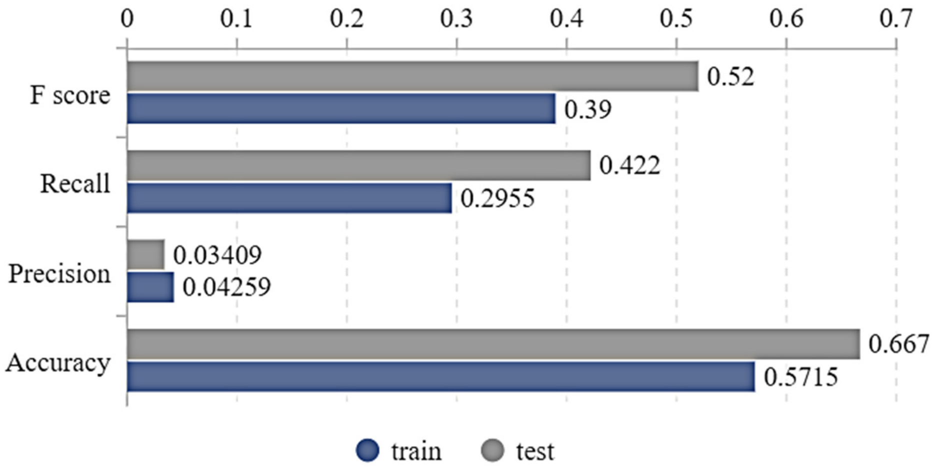

Figure 2. Results for accuracy, precision, recall, and F values.

Figure 2 shows the accuracy of the method is not high in the classification task on the risky vehicle dataset, mainly because the collected dataset contains a large amount of data belonging to the first type of risk, that is, the data of the risk of the vehicle itself accounts for about 70% of the total data, while the second type of risk data accounts for about 25%, and the third type of risk data accounts for about 5%, which leads to inaccurate prediction of the second and third types of risk. Secondly, in different time steps, the data of the three risk types are unevenly distributed. Finally, the distinction between the specific types of the three types is not obvious, for example, “driving a motor vehicle whose parts do not meet the technical standards, leaving the scene after a traffic accident” in the first type of risk, and “escaping after causing a traffic accident, constituting a criminal act” in the third type of risk is not obvious.

The change of the loss values for training and testing in the experiment with the number of iterations is shown in Figure 3.

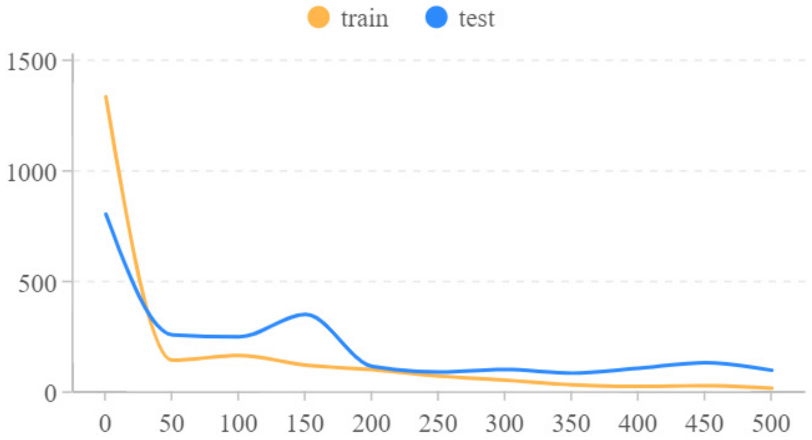

Figure 3. Loss values of training and testing with the number of iterations.

The variation of the loss value with the number of iterations during training and testing is shown in Figure 3. From Figure 3, it can be seen the loss value decreases with the increase of the number of iterations in both training and testing, and the loss value decreases very fast until 50 iterations. After 50 iterations, the decrease slows down and gradually leveled off after 200 iterations. After 50 iterations, the loss value of the test is higher than the loss value of the training, mainly when new data appear in the test dataset.

