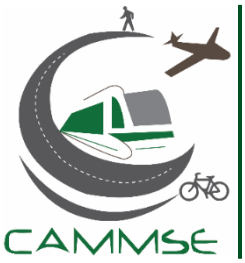


Center for Advanced Multimodal Mobility Solutions and Education

Project ID: 2021 Project 08

SHORT-TERM INTERSECTION TRAFFIC FLOW FORECASTING

Final Report

by

Yi Qi, Ph.D., P.E. (ORCID ID: https://orcid.org/0000-0002-6314-2626)
Professor and Chair, Department of Transportation Studies
Texas Southern University
TECH 215B, 3100 Cleburne Ave, Houston, TX 77004
Phone: 1-713-313-6809; Email: Yi.Qi@tsu.edu

Qun Zhao (ORCID ID: https://orcid.org/0000-0003-3760-9234)
Senior Research Associate, Department of Transportation Studies
Texas Southern University
TECH 208, 3100 Cleburne Ave, Houston, TX 77004
Phone: 1-713-313-1854; Email: qun.zhao@tsu.edu

Wenrui Qu (ORCID ID: https://orcid.org/0000-0003-4139-3544)
Assistant Professor, School of Mathematics and Statistics
Qilu University of Technology (Shandong Academy of Sciences)
University Road 3501, Jinan, China 250353
Email: qwr@qlu.edu.cn

Mehdi Azimi, Ph.D., P.E. (ORCID ID: https://orcid.org/0000-0001-5678-0323)
Assistant Professor, Department of Transportation Studies,
Texas Southern University
Phone: 1-713-313-1293; Email: Mehdi.Azimi@tsu.edu

for

Center for Advanced Multimodal Mobility Solutions and Education
(CAMMSE @ UNC Charlotte)
The University of North Carolina at Charlotte
9201 University City Blvd
Charlotte, NC 28223

September 2022

ACKNOWLEDGEMENTS

This project was funded by the Center for Advanced Multimodal Mobility Solutions and Education (CAMMSE @ UNC Charlotte), one of the Tier I University Transportation Centers that were selected in this nationwide competition, by the Office of the Assistant Secretary for Research and Technology (OST-R), U.S. Department of Transportation (US DOT), under the FAST Act. The authors are also very grateful for all of the time and effort spent by DOT and industry professionals to provide project information that was critical for the successful completion of this study.

DISCLAIMER

Table of Contents

EXCUTIVE SUMMARY ..... **xiii**

Chapter 1. Introduction **1**

1.1 Problem Statement 1

1.2 Objectives 3

1.3 Report Overview 3

Chapter 2. Literature Review **4**

2.1 Introduction 4

2.2 Artificial Neural Network (ANN) 4

2.3 KNN and Improved KNN Algorithm 5

2.3.1 Extended State Vector 5

2.3.2 Improved Distance Measurements 6

2.3.3 Improved Methods for Determining the K Value 6

2.3.4 Enhanced Prediction Algorithm 6

2.3.5 Improvements on Other Steps when Developing KNN Models 7

2.4 Entropy Weight Methods (EWM) 8

2.5 Summary 9

Chapter 3. Data Description and Processing **11**

3.1 Introduction 11

3.2 Calculating Vehicle Arrival Rate 11

3.3 Determining Time Interval 11

3.4 Abnormal Data 12

Chapter 4. Methodology **13**

4.1 Introduction 13

4.2 Clustering 14

4.3 K-Nearest Neighbor's Algorithm (KNN) 15

4.4 Backpropagation (BP) Neural Network 16

4.5 Elman Neural Network 16

4.6 Improved K-Nearest Neighbor's Algorithm 18

4.6.1 Weighted Distance Measurement 18

4.6.2 Optimized K Value 19

4.6.3 Improved Prediction Algorithm 20

4.7 Integrated Prediction Models Based on Entropy Weight Methods 20

4.7.1 Entropy Weight Method A (EWM-A) 20

4.7.2 Entropy Weight Method B (EWM-B) 22

4.7.3 Entropy Weight Method C (EWM-C) 23

Chapter 5. Model Evaluation **28**
5.1 Comparison of the Four Single Models 28
5.2 Comparison of Improved KNN, Elman and Three EWM-based Integrated Models 31
Chapter 6. Conclusions and Limitations **34**
6.1 Conclusions 34
6.2 Limitations 35
References **36**

List of Figures

Figure 2.1: Forecasting Mechanism Based on the Temporal and Spatial Dimensions 5
Figure 3.1: Vehicle Arrival Rate..... 11
Figure 4.1: Inputs and Outputs of the Developed Models..... 13
Figure 4.2: General Framework of the Clustering method -based Algorithm..... 15
Figure 4.3: General Framework of the KNN Algorithm. 16
Figure 4.4: Structure of BP Neural Network. 16
Figure 4.5: Structure Diagram of Elman Network..... 18
Figure 5.1: Traffic Flow Prediction for a Weekday..... 28
Figure 5.2: Traffic Flow Forecast for a Weekend..... 29
Figure 5.3: Comparison of RMSEs of Different Models..... 30
Figure 5.4: Comparison of CCPOs of Different Models..... 30
Figure 5.5: Traffic Flow Predictions for a Weekday..... 31
Figure 5.6: Traffic Flow Predictions for a Weekend..... 32

List of Tables

Table 4.2: KNN Model and Elman Model Weight Distribution Table 26
Table 5.1: Comparison of 6-day Average RMSEs and CCPOs of Different Models..... 31
Table 5.2: Comparison of MSE of Different Models 33

EXECUTIVE SUMMARY

The intersection is a bottleneck in an urban roadway network. As traffic demand increases, there is a growing congestion problem at urban intersections. Short-term traffic flow forecasting is crucial for advanced trip planning and traffic management. However, there are only a handful of existing models for forecasting intersection traffic flow. In addition, previous short-term traffic flow forecasting models usually were for predicting roadway conditions in a very short period, such as one minute or five minutes, which is often too late given that a driver may well be approaching the bottleneck already. Being able to accurately predict traffic congestion in about half-hour advance is very critical for advanced trip planning and traffic management.

To fill this gap, this research evaluated different methods used for short-term traffic flow forecasting. 24-h cycle by cycle traffic data collected at a signalized intersection in Jinan, China is used to develop models. First, single models are developed, including clustering, k-nearest neighbors (KNN), backpropagation neural network (BP), and Elman models. Next, an improved KNN model was developed to improve the prediction accuracy of the original KNN model. In addition, entropy-weight-method based integrated models are also developed. Three different types of entropy weight methods (EWMs), i.e., EWM-A, EWM-B, and EWM-C, have been used by previous studies for integrating prediction models. These three methods use very different ideas for determining the weights of individual models for integration. To compare the performances of these three EWMs, this study also applied them to develop integrated short-term traffic flow prediction models for the same selected signalized intersections by combining the improved KNN and Elman models. These two models were selected because they have been widely used for traffic flow prediction and have been approved to be able to achieve good performance. After that, three integrated models were developed by using the three different types of EWMs. The performances of the three integrated models were compared with improved KNN and Elman models.

The developed models are evaluated by applying them to the same intersection for forecasting the short-term traffic conditions on a different set of days. The prediction performance of these models was compared. We found that for the four single models, KNN outperforms other models. For EWMs based integrated models, the traffic flow predicted with the EWM-C model is the most accurate prediction for most of the days. Based on the model evaluation results, the advantages of using the EWM-C method were deliberated and the problems with the EWM-A and EWM-B methods were also discussed.

The objectives of this project are to (1) introduce different methods to predict short-term traffic flow at signalized intersections; (2) develop models with filed collected data; and (3) evaluate the performance of the different models.

Chapter 1. Introduction

1.1 Problem Statement

Intersections are the bottlenecks in the urban roadway network. As traffic demand increases, there is a growing congestion problem at urban intersections. Short-term traffic flow forecasting is crucial for advanced trip planning and traffic management, especially for the highly congested and densely signalized urban roadways. The main challenge in studying traffic flow problems is that the traffic flow data are unevenly distributed, highly dimensional, and dynamic changing (Gao et al., 2021). The presence of signalization gives traffic a Spatio-temporal behavior that is more random and difficult to study than in freeways (Head, 1995). In addition, traffic flow in urban signalized arterials has a certain temporal and spatial behavior that exhibits randomness which escapes the traditional perception of periodicity (monthly, weekly, daily, or even hourly periodicities) in traffic operations (Stathopoulos and Karlaftis, 2001).

Traffic flow forecasting can be classified into long-term and short-term forecasting. Long-term forecasting usually targets one or more whole days in the future. Short-term models forecast the traffic in the near future (such as a few minutes) based on current and past traffic conditions. It can provide the basis for optimal trip planning, route guidance, adaptive traffic signal control, and other advanced traffic management schemes. This study focuses on short-term traffic forecasting for signalized intersections. By reviewing the existing methods, it is noticed that some methods can predict short-term traffic flow effectively but cannot adapt well to large-scale data processing. Some methods take all the historical data as the input for forecasting instead of selecting a subset of data that is more likely to have a similar pattern as the target day for forecasting, which will have negative impacts on prediction accuracy and efficiency. In addition, most of the existing methods have only been applied for predicting traffic flow in a very short period, such as one minute or five minutes, which is often too late given that a driver may well be approaching the bottleneck already. For advanced trip planning and traffic management purposes, a model that can forecast traffic conditions about a half-hour in advance is needed. Therefore, the performance of the existing methods in predicting traffic flow in a longer time window needs to be investigated.

For this purpose, in this study, four representative existing individual methods, i.e., clustering, k-nearest neighbor algorithm (KNN), backpropagation (BP) neural network, and Elman neural network methods, are selected and applied for predicting the traffic flow condition at a signalized intersection in half-hour advance. We also developed an improved KNN model. In addition, integrated models are also developed by combining two individual methods: (1) improved KNN and (2) Elman Neural Network with entropy weight methods (EWM). These two models were selected because they have been widely used for traffic flow prediction and have been approved to be able to achieve good performance.

The entropy weights method (EWM) is a commonly used information-weighting method in decision-making. Entropy analysis has been applied to traffic and transportation planning since the 1980s (Erlander, 1980; Wilson, 1981). Previous studies applied entropy-based methods to identify different levels of the orderliness of traffic flow in a roadway network for the

purposes of incident detection, roadway safety analysis, and driving behavior analysis (Crisler and Storf, 2012; Gao et al., 2021; Kim et al., 2012; Koşun and Özdemir, 2017; Petrov, 2022; Xie et al., 2021).

It has been widely used in comprehensive evaluation studies that use different evaluation indexes (Dang and Dang, 2019; Sheng et al., 2021; Zhao et al., 2012). In these studies, the weights of different indexes are determined according to the degree of dispersion. The smaller the entropy value, the greater the degree of dispersion of the index, and the greater the in-fluence of the index on the comprehensive evaluation. Therefore, it should be signed with greater weight(Dang and Dang, 2019). Recently, EWM has been used in integrating different prediction models to get better predictions (Bai et al., 2020; Huang et al., 2017; Shan and Zhang, 2021). In these studies, the weights of different models, which quantitatively measure the importance of each model, were determined based on the degree of dispersion of the prediction errors. However, there are two different opinions on determining the weight of an individual model. Some studies believe that a smaller information entropy value means that the data are provided by many useful attributes, so a larger weight should be assigned and vice versa(Bai et al., 2020; Zhang et al., 2011). On the contrary, some studies suggest that a smaller entropy value of the prediction error indicates that the variation degree and uncertainty of model prediction is greater, and thereby, a smaller weight should be assigned to this model and vice versa (Chen and Li, 2009; Huang et al., 2017; Sun et al., 2021; J. Wang et al., 2016). One recent study (Shan and Zhang, 2021) indicates that there is a nonlinear relationship between the entropy value and model accuracy level. Both low-accuracy and high-accuracy prediction models can result in small entropy values of the model prediction errors. Thus, the weight cannot be assigned based on the entropy value alone. To address this problem, they proposed a new entropy weight method for model integration. The prediction ac-curacy level of each individual model was incorporated into the calculated weights to reduce the impact of the model with less accuracy, which results in the improved pre-diction accuracy of the integrated model.

These three different EWMs have all been used by researchers for integrating prediction models (Bai et al., 2020; Huang et al., 2017; Shan and Zhang, 2021; Zhang et al., 2011). They use very different ideas for determining the weights of individual models for integration. However, there is a lack of research that compares the performance of these different methods and identifies the best EWM for integrating prediction models. To address this problem, in this research, these three different entropy weight-based methods were applied to develop integrated models to predict the short-term traffic at signalized intersections. Their performances were compared and analyzed based on the results of this study.

In this study, three integrated models were developed by using the three different EWM-based methods. The developed integrated models were evaluated by comparing the predicted traffic flow rates with the traffic data collected at a real-world signalized intersection. In addition, the performance of each model was analyzed and compared with single models, and conclusions and recommendations of this study were provided.

1.2 Objectives

The objectives of this report are to (1) introduce different methods to predict short-term traffic flow at signalized intersections; (2) develop models with filed collected data; and (3) evaluate the performance of the different models by comparing the predicted results with observed results.

1.3 Report Overview

The remainder of this report is organized as follows: Chapter 2 presents a comprehensive review of the state-of-the-art and state-of-the-practice literature on the methods of short-term traffic flow forecasting problems. Chapter 3 introduces the data collection and processing at a real-world intersection that is used for model development and evaluation. Chapter 4 provides a detailed introduction of each method used in this study to develop short-term forecasting model, including five individual methods and three EWM-based integrated methods. Chapter 5 presents the model evaluation and comparison results and discussed the performance of each method. Chapter 6 concludes this report with a summary and a discussion of the limitations of this research.

Chapter 2. Literature Review

2.1 Introduction

To predict the short-term traffic flow, many existing models have been developed by previous researchers with different approaches. The existing methods can be classified into three categories: statistical methods, machine learning methods, and integrated methods combining two or more models.

Statistical models include the historical trend models (Jiang et al., 2013), nonparametric regression models (Davis and Nihan, 1991; Smith et al., 2002; Smith and Demetsky, 1994), and time-series models (Ahmed and Cook, n.d.; Hamed et al., 1995; Moorthy and Ratcliffe, 1988; Omkar and Kumar, 2017; Smith et al., 2002; Williams et al., 1998).

