Real-time traffic flow prediction using Big Data analytics

Dang-Khoa Tran[1,2], Dinh-Quang Hoang[1,2], Viet-Thang Le[1,2], Minh-Duc Nguyen Thai[1,2] and Trong-Hop Do[1,2]

[1] University of Information Technology, Ho Chi Minh City, Vietnam
[2] Vietnam National University, Ho Chi Minh City, Vietnam
{18520936,18521294, 18520356, 18520267}@gm.uit.edu.vn, hopdt@uit.edu.vn

Abstract. Traffic congestion is always a big problem to be solved in the world because of its negative effects. There are many ways to solve traffic congestion based on its causes. And one of the important factors needed to reduce traffic congestion is the control of traffic on the road at a reasonable level. There have been many algorithms proposed to deal with this problem. However, any algorithm has its own limitations such as slow processing time, no timeliness, complicated implementation methods, etc. In recent years, with the explosion of data, many tools and methods are also developed to handle big data. This has opened up many more solutions for traffic-related problems. In this paper, a real-time traffic prediction system is proposed with high accuracy, simple method and vivid visualization. The performance of the proposed system is verified through experiments results.

Keywords: real-time traffic flow · spark streaming · big data analytics.

1 Introduction

Our world is developing more and more, along with the rapid development of large cities, which has increased the number of people and vehicles in these urban areas. This has caused a very serious traffic congestion. The negative effects of traffic congestion on our lives are immense. It is estimated that Ho Chi Minh City loses about 1.2 million working hours each year, 1.3 billion USD/year due to traffic congestion and 2.3 billion USD due to environmental pollution from motor vehicles. According to an assessment from the Institute of Transport Strategy and Development (Ministry of Transport) also said that congestion causes damage to Hanoi every year about 1-1.2 billion USD. That's just statistical data in 2 cities in Vietnam. In addition, traffic congestion was also negative impacts on the environment and human health because it is one of the causes of air and noise pollution. Therefore, improving traffic conditions is an important issue in Vietnam and around the world. It is not enough to plan urban areas properly, expand the road system or reduce the traffic volume, a intelligent traffic regulation system will be a useful solution and optimization.

Recent years have witnessed a rapid development of information technology and especially artificial intelligent (AI). AI has been shown to be tremendously beneficial to many aspects of life, including traffic. More specifically, deep learning, an outstanding technique in artificial intelligent has been developed and solved problems in traffic such as vehicle detection, vehicle tracking, traffic flow prediction and many orther problems. Among them, traffic flow prediction is an important prerequisite for traffic regulation. Traffic flow prediction is a combination of time series prediction and Big Data analysis. There are many approaches to time series prediction problem based on deep learning, machine learning algorithms, etc. For example, using spatial temporal graph neural network [1], which can comprehensively capture spatial and temporal patterns and effectively aggregate information from adjacent roads. Another approach using model combining the attention Conv-LSTM and Bi-LSTM [2], this model extracts daily and weekly periodic features so as to capture variance tendency of the traffic flow from both previous and posterior directions based on the short-term as well as long-term spatial and temporal features. On the other hand, currently, almost everything is now interconnected due to the growth of the Internet of Things (IoT) which has led to an explosion of data. And traffic data is also a kind of big data, collected from numerous sensors on a lot of different roads. Not only that, they are also streaming data sources. Towards solving the traffic flow prediction problem over a wide range (predicted simultaneously on multiple paths) requires the tools and powerful technique can handle aforementioned big data. Apache Hadoop is an example, it is one of the first open source frameworks for storing and processing Big Data, enables distributed processing of large datasets on clusters of computers. Hadoop works on an algorithm called MapReduce. This algorithm will break the jobs into small parts and divide them among the machines in the distributed system. It then aggregates to the final result. Apache Spark is another framework for big data processing, it is a data processing engine for batch and streaming modes featuring SQL queries, Graph Processing and Machine Learning. Spark can process real-time data which from real-time event streams at a rate of millions of events per second. Compared to Hadoop, Spark's processing speed is many times faster. This makes Spark more suitable for many problems, especially in traffic.

In this paper, traffic flow is predicted in real time on Big Data platform, applied simultaneously to many roads, the prediction results are visualized on the heat map. Firstly, a Prophet model is built to predict traffic flow at next time steps in the future. Training data is collected from sensors on the roads, including information about the average speed, time, location of the sensors and so on. And Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well [8]. After that, the results predicted from the model are displayed on a real-time heat map. Experimental entire process are used PySpark, an interface for Apache Spark in

Python. Details of the methods and experimental results are presented in the following sections of this paper.