The large-scale knowledge graph benefits from the time-aware knowledge representation learning method outlined in related work. These methods are not useful because the risky vehicle dataset in this paper is small and the edge information is sparse. Compared with the methods based on matrix decomposition and random walk, the presented method plays an important role in dealing with the knowledge graph whose structure changes greatly with time, and the node information changes little. However, the two methods are aimed at the change of node information namely the feature vector, so it is not suitable for the dynamic knowledge graph construction of risky vehicles.

5. Conclusions

This paper explores a dynamic knowledge graph construction method for urban road risky vehicles. This method combines the relational graph convolutional network with LSTM to address the issue of dynamic and edge orientation of the knowledge graph for urban road risky vehicles. In this method, the relational graph convolutional network solves directionality, LSTM is used to realize the dynamics. The structure and implementation of the method are described. Link prediction experiments were performed on three datasets including risky vehicles, the SBM, and the Bitcoin Alpha. The results show the presented method is better for the dynamic construction of directional knowledge graphs compared with GCN, DynGEM, ROLAND, and RE-GCN methods.

The data distribution of three risk types in the risky vehicle dataset in this paper is unbalanced, and the application scope of the proposed method is small. Therefore, in future work, we will increase the risky vehicle dataset by gathering information on traffic accidents and other factors, and improve the method to accurately analyze and forecast risky vehicles.

Acknowledgments

This work is supported by National Natural Science Fund of China (61371143).

Conflict of interest

The authors declare there is no conflict of interest.

References

1. Made in China 2025, Issued by the State Council.
2. Y. C. Zou, *Construction and Application of Traffic Knowledge Graph Based on Multi-source Data Fusion (in Chinese)*, Master's thesis, Dalian University of Technology, 2020. Available from: https://cdmd.cnki.com.cn/Article/CDMD-10141-1020653436.htm.

3. L. Wang, *Research on Intelligent Knowledge Support of Urban Rail Transit Construction Safety Management Based on Knowledge Graph (in Chinese)*, Ph.D thesis, China University of Mining and Technology, 2019. https://doi.org/10.27623/d.cnki.gzkyu.2019.000408
4. G. L. Zhou, *Research on Prediction of Urban Traffic Congestion Area Based on Knowledge Graph and Deep Learning (in Chinese)*, Master's thesis, University of Science and Technology of China, 2019. Available from: https://kns.cnki.net/kcms2/article/abstract?v=3uoqIhG8C475KOm_zrgu4lQARvep2SAkOsSuGHvNoCRcTRpJSuXuqaqG2zp1ftApp1d23kvjOO2oSeVFvORibKCV9PhY0Iws&uniplatform=NZKPT&src=copy.
5. M. Zhou, *Study on Vehicle Lane Change Risk Situation Based on Driving Behavior and Driving Style Inclination (in Chinese)*, Chang'an University, 2022. https://doi.org/10.26976/d.cnki.gchau.2022.000496
6. S. Pantangi, G. Fountas, P. Anastasopoulos, J. Pierowicz, K. Majka, A. Blatt, Do High Visibility Enforcement programs affect aggressive driving behavior? An empirical analysis using Naturalistic Driving Study data, *Accid. Anal. Prev.*, **138** (2020), 105361. https://doi.org/10.1016/j.aap.2019.105361
7. P. Sârbescu, A. Rusu, Personality predictors of speeding: anger-aggression and impulsive-Sensation Seeking. A systematic review and meta-analysis, *J. Saf. Res.*, **77** (2021), 86−98. https://doi.org/10.1016/j.jsr.2021.02.004
8. J. Kovaceva, I. Isaksson-Hellman, N. Murgovski, Identification of aggressive driving from naturalistic data in car-following situations, *J. Saf. Res.*, **73** (2020), 225−234. https://doi.org/10.1016/j.jsr.2020.03.003
9. Y. Xu, S. Bao, A. K. Pradhan, Modeling drivers' reaction when being tailgated: a random forests method, *J. Saf. Res.*, **78** (2021), 28−35. https://doi.org/10.1016/j.jsr.2021.05.004
10. Y. Dai, *Research and Application of Risk Assessment of Dangerous Goods Road Transport Vehicles Based on Hierarchical Fuzzy Network Model (in Chinese)*, Chang'an University, 2022. https://doi.org/10.26976/d.cnki.gchau.2022.002093
11. Q. Yang, *Research on the Risk Analysis Method of Lane Change on Expressway Vehicles (in Chinese)*, The People's Public Security University of China, 2022. https://doi.org/10.27634/d.cnki.gzrgu. 2022.000006
12. K. Han, *Research on the Video Detection Method for Violations of not Driving in the Guide Lane (in Chinese)*, Nanjing University of Science and Technology, 2019. https://doi.org/10.27241/d.cnki.gnjgu.2019.001552
13. Y. N. Cui, J. Li, L. Shen, Y. Shen, L. Qiao, J. Bo, Duration-HyTE: A Time-aware knowledge representation learning method based on duration modeling, *J. Comput. Res. Dev.*, **57** (2020), 1239–1251. https://doi.org/10.7544/issn1000-1239.2020.20190253
14. M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, in *Advances in Neural Information Processing Systems*, 2002. Available from: https://papers.nips.cc/paper/2001/file/f106b7f99d2cb30c3db1c3cc0fde9ccb-Paper.pdf.
15. S. T. Roweis, L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, *Science*, **290** (2000), 2323–2326. https://doi.org/10.1126/science.290.5500.2323
16. J. Li, H. Dani, X. Hu, J. Tang, Y. Chang, H. Liu, Attributed network embedding for learning in a dynamic environment, in *Proceedings of the 2017 ACM Conference on Information and Knowledge Management*, (2017), 387−396. https://doi.org/10.1145/3132847.3132919

