Australasian Transport Research Forum 2022 Proceedings
28-30 September, Adelaide, Australia
Publication website: http://www.atrf.info

Online Traffic Incident Prediction with Hybrid Graph-based Neural Network and Continual Learning

Thanh Tran[1], Dan He[1], Jiwon Kim[1], and Mark Hickman[1]
[1]School of Civil Engineering, The University of Queensland, St. Lucia, QLD 4072
Email for correspondence: d.he@uq.edu.au

1. Introduction

Driven by the prevalence of different data collection techniques, e.g., loop detectors, and GPS devices, a massive amount of traffic data have been captured over many years. This has led to an active development of data-driven models, particularly machine learning (ML) models that learn patterns from historical data. Traffic data collected over a long period often reveal a long-term evolution of traffic patterns and topological changes in road networks. This has an important implication that ML models developed and trained with traffic data at one point would need to continue to evolve over time to capture changes in traffic data. One of the key application areas of ML in traffic networks is real-time traffic incident prediction, which aims to predict the probability of incident occurrence within a specific region or road segment ahead of time to enable proactive traffic and incident management for preventing congestion. Generally, ML-based incident prediction models are capable of unveiling patterns in traffic network conditions, driver behaviours, and traffic flow dynamics that are associated with incident occurrences by mapping large amounts of incident records to traffic and driver-related data. The prediction results can be used to infer the potential impact or risk of incidents, monitor safety responses, and inform disruption management strategies. However, with the continuous changes in traffic data, a pre-trained ML model might be inadaptable to new traffic patterns, resulting in poor prediction performance. Additionally, frequent retraining and updating of the online model are inefficient and unnecessary when the traffic patterns remain stable. Consequently, solutions to efficiently capture patterns from new data while consolidating historical information are desirable for real-time incident prediction.

In the literature, many ML methods have been applied in traffic incident prediction models. Recently, deep learning models have gained popularity, with its ability to capture non-linear, spatial and temporal correlations in traffic data more accurately. In particular, the discriminative power of graph neural networks (GNNs) has attracted the attention of transport researchers as road networks are essentially graphs and the relationships among road links can be effectively modelled using graph structure and connectivity. As such, many recent incident prediction models have used GNN-based methods: they are leveraged to extract spatial correlation from the non-Euclidean data and can be used with sequence models like RNNs mining temporal trends of traffic incidents. For instance, Yu et al. (2021) addressed the link-level accident prediction problem by proposing a graph-based model to predict link-level incident risk by learning spatial-temporal, external features from a graph that represents a road network. Wang et al. (2021) proposed a region-wide accident risk prediction model named GSNet to capture the geographical and semantic aspects from undirected graphs that represent different characteristics of a road network. In our previous work, we have proposed a Hybrid Graph-based Neural Network (HGNN) (Tran, et al., 2021), which is specifically designed to address the challenge of incorporating multiple heterogeneous data sources in traffic incident prediction by taking as input multiple graphs with different structures representing different data sources.

However, the existing studies fail to consider the issue of implementing the developed model over a long period, where the model should adapt to long-term pattern changes in data (e.g., flow pattern change, addition/deletion of sensors, driver behaviour change) efficiently and effectively. A naïve solution is to retain the models following a fixed schedule (e.g., weekly, monthly or yearly) to capture new patterns. It is, however, difficult to manually find a right schedule; it would be inefficient to retrain large-scale models too frequently due to high computation costs, while retraining models too sparsely could miss important pattern changes that happen between retraining points. Some studies have proposed more automated retraining schemes, where input features are monitored during online implementation and retraining is recommended when significant changes occur in any of those features. This approach also has limitations that it is complex and time-consuming to analyse thousands of features in large-scale models and, more importantly, some of the significant changes in input features may not have a direct impact on the output. Therefore, it is desirable to concentrate on important patterns that are likely to impact the output (incident occurrence), rather than blindly detecting all the changes in input features. This idea has been investigated in the recent literature. Chen et al. (2021) developed a streaming traffic forecasting model based on Continual Learning (CL). Additionally, an incremental clustering method is developed by Zhao et al. (2021) to identify patterns and outliers in flight data while incrementally updating its clusters as new data come in online.