2 Related works

Hitherto, there have been many research works on traffic flow prediction problem. In 2014, Yisheng Lv et al proposed a deep-learning-based traffic flow prediction method [4]. They used a stacked autoencoder (SAE) model is trained in a layerwise greedy fashion to extract generic traffic flow features. The spatial and temporal correlations are also considered in the model. Following that, a method using deep learning approach proposed by Rui Fu et al (2016) [5], Long Sort Term Memory Neural Network (LSTM NN) and Gated Recurrent Neural Network (GRU NN) models are applied to predict traffic flow. They used Adam optimizer with adaptive learning rates, the results show that GRU NN model perform a little than LSTM NN model. Nicholas G. Polson et al proposed a deep learning architecture that combines a linear model that is fitted using l_1 regularization and a sequence of tanh layers [6]. Another method used machine learning-based models for real-time traffic prediction was proposed by Sun et al in 2020. They used many models such as Artificial Neural Network (ANN), Support Vector Regression (SVR), Long Sort Term Memory Neural Network and compare their results.

3 Proposed real-time traffic flow prediction architecture

3.1 Proposed system architecture

The proposed system architecture consists of four parts: Data source, Batch Processing, Stream Processing and Stream Visualizing as described in Fig.1.

The entire data of the system are collected from a reliable data source, which is published on IEEE DataPort. Data is collected in two ways. The first way is to download a large amount of data enough to training the prediction model in Batch Processing part. The other way, data is received by Spark Structure Streaming in Streaming Processing part as input to the real-time traffic flow prediction model trained in Batch Processing part. In Batch Processing, the data collected from the Data Source is saved in csv format and goes through preprocessing steps before put in to training model. The preprocessing steps are also performed similarly for the data in Stream Processing part. The model is trained in Batch Processing part will get these data as input to make predictions about traffic flow in the future using Spark Streaming. Final results are shown on the heatmap.

3.2 Receive and process traffic streaming data

In the data science process, data collection and preprocessing is one of the important stages, determining the success of the entire process. In this paper,

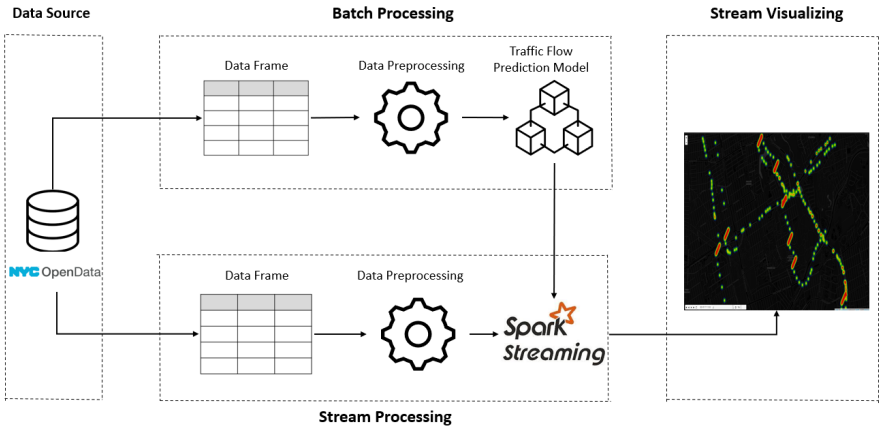

Fig. 1. System architecture

Real-time Traffic Speed Data is selected as the data set for the problem. The dataset is taken from NYC Open Data that is is free public data published by New York City agencies and other partners. The data consists of 13 attributes containing information about time, average speed of road segments, road segment name, location of sensors, etc. The dataset has more than 50 million data samples and is constantly being updated from real-world sensors. The process of collecting and processing this data set is summarized as schematically in Fig 2 .

Fig. 2. Data collection and processing process

First, the data is downloaded directly from NYC Open Data, which is historical data and is used to train the prediction model. The downloaded data is in csv format and undergoes some preprocessing steps such as drop unnecessary columns, error data lines, etc. It is then divided into two training and test data sets to train the model. After the model is trained, the data will continue to

be downloaded as input for the predictive model. This data is current and collected in real time using the NYC Open Data API provided. However, the data collected from the API is in JSON format, before preprocessing as above, they must be convert to csv format.