Machine learning models include clustering-based methods (Han et al., 2019a), neural network models (Dougherty et al., 1993; Niu et al., 2015; Pamuła, 2013; Park et al., 1998; Smith and Demetsky, 1997; Zhang et al., 1997; Zhao et al., 2008), Kalman filter models (Okutani and Stephanedes, 1984), support vector machine models (Fu et al., 2013; D. Wang et al., 2016), decision tree-based methods (Liu and Wu, 2017; D. Wang et al., 2016; Xia and Chen, 2017; Yang et al., 2017) and so on. With more and more traffic data available recently, many other machine learning models have been developed. For example, researchers built a deep architecture model stacked with an autoencoder model to predict traffic flow (Lv et al., 2015).

Integrated methods (or hybrid methods) have become more popular recently with promising results. Examples include combining some clustering methods with either time-series analysis or neural network models (Kuchipudi and Chien, 2003; Liu et al., 2018; Park et al., 1998; Song et al., 2010; Yin et al., 2002), combining backpropagation (BP) neural network with the radial basis function (RBF) neural network (Zheng et al., 2006). To combine two and more individual models, the most common way is to use the regression method, however, recently, Entropy Weight Method (EWM) is getting popular for integrating models.

In this part, research on the existing short-term forecasting methods and Entropy Weight Method (EWM) will be introduced.

2.2 Artificial Neural Network (ANN)

Artificial neural network (ANN) is one widely used forecasting method. It has the non-linear mapping and non-parametric characteristics and has great application potential in traffic flow prediction (Smith and Demetsky, 1994). Many researchers have applied the ANN or Back Propagation (BP) neural network to predict traffic flow rate or congestion levels (Çetiner et al., 2010; Ho and Ioannou, 1996; Jiber et al., 2018; Karim et al., n.d.; Kashyap et al., 2022; Khotanzad and Sadek, 2003; Kou et al., 2018; Kumar et al., 2013; Sharma et al., 2018; Shenfield et al., 2018; Zheng et al., 2006). Recently, a dynamic feedback neural network called Elman was used in traffic flow prediction and showed improved results (Ishak et al., 2003; Li and Lu, 2009; Ma and Wang, 2015; Qu et al., 2020; Zhao et al., 2008). Elman neural network adds a context layer to the network, which makes the output of the network at the current moment not only

depend on the current inputs but also related to the inputs at the previous moment using a memory function. This feature makes the Elman model outperform the traditional BP model (Qu et al., 2020).

2.3 KNN and Improved KNN Algorithm

The K-Nearest Neighbor Algorithm (KNN), a classic non-parametric regression method, has been widely used in short-term traffic forecasting. It has been approved to be able to achieve good performance (Cai et al., 2016; Kou et al., 2018; Smith et al., 2002; Wu et al., 2014; Yu et al., 2016). In these studies, several KNN-based models were developed by improving the basic KNN algorithm. In summary, the KNN algorithm can be improved in four aspects.

2.3.1 Extended State Vector

Different state vectors can be used for the KNN regression. More precisely, there is an infinite number of possible state vectors. It is believed the most reasonable features to be used are present and time-lagged values of the time series:

$$x(t)=[V(t), V(t-1), V(t-2) \ldots V(t-d)]$$

Where d is the selected lag.

However, another research (Smith et al., 2002) has shown that using past average values yields more accurate forecasts.

In a traditional KNN model, the state vector reflects the condition of traffic flow on the target link. However, in urban networks, none of the links stand in isolation. The traffic condition of the target link is bound to be affected by that of the upstream and downstream links, especially when congestion occurs on either of the adjacent links. Therefore, theoretically speaking, if the spatial information is considered in the state vector, more similar neighbors would be found and then the forecasting accuracy of the model would be largely improved. Wu et al. (Wu et al., 2014) redefined the state vector by considering both the spatial and temporal information. The forecasting mechanism is illustrated in the following figure.

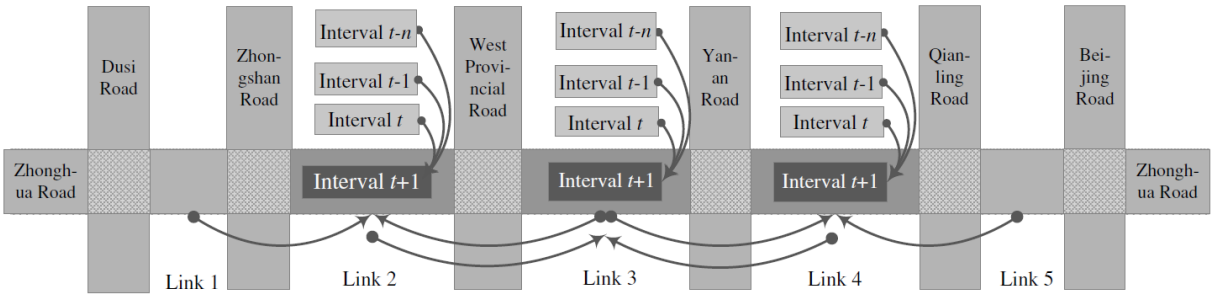


Figure 2.1: Forecasting Mechanism Based on the Temporal and Spatial Dimensions

2.3.2 Improved Distance Measurements

The common method of measuring "proximity" in non-parametric regression is to use Euclidean distance (Chomboon et al., 2015; Lopes and Ribeiro, 2015) or weighted Euclidean distance (Chomboon et al., 2015) to calculate the distance between state vectors. There are other distance measuring methods that have been utilized by researchers, such as the Manhattan distance (Chomboon et al., 2015; Gao and Li, 2020; Lopes and Ribeiro, 2015; Mulak and Talhar, 2013), Has-sanat distance (Abu Alfeilat et al., 2019; Alkasassbeh et al., 2015), and Chi-square (Hu et al., 2016). Improvements include using the weighted Euclidean distance by considering different factors. For example, Yu et al. (Yu et al., 2016) suggested that weights should be assigned based on the close degree between time components in the state vector and the forecasting time. Habtemichael and Cetin also recommended giving more weight to the recent measurements and less to the older ones (Habtemichael and Cetin, 2016).

2.3.3 Improved Methods for Determining the K Value

Based on the calculated distance, the K nearest neighbors can be identified. The KNN model is sensitive to the selected K value, and the K value affects the model accuracy (Zhang et al., 2018). Previous studies have used different methods to determine the K value based on average absolute percentage error, relative error, and root mean square error (Cai et al., 2016; Ghosh, 2006; Kou et al., 2018; Lall and Sharma, 1996; Liu et al., 2010; Smith et al., 2002; Wu et al., 2014; Yu et al., 2016).

2.3.4 Enhanced Prediction Algorithm

For the KNN method, the model prediction is mainly based on the simple average or weighted average of the K nearest neighbors. There are different methods to determine the weights. For example, ref. (Kou et al., 2018; Wu et al., 2014; Yu et al., 2016) used the inverse distance as the weight, and ref. (Cai et al., 2016) used the Gaussian function to determine the weights of the selected neighbors.

Similar to the state vector, there is also an infinite number of possible forecast estimations.

In one research (Wu et al., 2014), the authors introduced two prediction algorithms, which are:

- The simple average: produces the output as the average of the nearest neighbors

$$S(t+1) = \frac{1}{k}\sum_{m=1}^{k} S_{hm}(t+1)$$

- The weighted average: uses the inverse distance as the weight

$$S(t+1) = \sum_{m=1}^{k} \frac{d_m^{-1}}{\sum_{m=1}^{k} d_m^{-1}} S_{hm}(t+1)$$

2.3.5 Improvements on Other Steps when Developing KNN Models

Other improvements include weight assignment, search step length, window size, multi-time-step models, and others.

(Habtemichael and Cetin, 2016) proposed an enhanced K-nearest neighbor (K-NN) algorithm to predict short-term traffic based on identifying similar traffic patterns. After selecting candidate profiles for prediction, the rank-exponent method was used to weight assignment because of its advantageous as it provides some degree of flexibility in the way weights are assigned by adjusting the weight dispersion measure.

In their research (Wu et al., 2014), the authors adapted the tradition KNN model to improve forecasting accuracy. One improvement was applying the exponent weighting method to determine the weighted distance.

It is very important that optimal lag duration and number of candidates are used to minimize the forecast error. Lag duration affects the performance of the K-NN-based traffic forecast as it is the main variable that identifies similar traffic patterns. In their study, Habtemichael and Cetin (Habtemichael and Cetin, 2016) considered a series of lag durations, ranging from just one hour up to 23 h. By comparing forecast accuracy, it was observed that with an increase in lag duration, forecast errors increase. This shows that the optimal lag duration for identifying similar traffic patterns should be of relatively short duration; in our case, a one-hour lag duration is found to be most suitable.

Most existing KNN algorithms are single-step which has two main disadvantages: (i) generating overlapping nearest neighbors when the method is extended to multiple- step forecasting as demonstrated later; (ii) sensitive to noisy neighbors. To remedy these serious limitations, (Zheng and Su, 2014) also proposed a two-step approach to enhance the performance of the KNN method in forecasting short-term traffic volume. The improvements including 1) introducing a time constraint window when selecting k-nearest neighbors, 2) ranking the local minima of the distances between the state vectors to avoid overlappings among candidates, and 3) developing a novel algorithm with attractive analytical features to control extreme values' undesirable impact.

Yu et al. (Yu et al., 2016) developed a model to predict short-term traffic conditions based on k-NN models that consider both temporal and spatial information. In addition, a multi-time-step prediction model was proposed based on the single-time-step model. To validate the performance of the proposed k-NN model, a case study was conducted with the GPS data of taxis collected in Foshan city, China. As shown in the following figure, the results show that the k-NN model has a better forecasting accuracy than the artificial neural network model, real-time-data model, and history-data model, and slightly worse than SVM. Besides, the prediction accuracy was observed to decrease with the increment of prediction steps.

In one research (Zhang et al., 2013), authors developed a short-term freeway traffic flow prediction method based on road section traffic flow structure pattern. In their study, the authors first conducted the structural analysis of freeway road section traffic flow, and then based on the identified pattern, a new short-term traffic prediction method was proposed. In this paper, the

traffic flow structure pattern is the relationship between a certain road section of the freeway and the upstream stations. With the real freeway toll data and a few detective video camera data, the authors revealed the stability pattern of freeway section traffic flow and further verified the stability of this pattern with Coefficient of Variation as an index. Based on the traffic flow structure pattern, an improved Local Weighted Learning (LML) model was proposed. The new prediction method uses the upstream station entrance flow to correct the prediction of the section traffic flow. Finally, experimental studies based on real data show that the proposed algorithm is reasonable and feasible and that the accuracy is improved under abnormal traffic conditions. In addition, it was also found that the traffic flow of the current section had great correlations with the flow of its upstream stations.

One research (Pang et al., 2016) proposed a three-layer K-nearest neighbor non-parametric regression algorithm to forecast short-term traffic flow. Specifically, two screening layers based on shape similarity were introduced in K nearest neighbor non-parametric regression method, and the forecasting results were output using the weighted averaging on the reciprocal values of the shape similarity distances and the most-similar-point distance adjustment method.

2.4 Entropy Weight Methods (EWM)

The EWM is one of the weighting methods that measure the dispersion level of different information sources in decision-making. It has been widely used in comprehensive evaluation studies where the weights of different indexes are determined according to the entropy value of the different evaluation indexes. For example, Dang and Dang (Dang and Dang, 2019) used a multi-standard decision-making method to evaluate the environmental quality of the Organization for Economic Cooperation and Development countries. The weights and method standards were determined based on the entropy weight method. Zhao et al. (Zhao et al., 2012) developed an entropy-based model to predict automobile engine fault diagnosis. The weight of each factor in the evaluation was determined based on entropy. In all these comprehensive evaluation studies, a smaller entropy value of the indicator means a greater degree of dispersion, thereby it has a greater impact and should be assigned a greater weight.

Except for comprehensive evaluation studies, researchers also applied EWM methods for integrating different prediction models to improve prediction accuracy. In these studies, the weights of different models were determined based on the entropy of the model prediction errors. There are two different opinions on determining the weight of each individual model. Some studies believe that a model with a smaller entropy value of prediction error should be assigned a greater weight. For example, in a study (Bai et al., 2020), to predict the critical frequency of the ionosphere, authors used the entropy method to assign weights to the two single prediction results of Union Radio Scientifique Internationale and the International Radio Consultative Committee to develop an integrated prediction model. In this study, it was stated that a small information entropy value means the data are provided by many useful attributes, so a large weight should be assigned to this model. In another study (Zhang et al., 2011), to increase the prediction accuracy of software reliability failure data, authors established an interacted prediction model using the EWM. In this study, it was believed that if the value of in-formation entropy is smaller, the uncertainty is smaller, and greater weight should be given. It can be concluded that for the above papers, the basic idea for assigning weights to different models is

that the smaller the entropy value of the prediction error of an individual model, the greater the weight should be assigned. This type of EWM is referred to as type A EWM (EWM-A) in this study.

On the contrary, some other studies believe that a smaller entropy value of the prediction error indicates that the variation degree and uncertainty of model prediction is greater, thereby a smaller weight should be assigned to this model. For example, to accurately predict the Normalized Vegetation Difference Index (NDVI) in the Yel-low River basin, Huang et al. (Huang et al., 2017) developed a forecasting model by combining three in-dividual models, i.e., multilinear regression (MLR), artificial neural network (ANN), and support vector machine (SVM) models. The method used to determine the weight is EWM. The idea is that if the prediction error of a single prediction model varies greatly, the entropy value of the model is small, indicating that the model does not perform well and should be given a small weight. In another study, Sun et al. [8] used the same EWM to assign weights to the gray GM (1,1) model and the gray Verhulst model for predicting the bearing capacity of anchor bolts. Chen and Li [9] also used the same EWM to develop an integrated prediction model for unit crop yield prediction. To predict sintering energy consumption, Wang et al. used this EWM to assign weights to two sintering energy consumption models [10]. For all the above papers that used EWM for model integration, the basic idea for assigning weights to different models is that if an individual model has a smaller entropy value of prediction error, the prediction variance in a model is larger, and a smaller weight should be assigned to it. This type of EWM is referred to as type B EWM (EWM-B) in this study.

Besides these two commonly used EWMs, recently, Shan and Zhang (Shan and Zhang, 2021) proposed another EWM-based method for model integration. The authors indicate that there is a nonlinear relationship between the entropy value and model accuracy level. Both low-accuracy and high-accuracy prediction models can result in small entropy values of the model prediction errors. Thus, the weight cannot be assigned based on the entropy value alone. To address this problem, they proposed using a weighted entropy of the model prediction error, and the prediction accuracy level of the individual model was incorporated into this weighted entropy. In this way, the impact of the model with low accuracy can be reduced and the integrated model can be improved. This type of EWM is referred to as type C EWM (EWM-C) in this study.