17. G. H. Nguyen, J. B. Lee, R. A. Rossi, N. K. Ahmed, E. Koh, S. Kim, Continuous-time dynamic network embeddings, in *Companion Proceedings of the Web Conference 2018*, (2018), 969−976. https://doi.org/10.1145/3184558.3191526
18. W. Yu, W. Cheng, C. Aggarwal, K. Zhang, H. Chen, W. Wang, NetWalk: A flexible deep embedding approach for anomaly detection in dynamic networks, in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, (2018), 2672−2681. https://doi.org/10.1145/3219819.3220024
19. R. Trivedi, M. Farajtabar, P. Biswal, H. Zha, Representation learning over dynamic graphs, preprint, arXiv:1803.04051.
20. Y. Zuo, G. Liu, H. Lin, J. Guo, X. Hu, J. Wu, Embedding temporal network via neighborhood formation, in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, (2018), 2857–2866. https://doi.org/10.1145/3219819.3220054
21. B. Lyu, Y. Yang, S. Wen, T. Huang, K. Li, Neural architecture search for portrait parsing, *IEEE Trans. Neural Networks Learn. Syst.*, **34** (2023), 1112−1121. https://doi.org/10.1109/TNNLS.2021.3104872
22. W. Li, S. Wen, K. Shi, Y. Yang, T. Huang, Neural architecture search with a lightweight transformer for text-to-image synthesis, *IEEE Trans. Network Sci. Eng.*, **9** (2022), 1567−1576. https://doi.org/10.1109/TNSE.2022.3147787
23. W. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, preprint, arXiv:1609.02907.
24. B. Lyu, M. Hamdi, Y. Yang, Y. Cao, Z. Yan, K. Li, et al., Efficient spectral graph convolutional network deployment on memristive crossbars, *IEEE Trans. Emerging Top. Comput. Intell.*, **7** (2022), 415−425. https://doi.org/10.1109/TETCI.2022.3210998
25. B. Lyu, S. Wen, K. Shi, T. Huang, Multiobjective reinforcement learning-based neural architecture search for efficient portrait parsing, *IEEE Trans. Cybern.*, **53** (2021), 1158−1169. https://doi.org/10.1109/TCYB.2021.3104866
26. J. Zhu, X. Han, H. Deng, C. Tao, L. Zhao, P. Wang, et al., KST-GCN: A knowledge-driven spatial-temporal graph convolutional network for traffic forecasting, *IEEE Trans. Intell. Transp. Syst.*, **23** (2022), 15055–15065. https://doi.org/10.1109/TITS.2021.3136287
27. Y. Fu, S. Okada, L. Wang, L. Guo, Y. Song, J. Liu, et al., CONSK-GCN: Conversational semantic- and knowledge-oriented graph convolutional network for multimodal emotion recognition, in *2021 IEEE International Conference on Multimedia and Expo (ICME)*, (2021), 1–6. https://doi.org/10.1109/ICME51207.2021.9428438
28. J. Yang, W. Zhou, L. Wei, J. Lin, J. Han, S. Hu, RE-GCN: Relation enhanced graph convolutional network for entity alignment in heterogeneous knowledge graphs, in *Database Systems for Advanced Applications*, **12113** (2020), 432–447. https://doi.org/10.1007/978-3-030-59416-9_26
29. Z. Liu, Z. D. Jiang, F. Wei, OD-GCN object detection by knowledge graph with GCN, preprint, arXiv:1908.04385v1.
30. L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, et al., T-GCN: A temporal graph convolutional network for traffic prediction, *IEEE Trans. Intell. Transp. Syst.*, **21** (2020), 3848–3858. https://doi.org/10.1109/TITS.2019.2935152.