Motivated by this, we propose an online traffic incident prediction model with Continual Learning (CL) based on our previous proposed Hybrid Graph-based Neural Network (HGNN), which aims to detect pattern changes automatically from data streams as a trigger of model retraining rather than retraining in a fixed schedule. This detection mechanism selectively learns from only new data with new patterns while keeping stable patterns from old data. As a result, the model can effectively and efficiently update its knowledge on new patterns while consolidating the knowledge learned previously rather than using all data for retraining from scratch. To further justify the motivation of our proposal and study the effectiveness of consolidation, in this paper, we conduct a set of experiments on HGNN with streaming traffic data across several years. Firstly, we train and test our model using traffic data in different years, where the traffic patterns are potentially different, to show how the performance is influenced by the evolution of traffic patterns. Moreover, to consolidate historical traffic knowledge and transfer it to the current model for better prediction, we adopt strategies of parameter smoothing from data and model perspectives. The results of our experiments demonstrate the effectiveness of consolidation and the potential of detecting changes in traffic patterns with high efficiency over the long-term period, which will be addressed in the full paper version.

2. Framework Overview

In our work, we aim to develop an online framework that can support real-time incident prediction with continual learning. Previously, we have developed a data-driven model, named HGNN that predicts the occurrence of traffic incidents within a given sub-area during a future time interval (e.g., next 15, 30, and 60 minutes) by flexibly merging different data sources capturing various aspects of the area's road conditions. In particular, HGNN, which is a network-wide model, takes the spatio-temporal information over a certain period (e.g., 30 minutes) backing from the prediction time stamp for any given sub-area as input and outputs the incident probability of that area by training with data from randomly sampled sub-areas within the whole study network. We consider different types of features, including traffic features (e.g., flow, occupancy, speed), link features (e.g., length of link), and temporal features (e.g., time of day, holiday), where the underlying network structure might be different. As a result, graph neural networks are used to encode the correlations from network structures, and different features for incident prediction. To support real-time prediction, at this stage, we train HGNN based on historical data from previous years and make use of the trained model to predict real-time incident risk for any given sub-area. However, it has significant challenges in dealing with pattern changes from complex inputs over a long-term period thereby producing more false alarms and being low efficient. As a result, our current goal is to develop an online framework where incident predictors can effectively learn new and old data with significantly reduced processing time and memory usage.

Our framework mainly consists of two components. The first component concentrates on efficiently consolidating historical knowledge. Specifically, this part contains two main modules including Information Replay or Rehearsal and Regularization Module. Information Replay helps the trained model to remember old patterns by learning from a small proportion of all of the old data. The regularization module leverages the loss function using the loss term to help consolidate knowledge in the learning process for new patterns and retain existing knowledge. As for the second component, we aim to build a module that automatically detects the changes in traffic pattern from newly captured data and make use of the new patterns for model retrain. A straightforward way of detecting the pattern changes is to measure the differences in the embedded features between historical and new representations learnt by our original model. However, our model is only able to capture pattern changes at the sub-area level. Thus, we have to tackle this issue by introducing a module that can detect the pattern changes dynamically and globally within the study network. For the rest of the paper, we first introduce the methodology of the first component, and then we show some experimental results correspondingly. The effectiveness of the first component and the potential solution for the second component will be discussed at the end of this paper.

3. Historical Data Knowledge Consolidation

Intuitively, the historical data patterns could help predict the current incident probability as similar cases might occur years apart. However, the existing models may forget previous knowledge it has learned when the patterns from new data are captured by these models, since they concentrate on those new patterns only. This phenomenon is usually defined as *catastrophic forgetting*. To mitigate such a phenomenon, a common way is to track back to the sources where this knowledge is from by retaining the model with both historical and current captured data. Rather than making use of all the historical data resulting in high computational cost, in our work, we introduce an *information replay module* that only considers a small part of the historical data while capturing knowledge from old patterns. On the other hand, the *regularization module* is introduced to consolidate data knowledge from model perspectives by preserving important parameters.

3.1 Information Replay Module

One straightforward way to alleviate catastrophic forgetting is to make use of a portion of the old data to be interleaved with the data from the current task, so that the graph-based model does not forget existing knowledge. In our work, to consider the trade-off between efficiency and effectiveness, we perform a simple strategy where only a limited number of historical data (i.e., 5%) is used for model retraining as a replay so that existing knowledge is maintained.

