Copyright ⓒ 2022 KSII

A Real Time Traffic Flow Model Based on Deep Learning

Shuai Zhang[1,2], Cai Y. Pei[2], and Wen Y. Liu[1*]
[1] School of Information Science and Engineering, Yanshan University
Qinhuangdao, 066004, Hebei, China
[2]School of Mathematics and Information Science and Technology, Hebei Normal University of Science & Technology, Qinhuangdao, 066004, China
[e-mail: 15903363750@163.com]
[*]Corresponding author: Wen Y. Liu

Received April 6, 2022; revised May 9, 2022; accepted June 1, 2022; published August 31, 2022

Abstract

Urban development has brought about the increasing saturation of urban traffic demand, and traffic congestion has become the primary problem in transportation. Roads are in a state of waiting in line or even congestion, which seriously affects people's enthusiasm and efficiency of travel. This paper mainly studies the discrete domain path planning method based on the flow data. Taking the traffic flow data based on the highway network structure as the research object, this paper uses the deep learning theory technology to complete the path weight determination process, optimizes the path planning algorithm, realizes the vehicle path planning application for the expressway, and carries on the deployment operation in the highway company. The path topology is constructed to transform the actual road information into abstract space that the machine can understand. An appropriate data structure is used for storage, and a path topology based on the modeling background of expressway is constructed to realize the mutual mapping between the two. Experiments show that the proposed method can further reduce the interpolation error, and the interpolation error in the case of random missing is smaller than that in the other two missing modes. In order to improve the real-time performance of vehicle path planning, the association features are selected, the path weights are calculated comprehensively, and the traditional path planning algorithm structure is optimized. It is of great significance for the sustainable development of cities.

Keywords: Real-time traffic data, Data flow model, Data model, deep learning, path topology, traffic flow, State discrimination algorithm

1. Introduction

At present, the highway bears more traffic pressure due to the diversification of its route interchange mode and the high service of infrastructure. Navigation map can only collect traffic information by acquiring users' basic information, but it obviously can't describe the real traffic. According to the characteristics of the expressway system, the expressway company can use its own system data to perfectly describe the real-time operation status of the road network, and provide a more reasonable and feasible vehicle path planning application [1].

With the continuous progress of signal induction theory and video analysis technology, the emergence of high-end acquisition equipment such as speed radar, high-definition camera and electronic police, which make the expressway traffic flow data acquisition system become more and more complete. The traffic flow data collected has been expanded from several items at the beginning to dozens or even hundreds at present [2]. But in fact, these traffic flow data of different breadth have not been fully used reasonably. As an important part of the intelligent transportation system, traffic flow data can efficiently provide navigation and geographic information services for travelers in real time, and can guide travelers from the original location to the target location. The selected path planning strategy directly determines the quality of the driving path provided by route guidance to travelers. According to the dynamic traffic demand, the path planning technology involved in the vehicle route guidance system not only provides accurate path search results, but also needs to be able to calculate the results in real time with the dynamic change of traffic information to prevent the failure of the obtained path planning results [3]. Optimal path planning technology uses GPS, sensors and other intelligent devices to obtain the real-time operation status of the road network. The technology analyzes the accessibility of the original node and the target node in the road network, explores the accessible path between the original node and the target node, and sets some optimal rules, such as the lowest fuel consumption, congestion avoidance, etc., to select different schemes according to the optimization rules. Then, it presents that filtered result to the user for selection [4].

Kranti Kumar et al. [5] of Vellore University of Technology applied artificial neural network to short-term prediction of traffic volume. In addition to traffic volume, speed and density, the model uses time and day of the week as input variables. The validity of the model is verified by the traffic flow data of rural roads collected through field investigation. Jin young Ahn et al. [6] of Konkuk University proposed a real-time traffic flow prediction method based on Bayesian classifier and support vector regression (SVR), and verified that the estimation method based on SVR has higher accuracy than the linear regression method by actual experiments. However, there is a potential defect when this kind of predictive method is used to interpolate missing values, that is, the data after the missing values can not be used. The interpolation performance will be greatly reduced when the missing position is relatively early.

In this paper, a dynamic path weight determination method based on deep learning is studied. The real-time traffic state is discriminated by the state discrimination algorithm based on multi-model fusion, which is used to calculate the path weight, including feature selection, multi-feature clustering, real-time classification and other core contents. And the corresponding experiment is designed for analysis to verify the effectiveness of this study.