31. S. Abu-El-Haija, A. Kapoor, B. Perozzi, J. Lee, N-GCN: Multi-scale graph convolution for semi-supervised node classification, in *Proceedings of the 35th Uncertainty in Artificial Intelligence Conference*, **115** (2020), 841–851. Available from: http://proceedings.mlr.press/v115/abu-el-haija20a.html.
32. A. Pareja, G. Domeniconi, J. Chen, T. Ma, T. Suzumura, H. Kanezashi, et al., EvolveGCN: Evolving graph convolutional networks for dynamic graphs, in *Proceedings of the AAAI Conference on Artificial Intelligence*, **34** (2020), 5363−5370. https://doi.org/10.1609/aaai.v34i04.5984
33. P. Goyal, N. Kamra, X. He, Y. Liu, DynGEM: Deep embedding method for dynamic graphs, preprint, arXiv:1805.11273.
34. Z. Li, X. Jin, W. Li, S. Guan, J. Guo, H. Shen, et al., Temporal knowledge graph reasoning based on evolutional representation learning, in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, (2021), 408–417. https://doi.org/10.1145/3404835.3462963
35. J. Li, Z. Han, H. Cheng, J. Su, Predicting path failure in time-evolving graphs, in *the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, (2019), 1279–1289. https://doi.org/10.1145/3292500.3330847
36. H. Peng, H. Wang, B. Du, M. Z. A. Bhuiyan, H. Ma, J. Liu, et al., Spatial temporal incidence dynamic graph neural networks for traffic flow forecasting, *Inf. Sci.*, **521** (2020), 277–290. https://doi.org/10.1016/j.ins.2020.01.043
37. A. Taheri, K. Gimpel, T. Berger-Wolf, Learning to represent the evolution of dynamic graphs with recurrent models, in *Companion Proceedings of the 2019 World Wide Web Conference*, (2019), 301−307. https://doi.org/10.1145/3308560.3316581
38. X. Wang, Y. Ma, Y. Wang, W. Jin, X. Wang, J. Tang, et al., Traffic flow prediction via spatial temporal graph neural network, in *Proceedings of the Web Conference 2020*, (2020), 1082−1092. https://doi.org/10.1145/3366423.3380186
39. B. Yu, H. Yin, Z. Zhu, Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting, in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2018. Available from: https://www.ijcai.org/Proceedings/2018/0505.pdf.
40. J. You, T. Du, J. Leskovec, ROLAND: Graph learning framework for dynamic graphs, in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, (2022), 2358–2366. https://doi.org/10.1145/3534678.3539300
41. Road Traffic Safety Law of the People's Republic of China, The fifth meeting of the Standing Committee of the Tenth National People's Congress.
42. M. S. S. Thomas, N. Kipf, P. Bloem, R. van den Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, in *The Semantic Web*, **10843** (2018), 593–607. https://doi.org/10.1007/978-3-319-93417-4_38

AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0).