3.2 Regularization Module

Let $\tau \in \{1, 2, \dots, \mathcal{T}\}$ be a long time interval that the traffic patterns and network structure remine stable within τ. Assume that for each τ, we have a trained model Ψ_τ ideally capturing the patterns from $1, 2, \dots, \tau$. Thus we have a series of models $\{\Psi_1, \Psi_2, \dots, \Psi_\mathcal{T}\}$ over time. Each time to retrain the current model Ψ_τ, the regularization module forces the current training model to remember useful knowledge from the previously trained model $\Psi_{\tau-1}$. In order to preserve previous knowledge, we adopt the strategy elastic weight consolidation (EWC) derived from Kirkpatrick, et al., (2017), which adds a exquisite smoothing term to the loss fuction, formulated as follows.

$$\mathcal{L}_s = \lambda \sum_i \mathbf{F}_i(\Psi_\tau(i) - \Psi_{\tau-1}(i))^2, \tag{1}$$

Here λ is the weight of the smoothing term and $\mathbf{F}_i$ is the importance of the i-th parameter in the model, $\Psi_{\tau-1}$, which is estimated using Fisher Information as follows:

$$\mathbf{F} = \frac{1}{|X_{\tau-1}|} \sum_{x \in X_{\tau-1}} (\, g(\Psi_{\tau-1}\,;\, x)\, g(\Psi_{\tau-1}\,;\, x)^{\mathrm{T}}\,), \tag{2}$$

where g is the first-order derivative of the loss and X denotes the input featuers. Particularly, in Equation (1), the weight of the less essential parameters for historical trained model is smaller, and these parameters can better adapt to the new patterns. On the other hand, the more important parameters for historical model have higher weights, ensuring the preservation of historical knowledge.

4. Preliminary Results & Discussion

To verify the effectiveness and efficiency of our framework, we conduct experiments on a real-world dataset. Particularly, we use incident data from Brisbane, Australia. We select two-year data from 2017 and 2020 respectively based on the availability of data. And more importantly, the traffic patterns in these two years are potentially different since the ones in 2020 might be influenced by the impact of COVID-19.

4.1 Experimental Settings

In our experiment, our model is evaluated by five metrics: AUC (Area Under the Curve), Acc (Accuracy), Precision, Recall and F1 (F1-score), which are formulated as follows (TP:True positive; TN:True negative; FP:False positive; FN:False negative).

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}, \qquad \mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}},$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \qquad \mathrm{F1} = \frac{2 * \mathrm{Precision} * \mathrm{Recall}}{\mathrm{Recall} + \mathrm{Precision}}$$

We adopt the same traffic data sources, methods to construct features, and parameter setting in our previous work, i.e., HGNN (Tran, et al., 2021). Following settings in our previous work, data for each year contains 1600 generated samples and is split into training, validation and testing sets with a ratio of 60%, 20%, and 20%. Our task is to predict whether there will be an incident in the following 15 minutes by making use of up to 30 minutes of historical traffic data. Please refer to HGNN (Tran, et al., 2021) for more details on data preprocessing and feature extraction. Adam is used as the optimizer with a learning rate of 0.001. The batch size is 32 and the total number of epochs is 100 for each year with early stopping to accelerate the training process. The random search strategy is applied to find the best hyperparameters for the models.

4.2 Prediction Performance

First of all, we train and test HGNN on data captured in 2017 to obtain a trained model, denoted by HGNN-17 and the corresponding performance results are illustrated in Table 1, which is in line with the performance shown in the original paper.

Table 1: Prediction Performance of trained model based on 2017 data.

Models	Test data	AUC	Acc	F1	Precision	Recall
HGNN-17	2017	0.9118	86.25	0.8650	0.8493	0.8812

Then, HGNN-17 is updated by retraining with training data from 2020 incorporating knowledge consolidation (information replay and regularization) strategies before being tested using data from 2017 and 2020 respectively. For comparison, we also retain our model with training data from 2020 while no knowledge consolidation technique is applied. We denote the model with knowledge consolidation as HGNN-17(KC) and the one without knowledge consolidation as HGNN-17(NC). Table 2 shows the performance results of the retrained models. As can be seen in Table 2, after retraining the model with new data (from 2020), the performance of the models HGNN-17(NC) and HGNN-17(KC) testing on data from 2017 is worse than the ones shown in Table 1, while incorporating the knowledge consolidation strategies in model HGNN-17(KC) helps to improve the performance to some extend, e.g., the AUC increases from 0.6311 to 0.6869. Considering the performance of models testing on data from 2020. The performance of model HGNN-17(NC) is worse than HGNN-17(KC) in all metrics, which shows the effectiveness of the knowledge consolidation. Indeed, the information replay keeps some historical data while preventing the scenario where all the historical data is used with memory constraints. The regularization module helps to penalize changes in the most important weights for the previous model based on the computation of the importance of each weight (fisher information) and a squared regularization loss while not using any of the previous data.