The main innovations of this paper are:

(1) Construct a network topological structure and establish a path planning topological model which can be understood by a computer;

(2) Obtain that dynamic weight value of the path, namely calculating the path weight value base on the running state of the road network and by integrating multi-dimensional weight value factors;

(3) Optimize that structure of the traditional path planning algorithm and simplify the data storage structure.

2. Related work

2.1 State Identification Algorithm Based on Multi-model Fusion

The first step of the network topology planning is to construct the network structure with high quality, and the path planning algorithm cannot guarantee the network structure. In order to set a reasonable path weight computer system, it is necessary to obtain the current running state of the network in real time, especially the real-time status of the path in the network topology [7]. Compared with the traditional static data, stream data is difficult to be analyzed directly by standard data mining methods due to the limitations of data scale, timing, access rules, storage mechanism, response speed, randomness and so on [8]. It is necessary to add self updating mechanism to the traditional data mining model, so that it can adapt to the real-time change of stream data characteristics [9].

In this paper, a state identification algorithm based on multi-model fusion is proposed, which uses the traffic flow data of expressway to identify the real-time state of the road in the topology of highway network, which provides the theoretical basis and technical route for the determination of path weight and the subsequent application of vehicle path planning.

2.1.1 Algorithm Flow

The state discrimination algorithm based on multi-model fusion proposed in this paper consists of four modules: data pre-processing, feature selection, multi-feature clustering and real-time classification [10]. The algorithm flow is shown in **Fig. 1**.

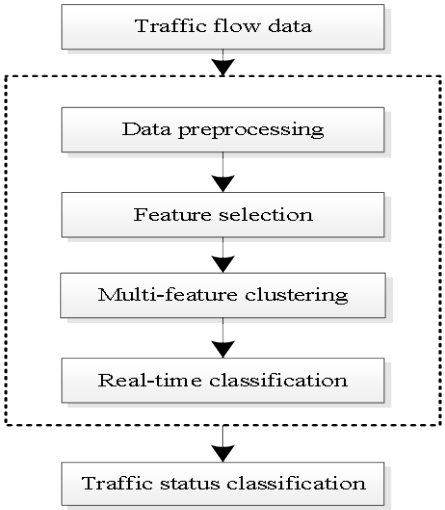

Fig. 1. Algorithm flow

（1）Data preprocessing module: The original data in the acquisition, transmission, storage process, cannot guarantee complete correctness, there must be many incomplete places, for example, data type inconsistency, data missing, data redundancy, etc. [11]. If the original data is not processed and used directly, and the low quality data is allowed to flow into the algorithm model, the learning process of the algorithm model will be greatly damaged. On the contrary, proper data preprocessing will significantly improve the quality and reliability of algorithm model decision [12].

（2）Feature selection module: with the exponential increase of data size and data complexity, it is often necessary to establish a super large algorithm structure to solve the problem, the algorithm complexity and response time increase suddenly, but in fact, most of the features (variables) are not helpful to the solution of the problem. It can be said that the process of solving the problem is redundant features (variables) [13]. This is naturally unacceptable for data itself, especially stream data, which is an infinite data stream over time. Therefore, in the process of model construction, it is very important to select effective data features (variables) to solve [14].

（3）Multi-feature clustering module: Different features have different description of the problem to be solved, which appears to be independent on the surface, but in fact has deep-seated connection. Multi-feature clustering is to divide the instance into several sub instances with obvious differences by analyzing the multi-dimensional features and using the similarity principle. In short, clustering analysis is to put the classified objects in multi-dimensional space, identify them according to the differences between them, divide the objects with the same attribute into the same class, and divide the objects with different attributes into different classes, so as to realize the "high cohesion and low coupling" between the categories, that is, the objects divided into the same class have high similarity and are divided into different classes There is a great difference between elephants [15].

（4）Real time classification module: Due to the particularity of stream data, the algorithm of the model has high real-time requirements, therefore, to establish real-time classifier in a steady stream flow data classification is very important, real-time classifier needs to be able to make quick response to flow into the model of internal data, can complete convection data classification process in a limited time, to avoid due to the large computational complexity and lead to a wide range of data queue and blocking [16].