In this paper, three integrated traffic flow prediction models were developed by using these three different types of EWMs, introduced above. Regarding the individual models, traffic flow forecasting has been intensively studied. Both parametric and non-parametric models were developed. Among all these models, the K-Nearest Neighbor (KNN) Algorithm and Artificial Neural Network (ANN) were approved to have good performance in predicting short-term traffic flow (Ishak et al., 2003; Li and Lu, 2009; Liu et al., 2018; Ma and Wang, 2015; Qu et al., 2020). Following is a brief introduction to the literature that used these two methods for developing traffic flow prediction models.

2.5 Summary

Some researchers compared the different approaches for short-term traffic flow forecasting. Smith and Demetsky (Smith and Demetsky, 1997) compared the historical average,

time-series, neural network, and nonparametric regression models, and found that the nonparametric regression model significantly outperformed the other models. Salotti et al. (Salotti et al., 2018) evaluated ten forecasting methods based on real-world city traffic data in two different contexts. They found the nonparametric approach (KNN) always outperforms other parametric methods in the particular city center context, while parametric approaches perform better in the freeway context.

By reviewing the existing methods, it can be seen that, for the signalized urban roadways, nonparametric regression models often outperform the traditional time series methodologies due to the rapid variations of traffic flow in urban areas (Abdulhai et al., 1999; Head, 1995). Also, the KNN regression model has been widely used for short-term traffic flow forecasting and has been reported to have good performance (Kindzerske and Ni, 2007; Yu et al., 2019; Zhang et al., 2013). In addition, it also has been reported that neural network models have many advantages over the classical statistical methods in short-term traffic flow forecasting (Han et al., 2019a; Vlahogianni et al., 2005). Furthermore, many previous studies have used clustering-based methods in traffic flow forecasting and have shown improved modeling performance (Han et al., 2019b; Song et al., 2018). Therefore, in this study, a representative nonparametric regression model, i.e., the KNN regression model, a clustering-based algorithm, and two neural network models were selected for comparing with the proposed integrated model.

Chapter 3. Data Description and Processing

3.1 Introduction

A signalized intersection in Jinan, China was selected as the study site. At this site, 156 days of traffic data were collected, from 1 October 2018 to 1 April 2019. Multi-dimensional data was collected, including:

- Cycle by cycle traffic flow rates
- Queue length
- Signal timing plan
- Day of the week
- Holiday or not

Since the raw data doesn't include the vehicle arrival rate, therefore, it should be calculated first with the data collected.

3.2 Calculating Vehicle Arrival Rate

Assume that the queue length is *Li* at the time *i*, the periodic flow at the stop line is *Zi*, and the number of vehicle arrivals is *Fi*.

If the periodic flow at the stop line is greater than or equal to the queue length at time *i+1*, then the number of vehicle arrivals at time *i* is the periodic flow at the stop line at time *i+1*; otherwise, the traffic at this time is congested, so the number of vehicle arrivals at time *i* is the queue length at time *i+1* minus the queue length at time *i* plus the periodic flow at the stop line at time *i+1*, as shown in Figure 3.1.

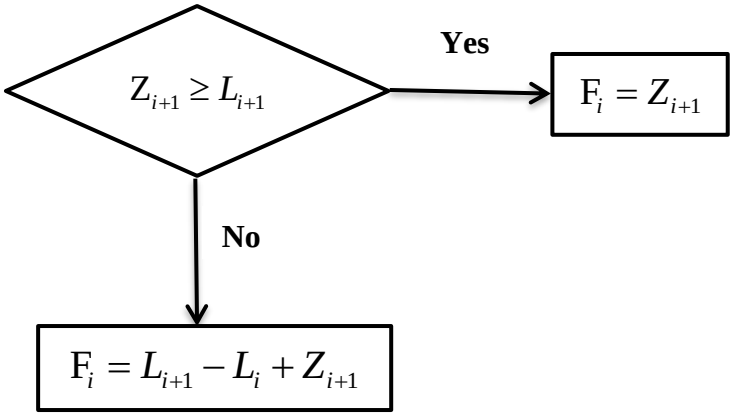


Figure 3.1: Vehicle Arrival Rate

3.3 Determining Time Interval

Since the cycle length of traffic lights is 1 minute and 30 seconds or 2 minutes depending on the time of the day, for uniformity, the cycle-by-cycle traffic flow data is aggregated at a 6-

min interval level. Thus, the traffic flow rate in this study is the number of arrival vehicles per 6 minutes.

3.4 Abnormal Data

The central data collection devices and management system are scheduled for regular maintenance every Tuesday, so there is no data available on Tuesday. After examination, it was found that there were many missing data from 21st to 29th November 2018, so these 9 days were removed. In addition, on some days, the data were missing for more than 3 consecutive time intervals (18 min). These data were also excluded from this study. For the data missing less than 18 min, for data imputation purposes, the average traffic flow rates before and after it were used as the estimated traffic flow rates for these time intervals.

The data were then divided into two sets, the training dataset, and the validation dataset. In this paper, 6 days' data from 27 March to 1 April were used for model validation, and the rest of the data were used as training data for model development. In addition, all data were also classified into three groups: weekdays, weekends, and holidays. Since there are only a few holidays during the study period, all the data collected during the holidays were excluded. Finally, only two groups of data, i.e., weekday and weekend, were included in both training and validation datasets.

Chapter 4. Methodology

4.1 Introduction

In this study, the short-term intersection traffic flow forecasting models are for predicting the amount of traffic that will arrive at an intersection 30 min later based on the arriving traffic flow rates during the past 3 h. Thus, mathematically, the model can be expressed by the following equation.

$$f\left(x_{t-29,}\ldots x_{t-1}, x_t\right) = x_{t+5} \quad (1)$$

where *t* is the current time interval and x_t is the arrival traffic flow rate during the current time interval. Since the traffic flow rate is at a 6-min interval, the vector $(x_{t+29} \ldots x_{t+1}, x_t)$ represent the arrival travel flow rates during the past 3 h and x_{t+5} represent the predicted rate of the traffic flow that will arrive at the intersection after half an hour. Figure 4.1 shows the inputs and outputs of the developed models.

In this paper, five single existing methods are considered for developing short-term intersection traffic flow forecasting models. They are clustering, k-nearest neighbor algorithm (KNN), improved KNN backpropagation (BP) neural network, and Elman neural network methods.

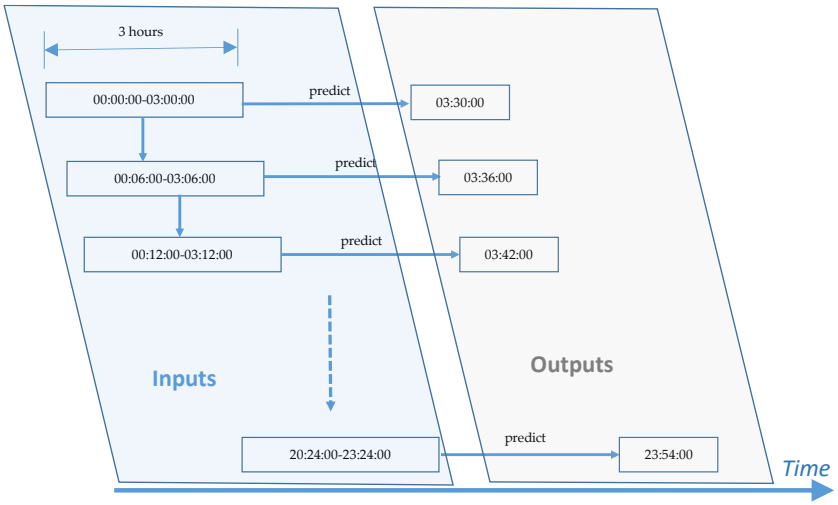


Figure 4.1: Inputs and Outputs of the Developed Models

In this paper, five individual methods are considered for developing short-term intersection traffic flow forecasting models. They are clustering, k-nearest neighbor algorithm (KNN), improved KNN model, backpropagation (BP) neural network, and Elman neural network. These models will be introduced one by one in the following section.

4.2 Clustering

Clustering is one of the most important unsupervised data mining methods. It groups a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). A classic definition for clustering is described as follows (Jain and Dubes, 1988; Xu and Tian, 2015):

- Instances, in the same cluster, must be similar as much as possible.
- Instances, in the different clusters, must be different as much as possible.
- Measurement for similarity and dissimilarity must be clear and have practical meaning.

In this study, a novel density-based clustering method developed by Rodriguez and Laio (Rodriguez and Laio, 2014) was used to divide the historical traffic flow data into different groups. A detailed introduction to this clustering method can be found in Song et al. (Song et al., 2018). In this study, the clustering method is used to find all the historical traffic flow records during the given period and on a given type of day (weekday or weekend) that has a similar traffic pattern as the target day. The period is 3 h before the current time t. To develop the clustering model, all the 3-h traffic flow vectors $(x_{t-29,} \dots x_{t-1}, x_t)$ were derived from the collected traffic data. Then, these vectors are divided into different groups according to the type of day (weekday or weekend) and the time of the day (t).

The similarity between the traffic flow data during the same 3-h period on different days was defined by the Euclidean distance as follows

$$d_{ij}^2 = \sum_{k=0}^{29} [x_{t-k}^i - x_{t-k}^j]^2 \; i \neq j \tag{2}$$

where *i* and *j* indicate different days, *t* is the current time interval and x_{t-k}^i is the arrival traffic flow rate during the time interval *t-k* in Day *i*.

The general framework of the clustering-based algorithm is presented in Figure 4.2. Given a target day *i* and a target time *t*, we will know the 3-h traffic flow condition before this time. Then, by applying the clustering method to the corresponding historical traffic flow vector group, the historical days that have a similar traffic pattern as the target day during the same 3-h period on the same type of day of the week can be identified. Then, by calculating the average traffic flow rates at time interval *t+5* (30 min after the target time *t*) in these identified days, the traffic flow rate for the target day 30 min after the target time t can be predicted.

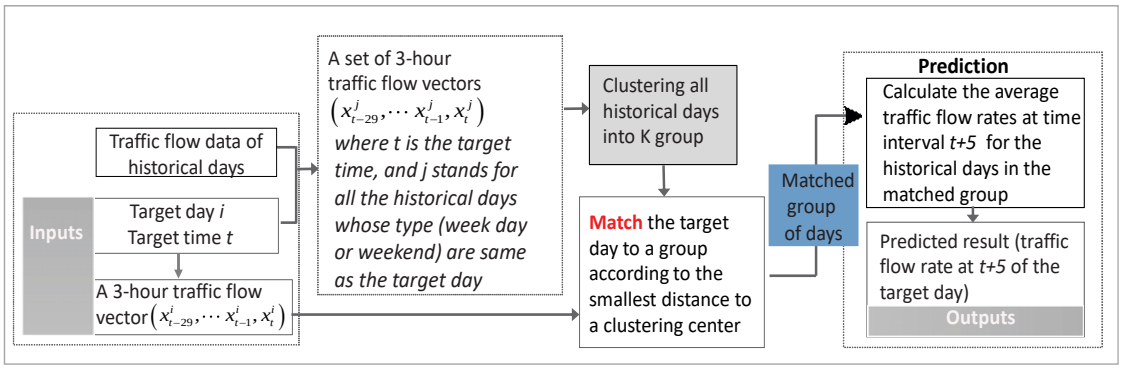


Figure 4.2: General Framework of the Clustering method -based Algorithm

4.3 K-Nearest Neighbor's Algorithm (KNN)

KNN is a non-parametric method that can be used for classification and regression. In this study, KNN was used for regression. The core idea for the KNN regression is to find the k nearest neighbors of a given set of inputs and use the average values of the k nearest neighbors as the outputs of the model. In this study, the input for the KNN model is the 3-h traffic flow vectors $(x_{t-29,} \dots x_{t-1}, x_t)$ of the target day at a given time t. To identify the k nearest neighbors for the inputs, the distance between this input and the samples in the training data needs to be defined at first. In this study, the Euclidean distance given before is also used for defining this distance. Similar to the Clustering method, the KNN method also aims to find out all the historical traffic flow records that have a similar pattern to the target day in a given time period and a given type of day (weekday or weekend), so as to predict the traffic flow rate in the next half hour for the target day. The only difference is that the Clustering method identifies all the traffic flow vectors in the same clustering group, while the KNN method only identifies the k most similar records that are closest to the inputted traffic flow vectors.

The general framework of the KNN model is presented in Figure 4.3. Basically, the KNN algorithm consists of the following key steps:

1) For a given target day *i* and a target time t, find the correspondent training dataset that contains historical traffic records (3-h traffic flow vectors) during the same type of days (weekday or weekend) at the target time *t*.
2) Calculate the distance between the inputted 3-h traffic flow vectors $(x_{t-29,} \dots x_{t-1}, x_t)$ and the traffic flow vectors in the training dataset according to the Euclidean distance given in Equation.
3) Select the *k* traffic flow vectors with the smallest distances.
4) According to the date and time of the selected traffic flow vectors, calculate the average traffic flow rate in these k days half an hour after the given time t, which will be the forecasted traffic flow rate for the target day *i* 30 min after the target time *t*.

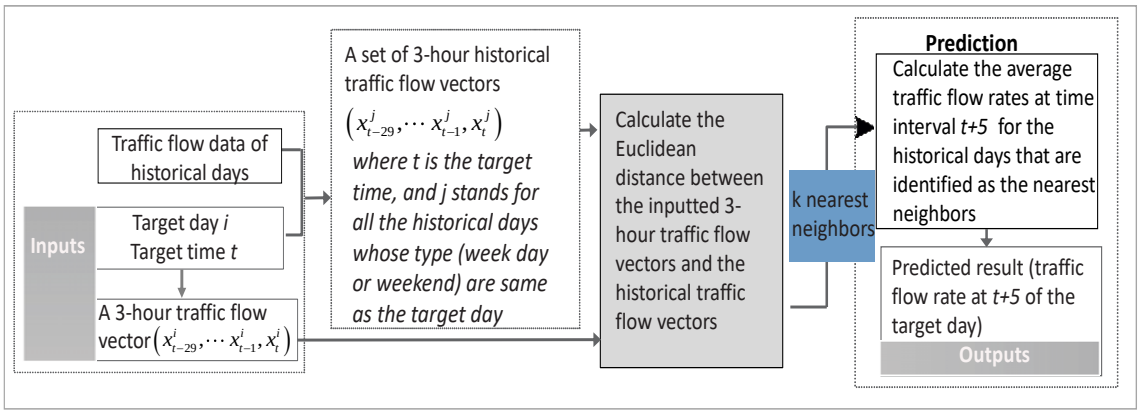


Figure 4.3: General Framework of the KNN Algorithm.