Table 2: Prediction Performance of HGNN-17 after using training data from 2020 data.

Models	Test data	AUC	Acc	F1	Precision	Recall
HGNN-17(NC)	2017	0.6311	58.12	0.4682	0.6413	0.3687
HGNN-17(NC)	2020	0.9287	86.88	0.877	0.8206	0.9437
HGNN-17(KC)	2017	**0.6869**	**64.69**	**0.5603**	**0.7422**	**0.4500**
HGNN-17(KC)	2020	**0.9437**	**89.06**	**0.8979**	**0.8415**	**0.9625**

While Table 2 shows experiments where HGNN-17 is retrained or updated by using only data from a single year (i.e., 2020), Table 3 describes the scenario where HGNN-17 is retrained by using all training data from both years (i.e., 2017 and 2020), denoted by HGNN-17(A). Particularly, in Table 3, the results show that model HGNN-17(A) performs not too bad when testing each year separately. However, comparing the ones training with single-year data (HGNN-17, HGNN-17(NC) and HGNN-17(KC)) the performance drops slightly, which is interesting. The potential reason could be the combination of traffic patterns forces the model to be more adaptable to various conditions while losing the pertinency for specific patterns.

Table 3: Prediction Performance of HGNN-17 after using training data from 2017 and 2020 data.

Models	Test data	AUC	Acc	F1	Precision	Recall
HGNN-17(A)	2017	0.9041	86.25	0.8650	0.8670	0.8562
HGNN-17(A)	2020	0.8986	85.31	0.8668	0.7927	0.9562

4.3 Discussion

From the preliminary experimental results, we observe that training the model with data containing different patterns can induce a decrease of performance, which shows the evidence of the catastrophic forgetting and highlights the motivation of our work to develop an online prediction model with continual learning. On the other hand, the results also demonstrate the superiority of the online model with knowledge consolidation (information replay and regularization) strategies compared to the existing offline models since given running time and memory constraints, experiments show better performance of the proposed online framework in dealing with dynamically growing data. Based on the promising preliminary results, our next step is to extend the model by introducing a module that is capable of detecting the pattern changes automatically, from which we retrain our model making use of the new patterns while consolidating the knowledge from the historical data.

5. Acknowledgements

The authors gratefully acknowledge the support of Australian Research Council (ARC), Queensland Department of Transport and Main Roads, and Transmax Pty Ltd under the ARC Linkage Project on Real-time Analytics for Traffic Management (LP180100018).

Reference

Chen, X., Wang, J. & Xie, K., 2021. TrafficStream: A Streaming Traffic Flow Forecasting Framework Based on Graph Neural Networks and Continual Learning. *IJCAI.*

Hossain, M. & Muromachi, Y., 2012. A Bayesian network based framework for real-time crash prediction on the basic freeway segments of urban expressways. *Accident Analysis & Prevention.*

Kirkpatrick, J. et al., 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences.*

Knyazev, B., Taylor, G. & Amer, M., 2019. Understanding attention and generalization in graph neural networks. *Advances in neural information processing systems.*

Lv, Y., Tang, S. & Zhao, H., 2009. Real-time highway traffic accident prediction based on the k-nearest neighbor method. *ICMTMA.*

Ren, H. et al., 2018. *A Deep Learning Approach to the Citywide Traffic Accident Risk.* s.l.:s.n.

Tran, T., He, D., Kim, J. & Hickman, M., 2021. *Data-driven Traffic Incident Prediction with Hybrid Graph-based Neural Network.* s.l.:Australasian Transport Research Forum.

Wang, B., Lin, Y., Guo, S. & Wan, H., 2021. Gsnet: Learning spatial-temporal correlations from geographical and semantic aspects for traffic accident risk forecasting. *AAAI.*

Xu, K. W. H. J. L. a. S. J., 2018. How powerful are graph neural networks?.

Yuan, Z., Zhou, X. & Yang, T., 2018. *Hetero-convlstm: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data.* s.l.:SIGKDD.

Yu, L. e. a., 2021. *Deep spatio-temporal graph convolutional network for traffic accident prediction.* s.l.:Neurocomputing.

Zhao, W., Li, L., Alam, S. & Wang, Y., 2021. An incremental clustering method for anomaly detection in flight data. *Transportation Research Part C: Emerging Technologies.*