2.1.2 Data Preprocessing

With the development of sensor technology and signal control system theory, more and more parameters used to describe traffic flow state can be collected, processed and stored by sensor equipment. At present, the acquisition of traffic flow data is mainly divided into two directions: manual acquisition technology and equipment acquisition technology [17].Manual acquisition technology is an ancient acquisition technology, which is mainly completed by manpower. It is time-consuming and laborious, and it is prone to record omissions and errors. At present, the mainstream acquisition technology is the equipment acquisition technology. As the name implies, the equipment acquisition technology refers to the automatic acquisition of traffic flow data with the help of intelligent devices such as sensor coil, radar, bayonet and electronic police [18].

When the acquisition equipment has problems in any stage of data acquisition, processing and storage, the acquired traffic flow data set will inevitably have obvious defects. If these defects are allowed to flow into the algorithm model without necessary treatment, it will

mislead the learning process of the algorithm model, seriously reduce the learning ability and greatly reduce the reliability of the model [19].Therefore, before the algorithm modeling, it is necessary to set the data preprocessing module [20]. The purpose of data preprocessing is to convert the original data input into high-quality input suitable for subsequent mining process, which usually includes integration, standardization, cleaning conversion and other technologies [21].

2.2 Data Standardization

Due to the different expression forms of different feature vectors, there are great differences in feature representation. In the subsequent modeling process, the feature vectors are easy to affect each other, affecting the discrimination accuracy of the algorithm [22]. As a result, all feature vectors are processed equally in the model[23].

Since there are few types of traffic flow eigenvectors, this paper uses the following method to normalize the eigenvectors, and all the eigenvectors are classified into [0-1][24].

(1) The maximum value Max is obtained by traversing all traffic flow eigenvectors;

(2) The minimum Min is obtained by traversing all traffic flow eigenvectors;

(3) Normalization is performed with the following equation:

$$x_{0-1}^{m} = \frac{x^{m} - Min}{Max - Min} \quad (1)$$

In equation 1, x_{0-1} is the normalized eigenvector, x is the eigenvector, Min is the minimum value of the eigenvector, and Max is the maximum value of the eigenvector[25].

2.3 Feature Selection

Feature selection is regarded as the process of selecting relevant feature subset and reducing data dimension by removing irrelevant and redundant features. Different feature vectors have different reflection angles for solving problems. The feature selection module mainly analyzes multi-dimensional features, deeply mines the meaning of different feature vectors, selects eigenvectors with high correlation with the problem to be solved, removes some irrelevant eigenvectors, and improves the efficiency and performance of the learning algorithm[26].

In this paper, based on the ability of feature vectors to distinguish the close samples, the correlation between different feature vectors and known categories in the training set is calculated, and different weights of different features are determined according to different correlations. Features whose weights are less than a certain threshold value will be deleted. To be specific, a sample S is randomly selected from the training set T, and k nearest neighbor sample Hk is found from the sample set of the same kind as S, and k nearest neighbor sample Mk is found from each sample set of different kind from S. Formula 2 updates the weight of each feature[27].

$$W(A) = W(A) - \beta sim\ ilarity_{H}(A) + \gamma difference_{M}(A) \quad (2)$$

In equation 2, A is a specific attribute feature, W(A) is the weight of the attribute feature, $\text{similarity}(A)$ is the similarity of the adjacent samples of the attribute feature, and $\text{difference}(A)$ is the difference degree of the adjacent samples of the attribute feature

$$_{H}(A) = \sum_{k}^{j=1} diff(A, S, H_{j}) / (mk) \quad (3)$$

Equation 3 calculates the sum of distances between a feature of sample S and the nearest sample Hk of the same class;

$$M(A)=\sum_{Cechas(R)}\frac{p(c)}{1-p(class(R))}\sum_{k}^{j=1}diff(A,S,M,(c)) \quad (4)$$

Equation 4 calculates the sum of distances between a feature of sample S and the nearest sample Mk of different classes.

In equation 3 and 4, $M_j(c)$ represents the j-nearest sample in class C, $diff(A,S,R)$ represents the difference between sample s and sample R on characteristic A, and its calculation equation is:

$$diff(A,S,R)=\begin{cases}\dfrac{S[A]-R[A]}{Max(A)-Min(A)} & if\ A\\ S[A]=R[A]\\ S[A]\neq R[A]\end{cases} \quad (5)$$

In equation 5, $Max(A)$ is the maximum value in the eigenvector and $Min(A)$ is the minimum value in the eigenvector.