4.4 Backpropagation (BP) Neural Network

The BP neural network algorithm is one of the most widely applied neural network models and has been applied for traffic flow forecasting in some previous studies (Salotti et al., 2018). In this study, a BP neural network is selected with three layers: (1) an input layer, (2) a hidden layer, and (3) an output layer. The overall structure of the BP neural network is presented in Figure 4. In this structure, there are 30 neurons in the input layer, i.e., $(x_{t-29,} \dots x_{t-1}, x_t)$ which represent the 3-h traffic flow rate vectors and 1 neuron in the output layer, i.e., x_{t+5} which is the traffic flow rate in half an hour; w_{ij} is the weight for the connection from neuron i in the input layer to neuron *j* in the hidden layer, and v_{jk} is the weight for the connection from neuron j in the hidden layer to neuron *k* in the output layer. The training of the BP model was based on the backpropagation method. In this study, the transfer function for the neurons in the hidden is chosen as the sigmoid function and the transfer function for the neurons in the output layer is a linear function.

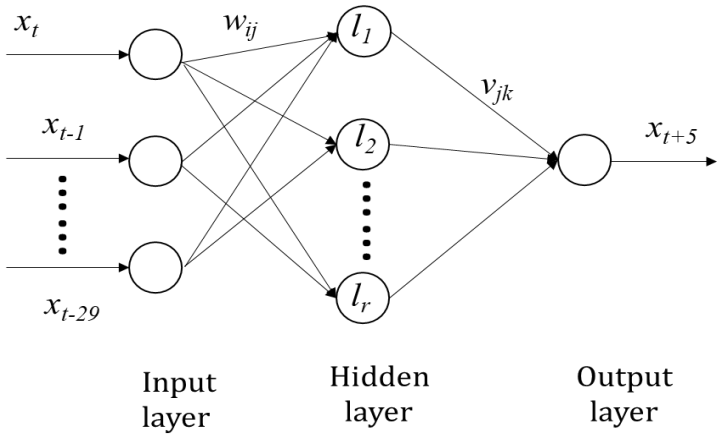


Figure 4.4: Structure of BP Neural Network.

4.5 Elman Neural Network

Elman neural network is a dynamic feedback neural network, which was proposed by Elman in 1990 for voice processing. Several previous studies have been conducted to develop

short-term traffic flow forecasting models based on Elman neural network (Dong et al., 2009; Niu et al., 2015; Song et al., 2010). The Elman neural network is generally considered to be a forward neural network with local memory units and local feedback connections. Its main structure is similar to the structure of the BP neural network, but it adds a context layer to the hidden layer based on the basic structure of the BP neural network. The context layer mainly receives feedback signals from the hidden layer as a delay operator. It was used to memorize the output value of the hidden layer neuron at the previous moment. Thus, the output of the context layer neuron is stored and then inputted to the hidden layer. It enhances the model's ability to process dynamic information.

In this study, a simple three-layer Elman network was adopted. Figure 4.5 shows the structure of the Elman network. The relationships between the neurons in different layers of the network can be expressed as:

$$s(t) = f\left(w_1 x_c(t) + w_2(x(t-1))\right) \tag{3}$$

$$x_c(t) = s(t-1) \tag{4}$$

$$y(t) = x(t+5) = g(w_3 s(t)) \tag{5}$$

where, t represents time and y, s, x, x_c represent the output neuron vector, hidden layer neuron vector, input neuron vector, and context neuron vector, respectively. w_3 is the connection weight matrix from the hidden layer to the output layer, w_2 is the connection weight matrix from the input layer to the hidden layer, and w_1 is the connection weight matrix from the context layer to the hidden layer, respectively. f (.) is the transfer function of the hidden layer neurons, using the tansig function, and g (.) is the output layer transfer function, using the tansig function. In this study, the Elman neural network uses the optimized gradient descent algorithm for training. Through learning and training, the difference between the actual output value and the output value of the network is used to continuously modify the weights and thresholds, so that the sum of squares of errors at the output layer of the network is minimized.

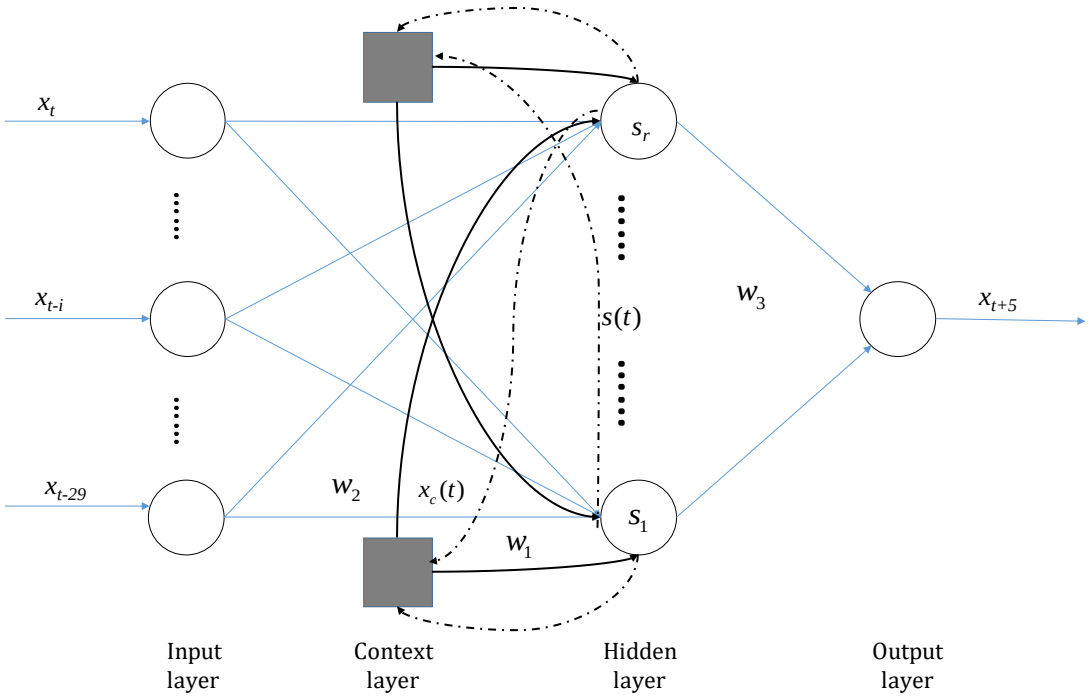


Figure 4.5: Structure Diagram of Elman Network

4.6 Improved K-Nearest Neighbor's Algorithm

In this research, an improved KNN model has also been developed by using weighted distance measurement, optimizing the K value, and improving the prediction algorithm.

4.6.1 Weighted Distance Measurement

The model developed in this research is to forecast the vehicle arrival rate at the intersection 30 min later based on the arriving rates in the previous 3 h. Therefore, the prediction model can be mathematically expressed as follows.

$$f\left(x_{t-29,}\ldots x_{t-1}, x_t\right) = x_{t+5} \tag{6}$$

where,

t is the current time interval;

x_t is the arrival traffic flow rate during the current time interval.

Since the traffic flow rate is at a 6 min interval, the vector (x_{t-29}, ... x_{t-1}, x_t) represents the arrival travel flow rates during the previous 3 h and x_{t+5} represents the predicted traffic flow rate that will arrive at the intersection in half an hour. According to Habtemichael and Cetin (Habtemichael and Cetin, 2016), the time factor should be considered in the traffic flow prediction, which means when calculating the similarity between current and historical traffic

flow data, more weight should be given to the more recently collected traffic flow data. According to this idea, the following weighted Euclidean distance is used:

$$d_{ij} = \sqrt{\sum_{t=T-29}^{T} \omega_t \times \left(x_{it} - y_{jt}\right)^2} \tag{7}$$

$$\omega_t = \frac{W_{t,norm}}{\sum_{t=T-29}^{T} W_{t,norm}} \tag{8}$$

where,

x_{it} is the number of vehicles arriving at the t[th] time interval on the i[th] day in the historical dataset;

x_{jt} is the number of vehicles arriving at the t[th] time interval on the j[th] day in the prediction dataset;

ω_t is a time-related weight coefficient;

$W_{t,norm}$ is the normalized temporal distance between the endpoint of t[th] time interval and the prediction time point, which can be expressed as follows:

$$W_{t,norm} = \frac{W_t - W_{\min}}{W_{\max} - W_{\min}} \tag{9}$$

where,

W_t is the temporal distance between the endpoint of tth time interval and the prediction time point (in the number of time intervals as the unit);

W_{max} is the longest temporal distance from the prediction time point;

W_{min} is the shortest temporal distance from the prediction time point.

4.6.2 Optimized K Value

Based on the distance calculated in Equation (2), the K nearest neighbors (the K historical days that have the traffic conditions most similar to the traffic condition at the targeted time t of the prediction day) can be selected. In the basic KNN model that was developed, a given k value (K = 10) was used. To improve the model prediction, in this study, different K values from 7 to 15 were tested and the K values that resulted in the lowest prediction error were selected for predicting the traffic flow rate at the study intersection.

4.6.3 Improved Prediction Algorithm

In the basic KNN model developed by, the average traffic flow rate of the selected K days was used for prediction. In this study, the weighted average method is used, and the neighboring distance is used as the weight. The basic idea is that if the traffic condition of the selected day is more similar to the predicted day, it should contribute more to the predicted traffic flow rates. Thus, the weighting coefficient of each neighbor can be calculated by Equation (10).

$$w_i = \frac{1/d_{ij}}{\sum 1/d_{ij}} \tag{10}$$

where d_{ij} represents the weighted Euclidean distance between the ith similar historical day and the prediction day (*j*th day) and is calculated by using Equation (7). Then, the predicted traffic flow at the given time *t+5* can be estimated using Equation (11).

$$\hat{x}_{t+5} = \sum_{i=1}^{k} w_i x_{i(t+5)}^{*}, \tag{11}$$

where $x^{*}_{i(t+5)}$ represents the number of vehicles arriving 30 min after the target time *t* during the *i*th historical day that was one of the selected K nearest neighbors.

4.7 Integrated Prediction Models Based on Entropy Weight Methods

The three different EWM methods that we introduced before will be used for integrating the two individual models, i.e., improved KNN and Elman models. Following are the steps to develop these three EWMs-based integrated models.

4.7.1 Entropy Weight Method A (EWM-A)

As mentioned in the literature review section, the EWM-A method is based on the idea that the smaller the entropy value of the prediction error of an individual model, the greater the weight should be assigned to it and vice versa. According to Bai et al. (Bai et al., 2020), by using the EWM-A method, the two selected individual models can be integrated through the following process:

Step 1: Calculate the absolute error weight of the individual model at time t by Equation (12).

$$p_{st} = \frac{|e_{st}|}{\sum_{t=1}^{m} |e_{st}|} \quad (s = 1, 2, \cdots, n; t = 1, 2, \cdots, m) \tag{12}$$

where,

$e_{st} = |\hat{y}_{st} - y_t|$,

s indicates different models,

n is the number of individual models (*n* = 2 in this study),

t represents the time, *m* is the number of prediction time points,

$\hat{y}_{st}$ is the predicted value of the sth individual model at time *t*,

y_t is the observed value.

Step 2: Calculate the entropy value of the sth individual model:

$$H_s = -k\sum_{t=1}^{m} p_{st} \ln p_{st} \quad (s = 1, 2, \cdots, n) \tag{13}$$

If $P_{st} = 0$, then $P_{st} \ln P_{st} = 0$, $k = \frac{1}{\ln m}$

Note that, according to the entropy concept, P_{st} in Equation (13) should be a probability of an event. However, according to Equation (12), P_{st} is a ratio of a prediction error to the sum of prediction errors instead of a probability. This is a critical problem with this type of EWM and will be discussed more in the model evaluation part.

Step 3: Calculate the weight of the sth individual model:

$$\omega_s = \frac{1 - H_s}{n - \sum_{s=1}^{n} H_s} \quad (s = 1, 2, \cdots, n) \tag{14}$$

In this study n = 2, thus, ω_s becomes:

$$\omega_s = \begin{cases} \dfrac{1 - H_1}{2 - H_1 - H_2} & s = 1 \\ \dfrac{1 - H_2}{2 - H_1 - H_2} & s = 2 \end{cases} \tag{15}$$

Note that, $0 \le \omega_s \le 1$, $\sum_{s=1}^{n} \omega_s = 1$.

Step 4: Integrate the predictions of individual models based on the calculated weights:

$$\hat{Y} = \sum_{s=1}^{n} \omega_s \hat{y}_s \tag{16}$$

where $\hat{y}_s$ is the predictions of the sth individual model.

4.7.2 Entropy Weight Method B (EWM-B)

Different from the EWM-A method, the EWM-B method is based on the idea that if an individual prediction model has a smaller entropy value of the prediction error, the variation degree and uncertainty in this model are greater, thereby a smaller weight coefficient should be assigned to this individual model. According to Huang et al. (Huang et al., 2017), the procedure of integrating the developed improved KNN model and Elman model based on EWM-B are as follows.

Step 1: Calculate the relative error weight of the individual prediction model:

$$p_{st} = \frac{|e_{st}|}{\sum_{t=1}^{m} |e_{st}|} \quad (s = 1, 2, \cdots, n; t = 1, 2, \cdots, m) \tag{17}$$

where,

$e_{st} = |\hat{y}_{st} - y_t|$,

s indicates different models,

n is the number of individual models (n = 2 in this study),

t represents the time,

m is the number of prediction time points,

$\hat{y}_{st}$ is the predicted value of the *s*th individual model at time t,

y_t is the observed value.