3.3 Traffic flow prediction model

As the core of the problem, the model determines most of the system's performance. In this paper, Prophet is the model chosen to train for traffic forecasting. Prophet is an open source software that is available in Python and R for forecasting time series data. It includes three main features: trend, seasonality, holidays [8], which can be represented as in equation (1).

$$y(t) = g(t) + h(t) + s(t) + \epsilon t \tag{1}$$

- g(t): piecewise linear or logistic growth curve for modelling non-periodic changes in time series.
- s(t): periodic changes (e.g. weekly/yearly seasonality).
- h(t): effects of holidays (user provided) with irregular schedules.
- ϵ (t): error term accounts for any unusual changes not accommodated by the model.

Prophet is robust to missing data and shifts in the trend, and typically handles outliers well. According to research by Taylor and Letham [8], Prophet is used in many applications on Facebook to provide reliable forecasts and works better than any other approach in the majority of cases. In this research, 132 Prophet models were trained corresponding to 132 roads in New York City. These models are combined into one main model for inclusion in Stream Processing part.

3.4 Online real-time traffic flow prediction pipeline

After the model is trained, it will be combined with real-time data get from the data source's API to perform traffic flow predictions according to the pipeline in Fig. 3

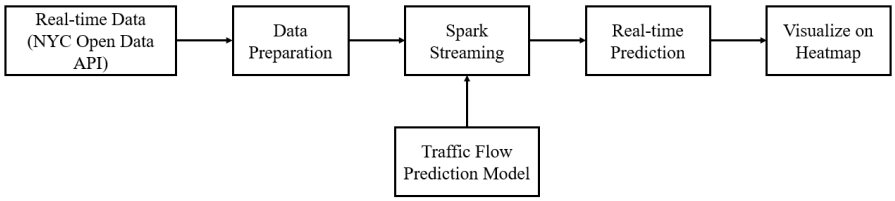

Fig. 3. Online real-time traffic flow prediction pipeline

Spark Streaming is used to crawl real-time data from the NYC Open Data API, then the data will be preprocessed to transform into the input format of the prediction model. The model will make predictions about the traffic flow of the roads for the next time period based on these current data. Folium framework is used to derive predictions from the model, and combine with the coordinate information of each road to visualize a heat map.

4 Experiment

4.1 Experiment procedure

The experimental steps of this research are shown according to the procedure described in Fig. 4. According to the procedure, the data collection and preprocessing are divided into two separate parts. The first part is to collect historical data for training and testing the predictive model. The rest will collect real-time data for the online real-time traffic flow prediction phase. The model used for training is Prophet. The trained Prophet model will be evaluated by two measures, root mean square error (RMSE) and mean absolute error (MAE), if the performance of the model is not satisfactory, the model will be updated. The process of training and evaluating and updating the model is repeated until the model achieves the desired results. The final model selected will be used to predict real-time online traffic with clean real-time data during the Aggregation and Prepare phase as input. Final results is the traffic flow in consecutive time intervals of the roads in the dataset. These results are combined with available coordinates of the roads to visualize a real-time heat map. The entire experimental process is processed on Colab Pro.

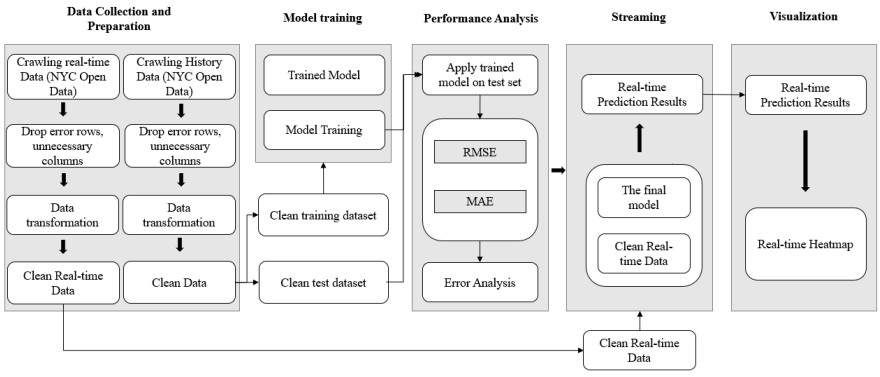

Fig. 4. Experiment procedure

4.2 Experiment result and discussion

The experimental results show that the trained model has relatively good performance. From the heat map in Figure 5, the high-traffic flow of the road are shown as orange and red areas and low-traffic areas in green.