According to the update of equation 2, when the sum of the distances from a feature of sample S to the nearest sample Hk of the same class is greater than the sum of the distances between the feature and the nearest sample Mk of different classes, the weight of the feature will be increased, that is, the feature has a positive effect on the classification of similar and non similar samples. On the contrary, when the sum of the distances between a feature of sample S and the nearest sample Hk of the same kind is the sum of the distances and when the distance is less than the sum of the distance between the feature and the nearest sample Mk of different classes, the weight will be reduced, that is, the feature has a negative effect on the classification of similar samples and non similar samples[28]. Of course, the selection of sample S may have certain randomness. Therefore, it can be repeated N times, and the average weight of each feature is taken as the final weight of the feature. If the weight of a feature is greater than 0.5, it is proved that the correlation between the feature and the problem to be solved is high; otherwise, it is proved that the correlation between the feature and the solved problem is low, especially if the weight of a feature is less than 0.5. The threshold value indicates that the feature has almost no relationship with the problem to be solved, and can be directly removed from the multi-dimensional feature vector group to achieve the purpose of feature selection[29].

Suppose that the training data set is T, the number of nearest neighbor samples is k, the number of sample sampling is n, the feature weight is W, and the threshold value of feature weight is ∂. The process is shown in algorithm 1

Algorithm 1 Feature selection algorithm

Feature selection algorithm

Input: training set T, number of nearest samples k, number of sample sampling n, characteristic threshold ∂.

Output: Wi for each feature weight

(1) $W_i = 0,\ \ W = \phi$

(2) for i = 1 to n do

K-nearest samples $H_j(j = 1, 2, 3, \ldots, k)$ are found from the similar sample set of S, and k nearest samples $M_j(c)$ is found from each different kind of sample set

(3) According to formula (4-2), the weights of all features in sample S are updated.

(4) if $w(A) \geq \partial$

Add the Ath feature to the W.

3. Depth Algorithms for Clustering Features

As the statistical distribution and probability distribution of stream data objects change with the passage of time, traditional clustering is difficult to adapt to the changing characteristics of stream data. Blind use of traditional clustering algorithm may lead to serious damage to the learning process of the model, and then wrong results can be obtained. Therefore, the traditional clustering model needs to be modified to some extent, and the established clustering model needs to be able to update the learning process adaptively with the changing characteristics of the stream data to improve the accuracy and reliability of the model.

This paper conducts multi-feature clustering analysis based on STREAM algorithm. Based on deep learning, STREAM algorithm introduces sliding window mechanism to solve problems in STREAM data clustering. The underlying framework is still a clustering algorithm.

In this paper, on the basis of K-Means algorithm, STREAM algorithm is used to realize the clustering process of the characteristics of convective data. The underlying structure algorithm of STREAM algorithm is K-Means algorithm, and batch processing mechanism is added to the superstructure to solve the problem of concept drift in the STREAM data. The STREAM algorithm flow is shown in **Fig. 2**.

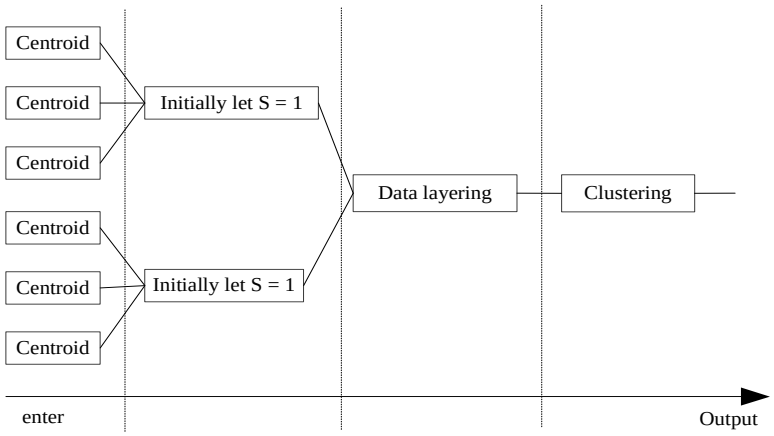

Fig. 2. STREAM algorithm flow

The STREAM algorithm flow is shown below.

(1) Initially set S=1, and calculate K S-level mass centers with k-means clustering algorithm for the initial m data.

(2) Repeat step 1 until m S-level mass centers are obtained.

(3) Calculate K S+1 mass center by using K-means clustering algorithm for m S mass centers.

(4) Repeat step 3 until m S+1 mass centers are obtained, S=S+1

(5) Repeat the above steps, that is, when m S level mass centers are obtained, K-Means algorithm is used for clustering to obtain K S+1 level mass centers; All the way to the final k centers of mass.