Step 2: Calculate the entropy value of the *s*th individual model:

$$H_s = -k \sum_{t=1}^{m} p_{st} \ln p_{st} \quad (s = 1, 2, \cdots, n) \tag{18}$$

If $P_{st} = 0$, then $P_{st} \ln P_{st} = 0$, $k = \frac{1}{\ln m}$

Step 3: Calculate the variation degree of the *s*th model:

$$D_s = 1 - H_s \quad (s = 1, 2, \cdots, n) \tag{19}$$

where, 0 < Hs < 1

Step 4: Calculate the weight coefficient of the *s*th individual model:

$$\omega_s = \frac{1}{n-1}\left(1-\frac{D_s}{\sum_{s=1}^{n} D_s}\right) \quad (s=1,2,\cdots,n) \tag{20}$$

Note that, in this study n = 2, thus:

$$\omega_s = 1-\frac{D_s}{\sum_{s=1}^{2} D_s} = 1-\frac{1-H_s}{2-H_1-H_2} = \begin{cases} \dfrac{1-H_2}{2-H_1-H_2} & s=1 \\ \dfrac{1-H_1}{2-H_1-H_2} & s=2 \end{cases} \tag{21}$$

Compared with the weight coefficients of EWM-A given in Equation (16), it can be seen that the weight coefficients of two individual models are simply swapped in EWM-B.

Step 5: Integrate the predictions of individual models based on the calculated weights:

$$\hat{Y} = \sum_{s=1}^{n} \omega_s \hat{y}_s \tag{22}$$

where,

$\hat{y}_s$ is the predictions of the sth individual model

4.7.3 Entropy Weight Method C (EWM-C)

In information theory, entropy is a measure of the uncertainty associated with a random variable. In the model integration, if we calculate the entropy based on the relative error of the individual prediction model as shown in Equation (12), both low and high accuracy of prediction models could all lead to a small entropy value because the error is relative to other errors. To address this problem, Shan and Zhang (Shan and Zhang, 2021) proposed to use a new EWM-based method (EWM-C) for model integration to take into account the prediction accuracy levels of the individual models. In this method, they used a weighted entropy of the model prediction error, and the prediction accuracy level of the individual model was incorporated into this weighted entropy. In this way, the impact of the model with low accuracy can be reduced and the prediction accuracy of the integrated model can be improved. Following is the detailed procedure for integrating the prediction models using the EWM-C method.

Step 1: Calculate the prediction accuracy of the sth individual model:

$$a_{st} = 100\%\left(1-\left|\frac{y_t-\hat{y}_{st}}{y_t}\right|\right) \quad (s=1,2,\cdots,n;t=1,2,\cdots,m) \tag{23}$$

where,

a_{st} is the prediction accuracy of the sth individual model at time t,

s indicates different models,

n is the number of individual models (n = 2 in this study),

t represents the time, m is the number of prediction time points,

$\hat{y}_{st}$ is the predicted value of the sth individual model at time t,

y_t is the observed value.

Step 2: Establish the matrix of model prediction accuracy

Then, the matrix of the prediction accuracy of different individual models can be expressed as follows:

$$A_{nm} = \begin{bmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{bmatrix} \tag{24}$$

Note that, the row vector $A_s = (a_{s1}, a_{s2}, \ldots, a_{sm})$ represents the accuracy of the sth individual model $S = (1,2, \ldots, n)$.

Step 3: Establish the matrix of accuracy level frequency

First, round the number in the matrix A_{nm} down to its integer (for example, 87.15% rounded down to 87%). Then, by counting the number of different accuracy levels, the following matrix of the accuracy level frequency can be established.

$$R_{nm} = \begin{bmatrix} \mathrm{r}_{11} & \cdots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{n1} & \cdots & r_{nm} \end{bmatrix} \tag{25}$$

where r_{st} represents the number of occurrences of a_{st} (integer part) in the row s.

Step 4: Calculate the weighted information entropy of the sth model

Then, the weighted information entropy of the sth model, i.e., Es, can be calculated by Equation (26).

$$E_s = -\sum_{t=1}^{m} w_{st} p_{st} \log p_{st} \quad (s = 1, 2, \cdots, n) \tag{26}$$

where,

$$p_{st} = \frac{r_{st}}{\sum_{t=1}^{m} r_{st}} \tag{27}$$

$$w_{st} = \begin{cases} 1 & a_{st} < X\% \\ 1 - \frac{N_{st}}{\sum_{t=1}^{m} N_{st}} & a_{st} \geq X\% \end{cases} \tag{28}$$

N_{st} is the number of a_{st} greater than the accuracy level $X\%$ in the sth row in matrix A (in this study $X\%$ = 80%).

Step 5: Calculate the weight coefficient of the *s*th individual model:

The weight coefficient of the individual model can be calculated based on the Es calculated in Step 4 as follows:

$$\omega_s = \frac{1}{ZE_s} \ (s = 1, 2, \cdots, n) \tag{29}$$

where,

Z is a normalization factor that ensures that all weights sum to 1.

Thus, when n = 2, the weight of the two individual models can be calculated as:

$$\omega_s = \begin{cases} \frac{E_2}{E_1 + E_2} & s = 1 \\ \frac{E_1}{E_1 + E_2} & s = 2 \end{cases} \tag{30}$$

Step 6: Integrate the predictions of individual models based on the calculated weights:

$$\hat{Y} = \sum_{s=1}^{n} \omega_s \hat{y}_s \tag{31}$$

where,

$\hat{y}_s$ is the predictions of the sth individual model

According to the three different EWM-based methods introduced above, different integrated models were developed for each day of the week except Tuesday. The weight coefficients estimated by using different EWM-based methods are presented in Table 4.2.

Table 4.1: KNN Model and Elman Model Weight Distribution Table

	Weight	Wed.	Thu.	Fri.	Sat.	Sun.	Mon.
EWM-A	ω_1	0.5579	0.5955	0.6189	0.5280	0.5277	0.5599
	ω_2	0.4421	0.4045	0.3811	0.4720	0.4723	0.4401
EWM-B	ω_1	0.4421	0.4045	0.3811	0.4720	0.4723	0.4401
	ω_2	0.5579	0.5955	0.6189	0.5280	0.5277	0.5599
EWM-C	ω_1	0.5673	0.5974	0.5738	0.5052	0.4954	0.5602
	ω_2	0.4327	0.4026	0.4262	0.4948	0.5046	0.4398

Note: ω_1 represent the weights of the improved KNN model. ω_2 represent the weights of the Elman model.

Chapter 5. Model Evaluation

The developed models were applied to the validation dataset, which is to predict the traffic flow rate (number of vehicles per 6 min) during one week from March 37, 2019 (Wednesday) to April 1, 2019 (Monday). The prediction starts at 3:30 am every day because the data collected during the first three hours from 0:00 am–3:00 am were used as the model inputs for the first prediction at 3:30 am. Thereafter, as the time window moves, a prediction will be generated every 6 min. In this chapter, the performances of single models are compared first, then the performance of three EWM-based integrated models is compared with two selected single models with better performance.

5.1 Comparison of the Four Individual Models

As an example, the prediction results for one weekday (March 27, 2019) and one weekend (March 31, 2019) are present in Figures 5.1 and 5.2. Figure 5.1 and Figure 5.2 show the observed traffic flow and the forecasted traffic flows with developed single models. It can be seen that KNN performs best sometimes. To compare the overall performance of different models, two quantitative performance measures were used.

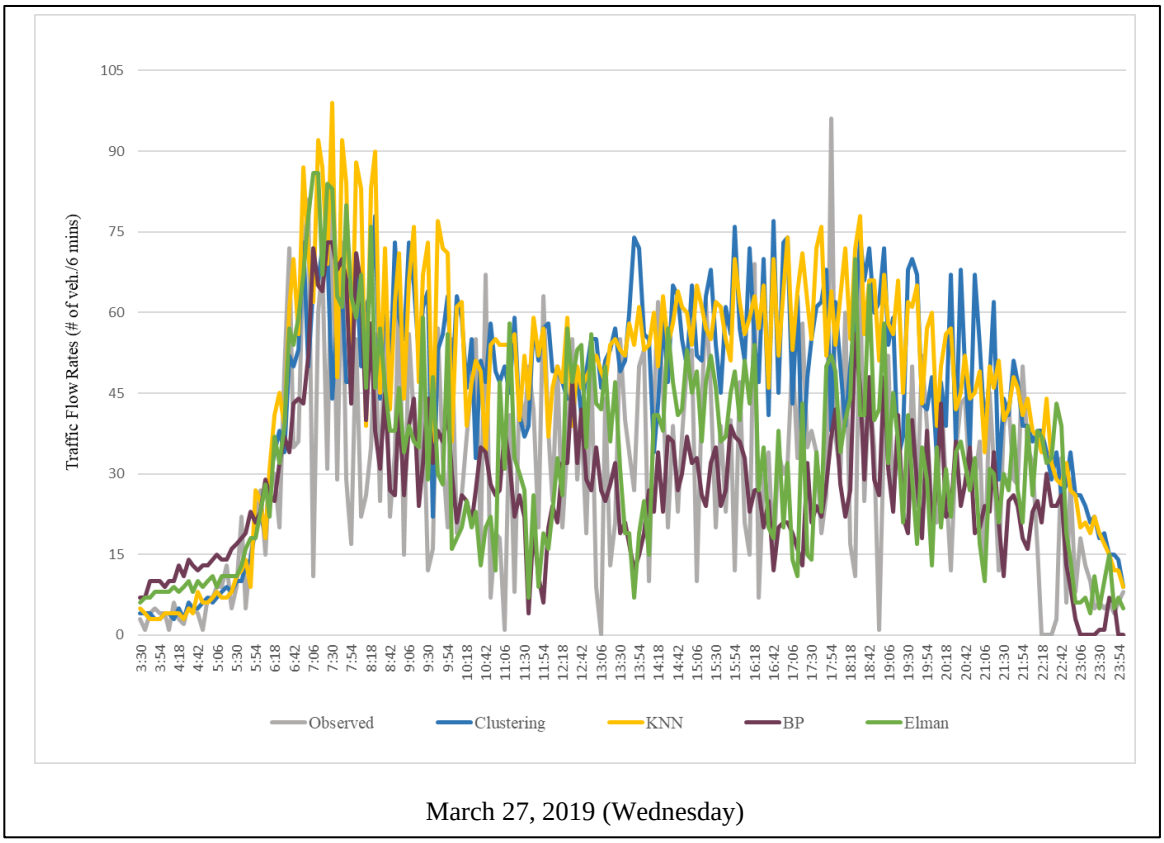


Figure 5.1: Traffic Flow Prediction for a Weekday

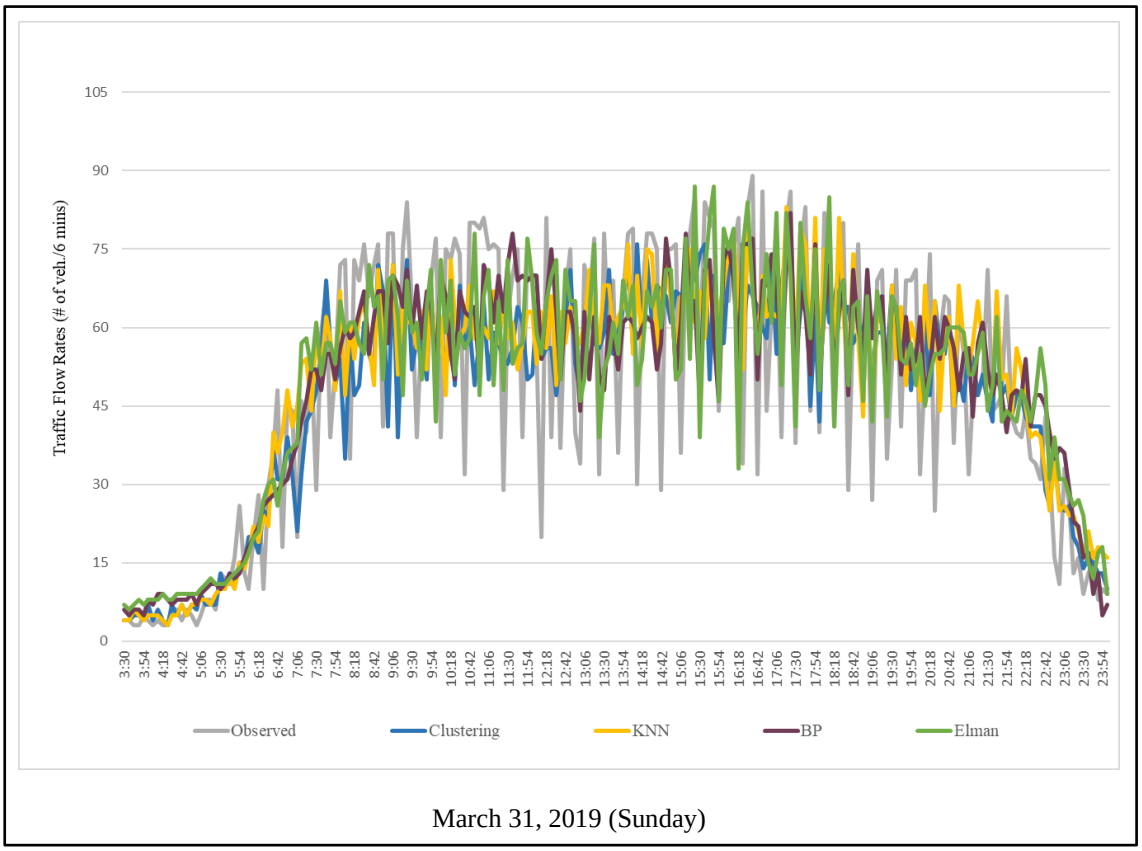


Figure 5.2: Traffic Flow Forecast for a Weekend.

To evaluate the prediction accuracy of different models, two measures were used: (1) the Root Mean Squared Error (RMSE), and (2) the Correlation Coefficient between the Predicted and Observed values (CCPO). The RMSE measures the differences between the predicted and observed values, which can be expressed by the following equation:

$$RMSE = \sqrt{\frac{(\hat{y}_t - y_t)2}{T}} \tag{32}$$

where, $\hat{y}_t$ is the predicted traffic flow rate, y_t is the observed traffic flow rate and T denotes the total number of time intervals during the prediction period. A lower RMSE value indicates a better prediction. The CCPO measures the correlation between the predicted and observed traffic flow rates (CCPO). A higher CCPO value indicates a better prediction. It is because a high CCPO value means that the predicted values and the observed values have the same trend of change.

According to these two measures, the performances of the developed models are compared and presented in Figures 5.3 and 5.4.