Fig. 5. Traffic flow heatmap

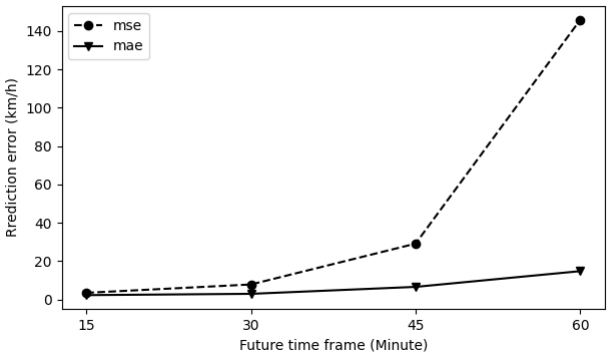

Fig. 6. Model performance over time

Model performance is also shown in Fig. 6 and Fig. 7. Figure 6 is the result of model evaluation based on RMSE and MAE and Figure 7 is the results of the model's speed prediction compared to the actual speed in a day. Accordingly,

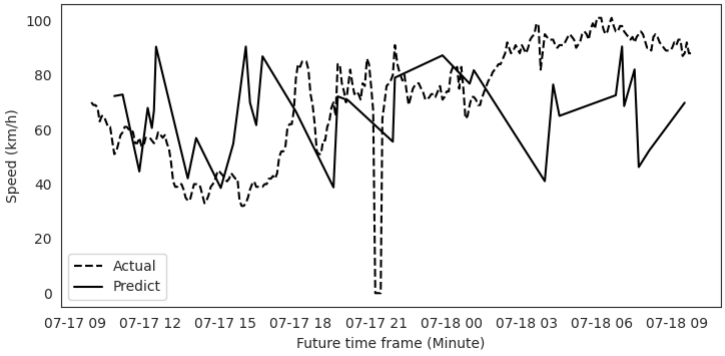

Fig. 7. The model's speed prediction results compared to the actual speed for a day

the predicted speed results are not really close to reality, there are still many differences and the model performance is lower when predicting for further time points. Therefore, in order for the model to work with a stable performance, it is necessary to update the model after a certain period of time. This helps the model capture more recent road information and make more accurate predictions. In addition, the model performance is not really high, partly because the Prophet model is suitable for trend forecasting problems based on long-term information. Meanwhile, the traffic flow is highly dependent on short-term information in the past.

5 Conclusion

Traffic flow prediction problem is an important part of intelligent traffic system. Many published researches use traditional machine learning methods, others use more modern methods such as deep learning. However, the methods are still limited when they cannot be optimized for real-time application. In this paper, a traffic prediction method based on a combination of big data analytics and modern machine learning algorithms is capable of making real-time predictions and can be visualized on a real-time heat map. Experimental results show that the proposed system has relatively good performance.

References

1. Wang, X., Ma, Y., Wang, Y., Jin, W., Wang, X., Tang, J., Jia, C., and amp; Yu, J. (2020). Traffic flow prediction via spatial temporal graph neural network. Proceedings of The Web Conference 2020.
2. Zheng, H., Lin, F., Feng, X., amd Chen, Y. (2020). A Hybrid Deep Learning Model With Attention-Based Conv-LSTM Networks for Short-Term Traffic Flow Prediction. IEEE Transactions on Intelligent Transportation Systems, 1–11.

3. Taylor, S. J., and amp; Letham, B. (2017). Forecasting at scale.
4. Lv, Y., Duan, Y., Kang, W., Li, Z., and Wang, F.-Y. (2014). Traffic Flow Prediction With Big Data: A Deep Learning Approach. IEEE Transactions on Intelligent Transportation Systems, 1–9.
5. Fu, R., Zhang, Z., Li, L. (2016). Using LSTM and GRU neural network methods for traffic flow prediction. 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC).
6. Polson, N. G., Sokolov, V. O. (2017). Deep learning for short-term traffic flow prediction. Transportation Research Part C: Emerging Technologies, 79, 1–17.
7. Sun, P., Aljeri, N., Boukerche, A. (2020). Machine Learning-Based Models for Real-time Traffic Flow Prediction in Vehicular Networks. IEEE Network, 1–8.
8. Taylor, S. J., Letham, B. (2017). Forecasting at Scale. The American Statistician, 72(1), 37–45.