STREAM algorithm uses K-Means algorithm to cluster the hierarchical data. Therefore, it is necessary to discuss k-means algorithm. K-Means algorithm divides different categories according to the distribution similarity of data points in multi-dimensional feature space. Specifically, *k* objects were randomly obtained from the data set and treated as the initial center of mass of k clusters. The rest of the objects were distributed to the nearest cluster according to their Euclidean distance from each cluster mass center, and the mass center of each cluster was recalculated. The process was repeated iteratively until the distortion function converged, and *k* fixed mass center was obtained. Specifically, the process of K-Means algorithm is shown as follows.

（1）k objects are randomly obtained from the data set as the initial center $\mu_1, \mu_2, \ldots, \mu_k$ of mass of the K clusters.

（2）For each object, the Euclidean distance between it and each cluster center point is calculated, and the corresponding object is divided again according to the minimum distance. The dividing standard is shown in equation 6.

$$C^{(i)} = \arg\min \| x^{(i)} - \mu_j \|^2 \tag{6}$$

In equation 6, C(i is the category of the i-th data object, x(i) is the i-th data object, and μ_j is the jth cluster center.

（3）Update the centers of mass $u_1, u_2, ..., u_k$ of k clusters according to equation 7

$$\mu_j = \frac{\sum_{m}^{i=1} x^{(i)} \mid C^{(i)} = j}{\sum_{m}^{i=1} 1 \mid C^{(i)} = j} \tag{7}$$

（4）Repeat steps 2-3 until the distortion function of equation 8 converges to obtain k centers of mass that do not change.

$$J(c, \mu) = \sum_{m}^{i=1} x^{(i)} - \mu_c \|^2 \tag{8}$$

Where, $j(c, \mu)$ is the distortion function and μ_c is the center after the clustering is completed.

4. Real-time Classifications

Based on the decision tree theory, the real-time classification module establishes a random forest model. With the classification classes obtained from the multi-feature clustering module as the training set, the real-time traffic flow is classified and the real-time traffic state is identified. The structure is shown in **Fig. 3**.

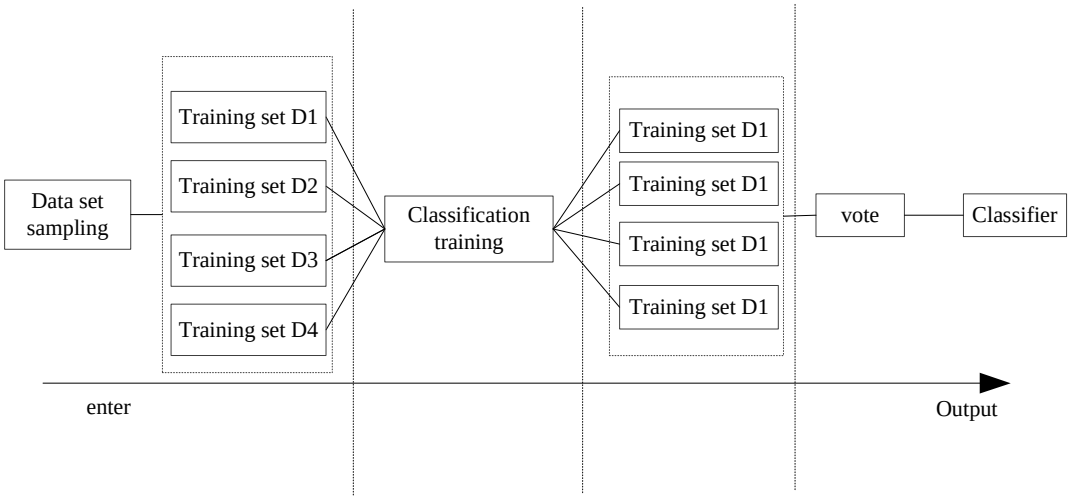

Fig. 3. Random forest algorithm

The algorithm flow is shown below.

(1) Randomly select m random samples in the sample set with put back process;

(2) For the feature set after feature selection, n features are randomly selected from the feature set to establish CART decision tree model;

(3) Repeat steps (1-2) k times to generate K CART decision trees, each with its own independent decision criteria;

(4) For the new data, through each tree decision, the category of the feature is finally determined.