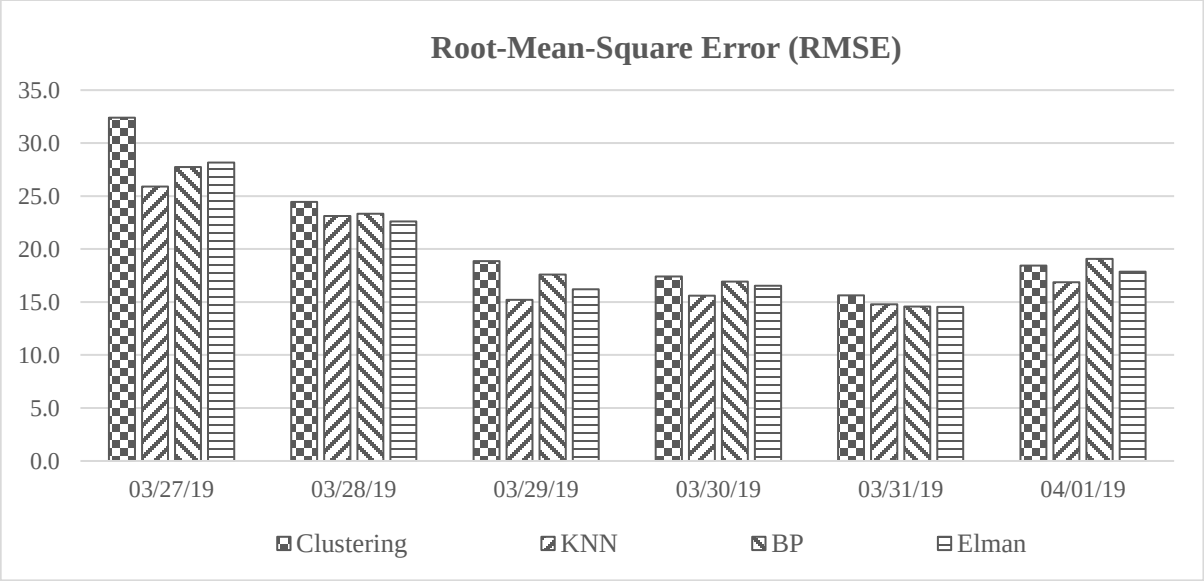


Figure 5.3: Comparison of RMSEs of Different Models

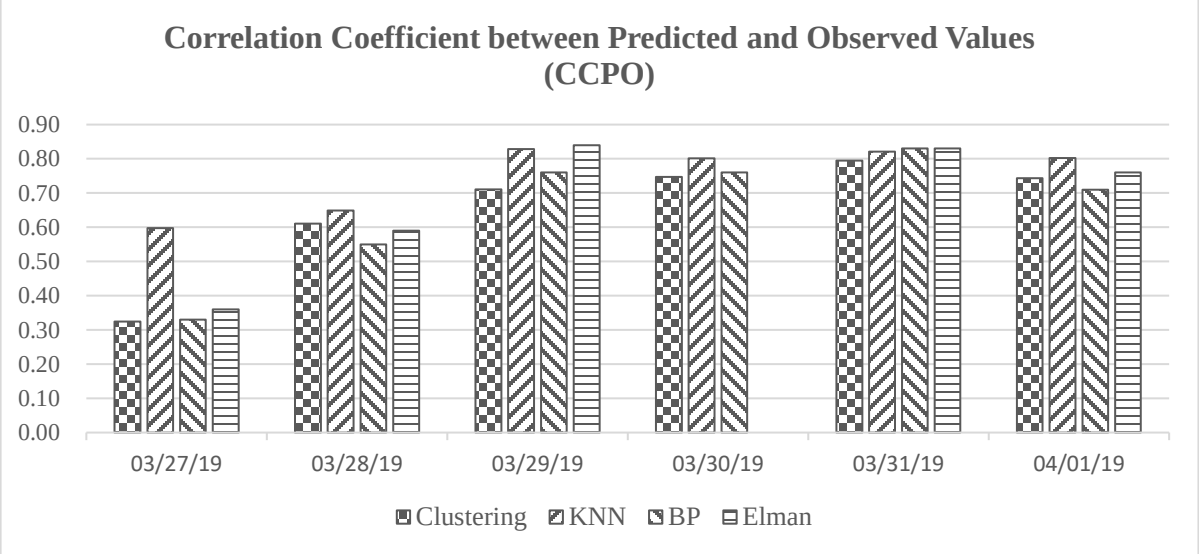


Figure 5.4: Comparison of CCPOs of Different Models

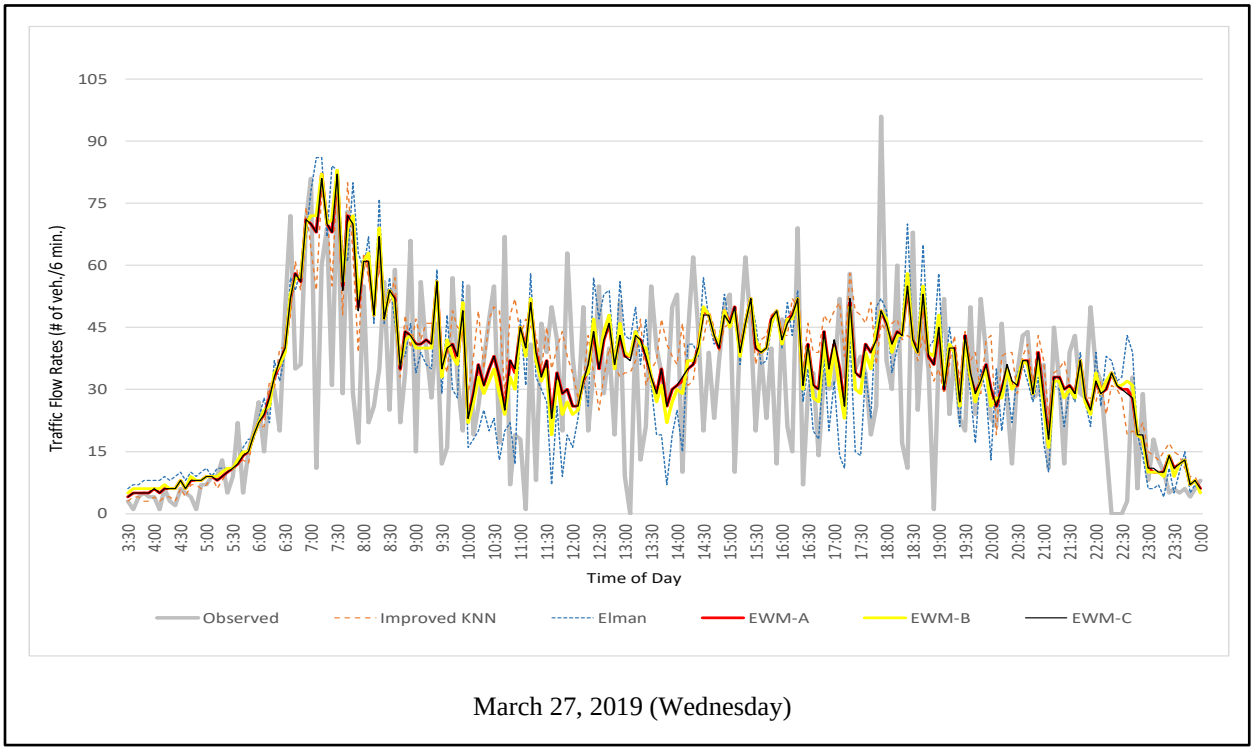
From Figures 5.3 and 5.4, it can be seen that the KNN model consistently outperforms the clustering model, which is consistent with the findings in the literature (Hou et al., 2018; Yu et al., 2016). Also, by comparing the BP model with the Elman model, it was found that the Elman model has better performance than the BP model in almost all six days. This result is also reasonable because the Elman model enhanced the BP model by adding a context layer that feeds back the hidden layer outputs in the previous timesteps. Table 5.1 lists the overall six-day average performance of these five models in terms of their RMSE and CCPO values.

Table 5.1: Comparison of 6-day Average RMSEs and CCPOs of Different Models

	Clustering	KNN	BP	Elman
Root-mean-square error	21.203	18.578	19.875	19.328
Correlation coefficient	0.66	0.75	0.66	0.68

5.2 Comparison of Improved KNN, Elman, and Three EWM-based Integrated Models

For model evaluation purposes, the developed improved KNN model, Elman model, and the three EWM-based integrated models were also applied to the test date, which includes 6 days of traffic flow data collected from 27 March 2019 to 1 April 2019 (Wednesday to Monday). The prediction starts at 3:30 am on each day and after that, a prediction is generated every six minutes. Figure 5.5 and Figure 5.6 show the predicted traffic flow rates of different models on 27 March 2019 (Wednesday) and 31 March 2019 (Sunday) respectively, along with the observed traffic flow rates on these two days. It can be seen that the traffic flow at this intersection fluctuates more during the weekday. The traffic remains heavy during the weekend while there is an obvious morning peak during the weekdays.



March 27, 2019 (Wednesday)

Figure 5.5: Traffic Flow Predictions for a Weekday

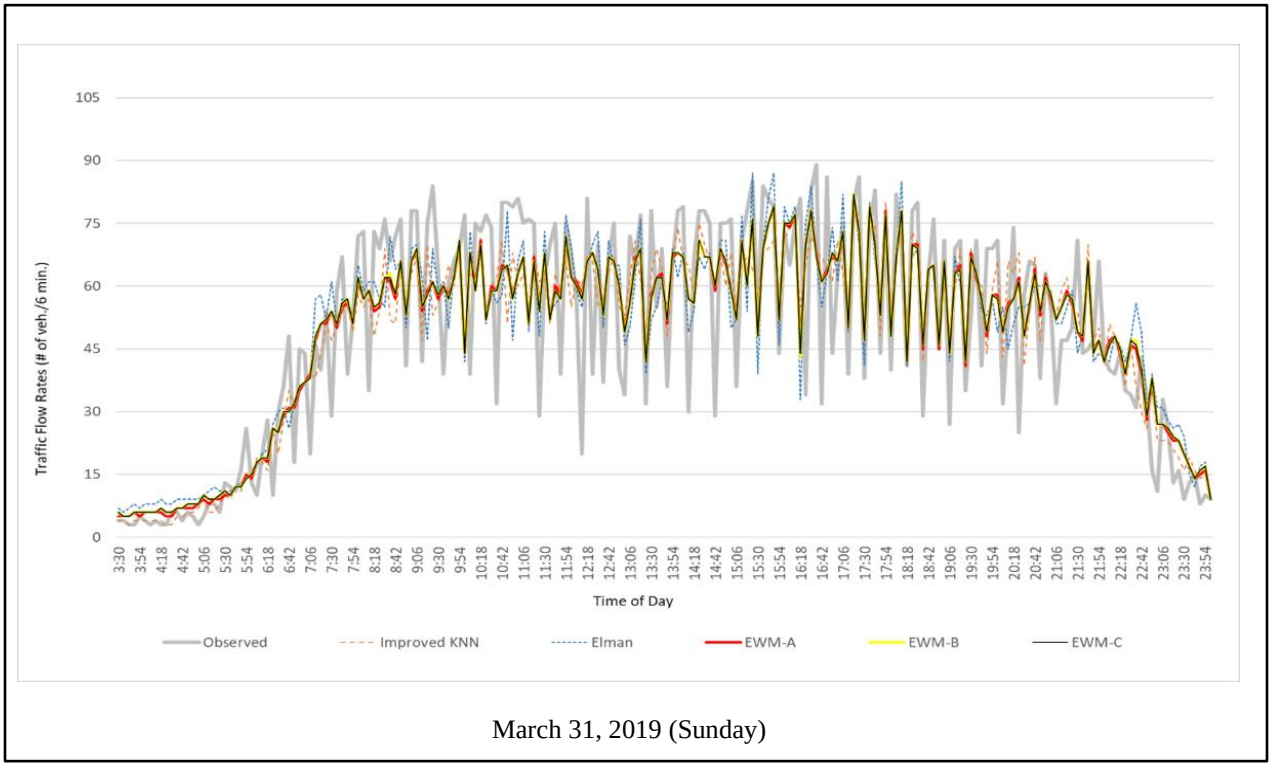


Figure 5.6: Traffic Flow Predictions for a Weekend

Figure 5.5 and Figure 5.6 show that overall, the integrated models can predict the trend of traffic flow rate very well. The prediction results of the two individual models have more variance than that of the integrated models. The predicted traffic flow rates of the three integrated models are in the middle of the predicted values of the two individual models. This proves that the integrated model combines the predictions of the improved KNN model and the Elman model.

Next, a performance measure called Mean Square Error (MSE) was used to evaluate the prediction accuracy. MSE measures the differences between the predicted traffic flow rate and observed data and can be calculated as follows:

$$MSE = \frac{\sum_{s=1}^{n} (\hat{y}_s - y_s)^2}{n}, \tag{33}$$

where,

$\hat{y}_s$ represents the predicted traffic flow rate in the sth time interval;

y_s represents the observed traffic flow rate in the sth time interval;

n represents the total number of time intervals in the forecast period.

A smaller MSE value represents a better model performance. MSEs of the models developed for different days are calculated and presented in Table 5.2. In addition, the results for

the three traffic flow prediction models, i.e., Basic BP and KNN were also included in Table 5.2 for comparison purposes.

Table 5.2: Comparison of MSE of Different Models

Model	3.27 (Wed.)	3.28 (Thu.)	3.29 (Fri.)	3.30 (Sat.)	3.31 (Sun.)	4.1 (Mon.)	Average
BP	769.8010	544.7767	309.5437	286.7621	212.2913	363.5728	414.4679
Elman	794.0899	511.1533	262.9230	273.4558	211.9528	319.3155	395.4817
KNN	670.9806	534.8592	231.0146	**243.3155**	218.4417	284.1707	363.7971
Improved KNN	310.5000	378.8010	226.3252	298.1456	187.6456	256.7816	276.3665
EWM-A	308.9466	372.3883	208.5777	253.0825	**179.5485**	248.2718	261.8026
EWM-B	320.8932	398.7524	215.8252	250.0631	181.1650	255.8544	270.4256
EWM-C	**307.3204**	**371.4563**	**208.4223**	253.0146	180.6845	**248.2718**	**261.5283**

Table 5.2 shows the improved KNN model outperforms the Elman model on most days. It was also found that the three EWM-based integrated models have better prediction accuracy than the individual models in most cases. This is reasonable because the integrated model can utilize the information provided by both individual models, which leads to improved model prediction accuracy. From Table 5.2, it can also be seen that, overall, the developed EWM-based integrated models outperform all three models developed in our previous study. In Table 5.2, we use bold numbers indicating the best predictions for different days of the week. It is clear that the performance of the integrated model developed using EWM-C is the best on most days and has the lowest average MSE. The accuracy level of the model developed using EWM-A is slightly lower than the one developed using EWM-C. Among the three integrated models, the EWM-B method has the worst performance, and it even performs worse than the individual model (improved KNN model) on Wednesday and Thursday (marked in red). The common problem with the EWM-B and EWM-A methods is that the P_{st} in entropy is defined as the ratio of a prediction error to the sum of prediction errors in this model (please see Equation (12)). Thus, if the error in the prediction model increases proportionally, its P_{st} will not change. In other words, the prediction errors e_{st} and $100e_{st}$ will result in the same P_{st} and same weight coefficients, which is unreasonable. In addition, according to the definitions of entropy, the P_{st} should be a probability instead of a proportion of overall prediction errors. On the other side, in the EWM-C method, the P_{st} is defined as the probability of the prediction error at a given accuracy level (please see Equation (27)). This definition of P_{st} avoids the problem in EWM-A and EWM-B. In addition, the model accuracy level was directly considered in the weight coefficients given in Equation (28). Thus, more weight will be given to the model with a higher accuracy level, and thereby, the integrated model predictions are more likely to be more accurate than those of the individual models.

Chapter 6. Conclusions and Limitations

6.1 Conclusions

In this study, four single models, one improved single model and three integrated models were developed to predict short-term traffic flow at signalized intersections. The five single models selected for this study included clustering, KNN, BP, Elman, and improved KNN models. Entropy weight method was used to integrate two individual models, which were improved KNN and Elman models. Three different types of entropy weight methods, i.e., EWM-A, EWM-B, and EWM-C, were introduced and applied to develop integrated models for short-term intersection traffic flow prediction. A signalized intersection in Jinan, China was selected as a study intersection. To evaluate the performance of the developed models, they were applied to the study intersection for forecasting six days' traffic flows. First, the performances of the four single models were evaluated. By comparing the RMSEs, CCPOs of these models, it was found that the KNN model can produce more accurate predictions than clustering, BP and Elman models at most of the time.

Next, the use of the entropy weight method for integrating individual prediction models was investigated. Three different types of entropy weight methods, i.e., EWM-A, EWM-B, and EWM-C, were introduced and applied to develop integrated models for short-term intersection traffic flow prediction. Two individual models, i.e., the improved KNN model, were developed at first. After that, three integrated models were developed using the three different EWMs. By comparing the performances of the developed models, it was found that the EWM-C model produced more accurate predictions than the other two integrated models. Although EWM-A and EWM-B have been used by many previous studies for model integration purposes, there is a critical problem with the definitions of entropy weight. The entropy should be defined based on the probability of prediction errors instead of the ratio of a prediction error to the sum of prediction errors. This problem will result in unreasonable weight coefficients for the models with different accuracy levels. Thus, both methods, i.e., EWM-A and EWM-B, are not recommended for integrating prediction models. On the other side, in EWM-C, entropy was defined based on the probability of the prediction error at a given accuracy level. This definition avoids the most critical problem in the EWM-A and EWM-B methods and the prediction accuracy level of the individual model was incorporated into the calculated weights. As a result, more weight will be given to the model with a higher accuracy level, which results in improved prediction accuracy. Thus, the EWM-C method was recommended for integrating prediction models.

This research work filled the gaps identified in the introduction part. First, the KNN model was developed only based on the historical days that have similar traffic pattern as the target day instead of all the historical traffic flow data. As a result, the developed model can better capture the traffic characteristics of the target day and provide more accurate predictions. Second, in this paper, the proposed model was designed for predicting the traffic flow condition in half-hour advance which can better meet the needs of advanced trip planning and traffic management. Finally, by comparing the performances of three EWMs, this research identifies EWM-C is the best method and recommend applying EWM-C for integrating models.

6.2 Limitations

Due to the data limitation, the developed model cannot be used for predicting the traffic flow conditions on a holiday. In the future, more data need to be collected for both holiday and non-holiday to develop a more comprehensive traffic flow prediction model. In addition, this research only used data collected from one intersection for developing and evaluating the proposed model. In the future, the proposed modeling method can be applied to more locations and using more data for further validation.

In addition, in this study, we only investigated the three existing EWMs. In the future, more research is needed to investigate how to improve the current EWMs to develop a better EWM for model integration purposes. For example, different thresholds for the model accuracy level in calculating the entropy for EWM-C need to be tested. In addition, the method for integrating more than two models also needs to be investigated. Furthermore, in this study, the traffic data were only collected at one signalized intersection, and due to the lack of traffic flow information on upstream and downstream intersections, the spatial factors cannot be considered in the developed model. In the future, it is necessary to collect more data from more intersections to further refine the developed model.

References

Abdulhai, B., Porwal, H., Recker, W., 1999. Short Term Freeway Traffic Flow Prediction Using Genetically-Optimized Time-Delay-Based Neural Networks.

Abu Alfeilat, H.A., Hassanat, A.B.A., Lasassmeh, O., Tarawneh, A.S., Alhasanat, M.B., Eyal Salman, H.S., Prasath, V.B.S., 2019. Effects of Distance Measure Choice on K-Nearest Neighbor Classifier Performance: A Review. Big Data 7, 221–248. https://doi.org/10.1089/big.2018.0175

Ahmed, M.S., Cook, A.R., n.d. Analysis of Freeway Traffic Time-Series Data by Using Box-Jenkins Techniques 9.

Alkasassbeh, M., Altarawneh, G.A., Hassanat, A.B.A., 2015. On Enhancing The Performance Of Nearest Neighbour Classifiers Using Hassanat Distance Metric.

Bai, H., Feng, F., Wang, J., Wu, T., 2020. A Combination Prediction Model of Long-Term Ionospheric foF2 Based on Entropy Weight Method. Entropy 22, 442. https://doi.org/10.3390/e22040442

Cai, P., Wang, Y., Lu, G., Chen, P., Ding, C., Sun, J., 2016. A spatiotemporal correlative k-nearest neighbor model for short-term traffic multistep forecasting. Transportation Research Part C: Emerging Technologies 62, 21–34. https://doi.org/10.1016/j.trc.2015.11.002

Çetiner, B.G., Sari, M., Borat, O., 2010. A Neural Network Based Traffic-Flow Prediction Model. Mathematical and Computational Applications 15, 269–278. https://doi.org/10.3390/mca15020269

Chen, Y., Li, Y., 2009. Entropy-Based Combining Prediction of Grey Time Series and Its Application, in: 2009 Second International Conference on Intelligent Computation Technology and Automation. Presented at the 2009 Second International Conference on Intelligent Computation Technology and Automation, pp. 37–40. https://doi.org/10.1109/ICICTA.2009.246

Chomboon, K., Chujai, P., Teerarassammee, P., Kerdprasop, K., Kerdprasop, N., 2015. An Empirical Study of Distance Metrics for k-Nearest Neighbor Algorithm. https://doi.org/10.12792/ICIAE2015.051

Crisler, M.C., Storf, H., 2012. A Decade of Steering Entropy - Use, Impact, and Further Application. Presented at the Transportation Research Board 91st Annual MeetingTransportation Research Board.

Dang, V.T., Dang, W.V.T., 2019. Multi-criteria decision-making in the evaluation of environmental quality of OECD countries: The entropy weight and VIKOR methods. International Journal of Ethics and Systems 36, 119–130. https://doi.org/10.1108/IJOES-06-2019-0101

Davis, G.A., Nihan, N.L., 1991. Nonparametric Regression and Short-Term Freeway Traffic Forecasting. Journal of Transportation Engineering 117, 178–188. https://doi.org/10.1061/(ASCE)0733-947X(1991)117:2(178)

Dong, C., Shao, C., Li, X., 2009. Short-Term Traffic Flow Forecasting of Road Network Based on Spatial-Temporal Characteristics of Traffic Flow, in: 2009 WRI World Congress on Computer Science and Information Engineering. Presented at the 2009 WRI World Congress on Computer Science and Information Engineering, pp. 645–650. https://doi.org/10.1109/CSIE.2009.567

Dougherty, M.S., Kirby, H.R., Boyle, R.D., 1993. THE USE OF NEURAL NETWORKS TO RECOGNISE AND PREDICT TRAFFIC CONGESTION. Traffic Engineering & Control 34.

Erlander, S., 1980. Optimal Spatial Interaction and the Gravity Model, Lecture Notes in Economics and Mathematical Systems. Springer Berlin Heidelberg, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-45515-5

Fu, G., Han, G.-Q., Lu, F., Xu, Z.-X., 2013. Short-term traffic flow forecasting model based on support vector machine regression. Huanan Ligong Daxue Xuebao/Journal of South China University of Technology (Natural Science) 41, 71–76. https://doi.org/10.3969/j.issn.1000-565X.2013.09.012

Gao, J., Zheng, D., Yang, S., 2021. Sensing the disturbed rhythm of city mobility with chaotic measures: anomaly awareness from traffic flows. J Ambient Intell Human Comput 12, 4347–4362. https://doi.org/10.1007/s12652-019-01338-7

Gao, X., Li, G., 2020. A KNN Model Based on Manhattan Distance to Identify the SNARE Proteins. IEEE Access 8, 112922–112931. https://doi.org/10.1109/ACCESS.2020.3003086

Ghosh, A.K., 2006. On optimum choice of k in nearest neighbor classification. Computational Statistics & Data Analysis 50, 3113–3123. https://doi.org/10.1016/j.csda.2005.06.007

Habtemichael, F.G., Cetin, M., 2016. Short-term traffic flow rate forecasting based on identifying similar traffic patterns. Transportation Research Part C: Emerging Technologies, Advanced Network Traffic Management: From dynamic state estimation to traffic control 66, 61–78. https://doi.org/10.1016/j.trc.2015.08.017

Hamed, M.M., Al-Masaeid, H.R., Said, Z.M.B., 1995. Short-Term Prediction of Traffic Volume in Urban Arterials. Journal of Transportation Engineering 121, 249–254. https://doi.org/10.1061/(ASCE)0733-947X(1995)121:3(249)

Han, L., Zheng, K., Zhao, L., Wang, X., Shen, X., 2019a. Short-Term Traffic Prediction Based on DeepCluster in Large-Scale Road Networks. IEEE Transactions on Vehicular Technology 68, 12301–12313. https://doi.org/10.1109/TVT.2019.2947080

Han, L., Zheng, K., Zhao, L., Wang, X., Shen, X., 2019b. Short-Term Traffic Prediction Based on DeepCluster in Large-Scale Road Networks. IEEE Transactions on Vehicular Technology 68, 12301–12313. https://doi.org/10.1109/TVT.2019.2947080

Head, K.L., 1995. EVENT-BASED SHORT-TERM TRAFFIC FLOW PREDICTION MODEL. Transportation Research Record.

Ho, F.-S., Ioannou, P., 1996. Traffic flow modeling and control using artificial neural networks. IEEE Control Systems Magazine 16, 16–26. https://doi.org/10.1109/37.537205

Hou, W., Li, D., Xu, C., Zhang, H., Li, T., 2018. An Advanced k Nearest Neighbor Classification Algorithm Based on KD-tree, in: 2018 IEEE International Conference of Safety Produce Informatization (IICSPI). Presented at the 2018 IEEE International Conference of Safety Produce Informatization (IICSPI), pp. 902–905. https://doi.org/10.1109/IICSPI.2018.8690508

Hu, L.-Y., Huang, M.-W., Ke, S.-W., Tsai, C.-F., 2016. The distance function effect on k-nearest neighbor classification for medical datasets. SpringerPlus 5, 1304. https://doi.org/10.1186/s40064-016-2941-7

Huang, S., Ming, B., Huang, Q., Leng, G., Hou, B., 2017. A Case Study on a Combination NDVI Forecasting Model Based on the Entropy Weight Method. Water Resour Manage 31, 3667–3681. https://doi.org/10.1007/s11269-017-1692-8

Ishak, S., Kotha, P., Alecsandru, C., 2003. Optimization of Dynamic Neural Network Performance for Short-Term Traffic Prediction. Transportation Research Record 1836, 45–56. https://doi.org/10.3141/1836-07
Jain, A.K., Dubes, R.C., 1988. Algorithms for clustering data. Prentice-Hall, Inc., USA.
Jiang, Y., guo, J., Zhao, J., 2013. Short-Term Traffic Flow's Forecasting by Fusing Wavelet Neural Network and Historical Trend Model. Modern Computer, 7.
Jiber, M., Lamouik, I., Ali, Y., Sabri, M.A., 2018. Traffic flow prediction using neural network, in: 2018 International Conference on Intelligent Systems and Computer Vision (ISCV). Presented at the 2018 International Conference on Intelligent Systems and Computer Vision (ISCV), pp. 1–4. https://doi.org/10.1109/ISACV.2018.8354066
Karim, A.M., Abdellah, A.M., Hamid, S., n.d. Long-term Traffic Flow Forecasting Based on an Artificial Neural Network. ASTES Journal.
Kashyap, A.A., Raviraj, S., Devarakonda, A., Nayak K, S.R., K V, S., Bhat, S.J., 2022. Traffic flow prediction models – A review of deep learning techniques. Cogent Engineering 9, 2010510. https://doi.org/10.1080/23311916.2021.2010510
Khotanzad, A., Sadek, N., 2003. Multi-scale high-speed network traffic prediction using combination of neural networks, in: Proceedings of the International Joint Conference on Neural Networks, 2003. Presented at the Proceedings of the International Joint Conference on Neural Networks, 2003., pp. 1071–1075 vol.2. https://doi.org/10.1109/IJCNN.2003.1223839
Kim, K., Pant, P., Yamashita, E., Brunner, I.M., 2012. Entropy and Accidents. Transportation Research Record 2280, 173–182. https://doi.org/10.3141/2280-19
Kindzerske, M.D., Ni, D., 2007. Composite Nearest Neighbor Nonparametric Regression to Improve Traffic Prediction. Transportation Research Record 1993, 30–35. https://doi.org/10.3141/1993-05
Koşun, Ç., Özdemir, S., 2017. An entropy-based analysis of lane changing behavior: An interactive approach.
Kou, F., Xu, W., Yang, H., 2018. Short-Term Traffic Flow Forecasting Considering Upstream Traffic Information. Presented at the 2018 International Conference on Mechanical, Electronic, Control and Automation Engineering (MECAE 2018), Atlantis Press, pp. 560–564. https://doi.org/10.2991/mecae-18.2018.86
Kuchipudi, C.M., Chien, S.I.J., 2003. Development of a Hybrid Model for Dynamic Travel-Time Prediction. Transportation Research Record 1855, 22–31. https://doi.org/10.3141/1855-03
Kumar, K., Parida, M., Katiyar, V.K., 2013. Short Term Traffic Flow Prediction for a Non Urban Highway Using Artificial Neural Network. Procedia - Social and Behavioral Sciences, 2nd Conference of Transportation Research Group of India (2nd CTRG) 104, 755–764. https://doi.org/10.1016/j.sbspro.2013.11.170
Lall, U., Sharma, A., 1996. A Nearest Neighbor Bootstrap For Resampling Hydrologic Time Series. Water Resources Research 32, 679–693. https://doi.org/10.1029/95WR02966
Li, R., Lu, H., 2009. Combined Neural Network Approach for Short-Term Urban Freeway Traffic Flow Prediction, in: Yu, W., He, H., Zhang, N. (Eds.), Advances in Neural Networks – ISNN 2009, Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, pp. 1017–1025. https://doi.org/10.1007/978-3-642-01513-7_112
Liu, H., Zhang, S., Zhao, J., Zhao, X., Mo, Y., 2010. A New Classification Algorithm Using Mutual Nearest Neighbors, in: 2010 Ninth International Conference on Grid and Cloud

Computing. Presented at the 2010 Ninth International Conference on Grid and Cloud Computing, pp. 52–57. https://doi.org/10.1109/GCC.2010.23
Liu, Y., Wu, H., 2017. Prediction of Road Traffic Congestion Based on Random Forest, in: 2017 10th International Symposium on Computational Intelligence and Design (ISCID). Presented at the 2017 10th International Symposium on Computational Intelligence and Design (ISCID), pp. 361–364. https://doi.org/10.1109/ISCID.2017.216
Liu, Z., Guo, J., Cao, J., Wei, Y., Huang, W., 2018. A hybrid short-term traffic flow forecasting method based on neural networks combined with K-nearest neighbor. PROMET-Traffic & Transportation 30.
Lopes, N., Ribeiro, B., 2015. On the Impact of Distance Metrics in Instance-Based Learning Algorithms, in: Paredes, R., Cardoso, J.S., Pardo, X.M. (Eds.), Pattern Recognition and Image Analysis, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 48–56. https://doi.org/10.1007/978-3-319-19390-8_6
Lv, Y., Duan, Y., Kang, W., Li, Z., Wang, F.-Y., 2015. Traffic Flow Prediction With Big Data: A Deep Learning Approach. IEEE Transactions on Intelligent Transportation Systems 16, 865–873. https://doi.org/10.1109/TITS.2014.2345663
Ma, W., Wang, R., 2015. Traffic flow forecasting research based on Bayesian normalized Elman neural network, in: 2015 IEEE Signal Processing and Signal Processing Education Workshop (SP/SPE). Presented at the 2015 IEEE Signal Processing and Signal Processing Education Workshop (SP/SPE), pp. 426–430. https://doi.org/10.1109/DSP-SPE.2015.7369592
Moorthy, C.K., Ratcliffe, B.G., 1988. Short term traffic forecasting using time series methods. Transportation Planning and Technology 12, 45–56. https://doi.org/10.1080/03081068808717359
Mulak, P., Talhar, N., 2013. Analysis of Distance Measures Using K-Nearest Neighbor Algorithm on KDD Dataset 4, 4.
Niu, Z., Jia, Y., Zhang, L., Liao, C., 2015. Prediction for Short-term Traffic Flow Based on Elman Neural Network Optimized by CPSO 7.
Okutani, I., Stephanedes, Y.J., 1984. Dynamic prediction of traffic volume through Kalman filtering theory. Transportation Research Part B: Methodological 18, 1–11. https://doi.org/10.1016/0191-2615(84)90002-X
Omkar, G., Kumar, S.V., 2017. Time series decomposition model for traffic flow forecasting in urban midblock sections, in: 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon). Presented at the 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon), pp. 720–723. https://doi.org/10.1109/SmartTechCon.2017.8358465
Pamuła, T., 2013. Short-Term Traffic Flow Forecasting Method Based on the Data from Video Detectors Using a Neural Network, in: Mikulski, J. (Ed.), Activities of Transport Telematics, Communications in Computer and Information Science. Springer, Berlin, Heidelberg, pp. 147–154. https://doi.org/10.1007/978-3-642-41647-7_19
Pang, X., Wang, C., Huang, G., 2016. A Short-Term Traffic Flow Forecasting Method Based on a Three-Layer K-Nearest Neighbor Non-Parametric Regression Algorithm. Journal of Transportation Technologies 6, 200–206. https://doi.org/10.4236/jtts.2016.64020
Park, B., Messer, C.J., Urbanik, T., 1998. Short-Term Freeway Traffic Volume Forecasting Using Radial Basis Function Neural Network. Transportation Research Record 1651, 39–47. https://doi.org/10.3141/1651-06

Petrov, A.I., 2022. Entropy Method of Road Safety Management: Case Study of the Russian Federation. Entropy (Basel) 24, 177. https://doi.org/10.3390/e24020177
Qu, W., Li, J., Yang, L., Li, D., Liu, S., Zhao, Q., Qi, Y., 2020. Short-Term Intersection Traffic Flow Forecasting. Sustainability 12, 8158. https://doi.org/10.3390/su12198158
Rodriguez, A., Laio, A., 2014. Clustering by fast search and find of density peaks | Science. Science 344, 1492–1496. https://doi.org/DOI: 10.1126/science.1242072
Salotti, J., Fenet, S., Billot, R., El Faouzi, N.-E., Solnon, C., 2018. Comparison of Traffic Forecasting Methods in Urban and Suburban Context, in: 2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI). Presented at the 2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 846–853. https://doi.org/10.1109/ICTAI.2018.00132
Shan, S., Zhang, S., 2021. A Weighted Hybrid Forecasting Model Based on Information Entropy. Electronic Technology & Software Engineering 5, 196–198.
Sharma, B., Kumar, S., Tiwari, P., Yadav, P., Nezhurina, M.I., 2018. ANN based short-term traffic flow forecasting in undivided two lane highway. Journal of Big Data 5, 48. https://doi.org/10.1186/s40537-018-0157-0
Shenfield, A., Day, D., Ayesh, A., 2018. Intelligent intrusion detection systems using artificial neural networks. ICT Express, SI on Artificial Intelligence and Machine Learning 4, 95–99. https://doi.org/10.1016/j.icte.2018.04.003
Sheng, J., Chen, T., Jin, W., Zhou, Y., 2021. Selection of Cost Allocation Methods for Power Grid Enterprises Based on Entropy Weight Method. J. Phys.: Conf. Ser. 1881, 022063. https://doi.org/10.1088/1742-6596/1881/2/022063
Smith, B.L., Demetsky, M.J., 1997. Traffic Flow Forecasting: Comparison of Modeling Approaches. Journal of Transportation Engineering 123, 261–266. https://doi.org/10.1061/(ASCE)0733-947X(1997)123:4(261)
Smith, B.L., Demetsky, M.J., 1994. Short-term Traffic Flow Prediction: Neural Network Approach. Transportation Research Record.
Smith, B.L., Williams, B.M., Keith Oswald, R., 2002. Comparison of parametric and nonparametric models for traffic flow forecasting. Transportation Research Part C: Emerging Technologies 10, 303–321. https://doi.org/10.1016/S0968-090X(02)00009-8
Song, X., Li, W., Ma, D., Wang, D., Qu, L., Wang, Y., 2018. A Match-Then-Predict Method for Daily Traffic Flow Forecasting Based on Group Method of Data Handling. Computer-Aided Civil and Infrastructure Engineering 33, 982–998. https://doi.org/10.1111/mice.12381
Song, X.-S., Li, H., Wu, B.-H., Li, A.-Z., 2010. Elman Neural Network Model of Traffic Flow Predicting in Mountain Expressway Tunnel, in: 2010 International Conference on Computational Intelligence and Software Engineering. Presented at the 2010 International Conference on Computational Intelligence and Software Engineering, pp. 1–4. https://doi.org/10.1109/CISE.2010.5677002
Stathopoulos, A., Karlaftis, M., 2001. Temporal and Spatial Variations of Real-Time Traffic Data in Urban Areas. Transportation Research Record 1768, 135–140. https://doi.org/10.3141/1768-16
Sun, X., Xing, H., Zhang, J., 2021. Research of combined grey model based on entropy weight for predicting anchor bolt bearing capacity. IOP Conf. Ser.: Earth Environ. Sci. 660, 012080. https://doi.org/10.1088/1755-1315/660/1/012080

Vlahogianni, E.I., Karlaftis, M.G., Golias, J.C., 2005. Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach. Transportation Research Part C: Emerging Technologies 13, 211–234. https://doi.org/10.1016/j.trc.2005.04.007

Wang, D., Zhang, Q., Wu, S., Li, X., Wang, R., 2016. Traffic flow forecast with urban transport network, in: 2016 IEEE International Conference on Intelligent Transportation Engineering (ICITE). Presented at the 2016 IEEE International Conference on Intelligent Transportation Engineering (ICITE), pp. 139–143. https://doi.org/10.1109/ICITE.2016.7581322

Wang, J., Qiao, F., Zhao, F., Sutherland, J.W., 2016. A Data-Driven Model for Energy Consumption in the Sintering Process. Journal of Manufacturing Science and Engineering 138. https://doi.org/10.1115/1.4033661

Williams, B.M., Durvasula, P.K., Brown, D.E., 1998. Urban Freeway Traffic Flow Prediction: Application of Seasonal Autoregressive Integrated Moving Average and Exponential Smoothing Models. Transportation Research Record 1644, 132–141. https://doi.org/10.3141/1644-14

Wilson, A.G., 1981. Optimization in locational and transport analysis. Wiley, Chichester.

Wu, S., Yang, Z., Zhu, X., Yu, B., 2014. Improved k-nn for Short-Term Traffic Forecasting Using Temporal and Spatial Information. Journal of Transportation Engineering 140, 04014026. https://doi.org/10.1061/(ASCE)TE.1943-5436.0000672

Xia, Y., Chen, J., 2017. Traffic Flow Forecasting Method based on Gradient Boosting Decision Tree. https://doi.org/10.2991/FMSMT-17.2017.87

Xie, L., Wu, C., Duan, M., Lyu, N., 2021. Analysis of Freeway Safety Influencing Factors on Driving Workload and Performance Based on the Gray Correlation Method. Journal of Advanced Transportation 2021, e6566207. https://doi.org/10.1155/2021/6566207

Xu, D., Tian, Y., 2015. A Comprehensive Survey of Clustering Algorithms. Ann. Data. Sci. 2, 165–193. https://doi.org/10.1007/s40745-015-0040-1

Yang, S., Wu, J., Du, Y., He, Y., Chen, X., 2017. Ensemble Learning for Short-Term Traffic Prediction Based on Gradient Boosting Machine [WWW Document]. Journal of Sensors. https://doi.org/10.1155/2017/7074143

Yin, H., Wong, S.C., Xu, J., Wong, C.K., 2002. URBAN TRAFFIC FLOW PREDICTION USING A FUZZY-NEURAL APPROACH. Transportation Research Part C: Emerging Technologies 10.

Yu, B., Song, X., Guan, F., Yang, Z., Yao, B., 2016. k-Nearest Neighbor Model for Multiple-Time-Step Prediction of Short-Term Traffic Condition. Journal of Transportation Engineering 142, 04016018. https://doi.org/10.1061/(ASCE)TE.1943-5436.0000816

Yu, S., Li, Y., Sheng, G., Lv, J., 2019. Research on Short-Term Traffic Flow Forecasting Based on KNN and Discrete Event Simulation, in: Li, J., Wang, Sen, Qin, S., Li, X., Wang, Shuliang (Eds.), Advanced Data Mining and Applications, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 853–862. https://doi.org/10.1007/978-3-030-35231-8_63

Zhang, H., Ritchie, S.G., Lo, Z.-P., 1997. MACROSCOPIC MODELING OF FREEWAY TRAFFIC USING AN ARTIFICIAL NEURAL NETWORK. Transportation Research Record.

Zhang, L., Liu, Q., Yang, W., Wei, N., Dong, D., 2013. An Improved K-nearest Neighbor Model for Short-term Traffic Flow Prediction. Procedia - Social and Behavioral Sciences,

Intelligent and Integrated Sustainable Multimodal Transportation Systems Proceedings from the 13th COTA International Conference of Transportation Professionals (CICTP2013) 96, 653–662. https://doi.org/10.1016/j.sbspro.2013.08.076

Zhang, Q., Zhu, X., Xu, K., 2011. Combination forecasting on software reliability based on entropy weight, in: Proceedings of 2011 International Conference on Electronic Mechanical Engineering and Information Technology. Presented at the Proceedings of 2011 International Conference on Electronic Mechanical Engineering and Information Technology, pp. 3095–3097. https://doi.org/10.1109/EMEIT.2011.6023742

Zhang, S., Cheng, D., Deng, Z., Zong, M., Deng, X., 2018. A novel kNN algorithm with data-driven k parameter computation. Pattern Recognition Letters, Special Issue on Pattern Discovery from Multi-Source Data (PDMSD) 109, 44–54. https://doi.org/10.1016/j.patrec.2017.09.036

Zhao, J., Gao, H., Jia, L., 2008. Short-term traffic flow forecasting model based on Elman neural network, in: 2008 27th Chinese Control Conference. Presented at the 2008 27th Chinese Control Conference, pp. 499–502. https://doi.org/10.1109/CHICC.2008.4605255

Zhao, Y., Kong, L., He, G., 2012. Entropy-based Grey Correlation Fault Diagnosis Prediction Model, in: 2012 4th International Conference on Intelligent Human-Machine Systems and Cybernetics. Presented at the 2012 4th International Conference on Intelligent Human-Machine Systems and Cybernetics, pp. 88–91. https://doi.org/DOI: 10.1109/IHMSC.2012.117

Zheng, W., Lee, D.-H., Shi, Q., 2006. Short-Term Freeway Traffic Flow Prediction: Bayesian Combined Neural Network Approach. J. Transp. Eng. 132, 114–121. https://doi.org/10.1061/(ASCE)0733-947X(2006)132:2(114)

Zheng, Z., Su, D., 2014. Short-term traffic volume forecasting: A k-nearest neighbor approach enhanced by constrained linearly sewing principle component algorithm. Transportation Research Part C: Emerging Technologies, Special Issue on Short-term Traffic Flow Forecasting 43, 143–157. https://doi.org/10.1016/j.trc.2014.02.009