5. Simulation Experiment

5.1 Data Preparation

In order to verify the effectiveness of the STREAM method, an experiment of missing value interpolation is carried out on a traffic flow database of a public data set. The data of 8 stations in a small road network were selected for the traffic flow data interpolation experiment. The small network consists of a cross-road network. The east-west trend is 100 kilometers long, and the north-south trend is 60 kilometers long. Simulated traffic flow data are collected in this area by induction coils buried under the road. Considering the time complexity of STREAM algorithm, a small network in this region is selected. The actual map of the small road network and the location and number of the detection station are shown in **Fig. 4**.

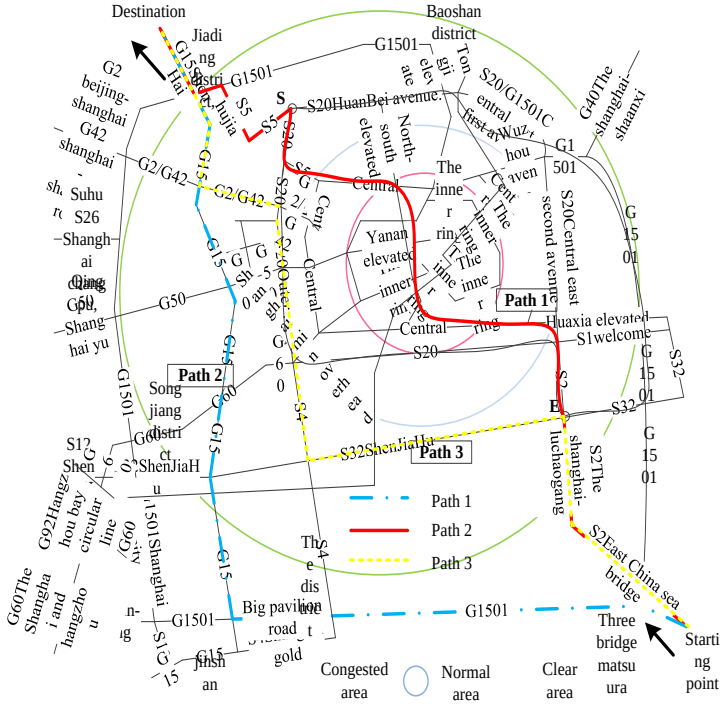

Fig. 4. Distribution map of traffic flow detection stations in a small road network

Among them, there are eight stations on the highway. The data acquisition time is from September 1, 2019 to October 31, 2020. Considering that there is a significant difference between the traffic flow pattern of holidays and normal working days, the data of weekends and holidays are excluded, and only the interpolation of traffic flow data of normal working days is studied. Finally, the 36-day data with the least data loss rate is selected, so the total number of traffic flow samples is $8 \times 36 = 288$. The detector collects data at an interval of 5 minutes, and the sample dimension is 288.

This paper simulates three data missing patterns:

(1) Missing completely at random (MCAR)

(2) Missing at random (MAR)

(3) Mixed missing（MIXED, MCAR and MAR account for 50% respectively).

5.2 Experimental Environment Configuration

On the selected traffic flow data matrix, six interpolation methods are compared, respectively

(1) The Nearest neighbor mean interpolation (TNAI) based on time correlation

(2) Mean interpolation (TAI) based on time Correlation

(3) Probability principal component Analysis (PPCA)

(4) Low-rank matrix completion (LRMC)

(5) Low-rank matrix Completion algorithm (CLRMC) based on data Correlation

(6) STREAM, the integrated learning version

Among them, the TNAI interpolation method uses the average value of the observation value closest to the position time of the missing value in the same detector to complete. TAI interpolation method uses the average value of the observation value closest to the missing value position time in different days of the same detector to estimate the missing value. TNAI

and TAI are the basic interpolation methods. It is worth noting that some recent studies show that PPCA and LRMC achieve better results in the interpolation of missing values of traffic data. The experimental hardware configuration in this paper is shown in **Table 1**.

Table 1. Experimental configuration

Configuration	Model
CPU	Intel Core i7-7900X 3.30GHz
Memory	32G DDR43000
Hard disk	SSD 1T
Operating system	Windows1064bit
Interpolation algorithm software	MATLAB R2016b

5.3 Correlation Analysis of Traffic Flow Data

The correlation between samples is the basis of missing value interpolation. In order to deeply reflect the correlation between traffic flow samples, the Pearson correlation coefficient between traffic samples collected by different detectors on different working days is calculated. Pearson correlation coefficient between sample x_i and x_j is calculated by equation 1. The Pearson correlation coefficients between all samples are shown in **Fig. 5**.

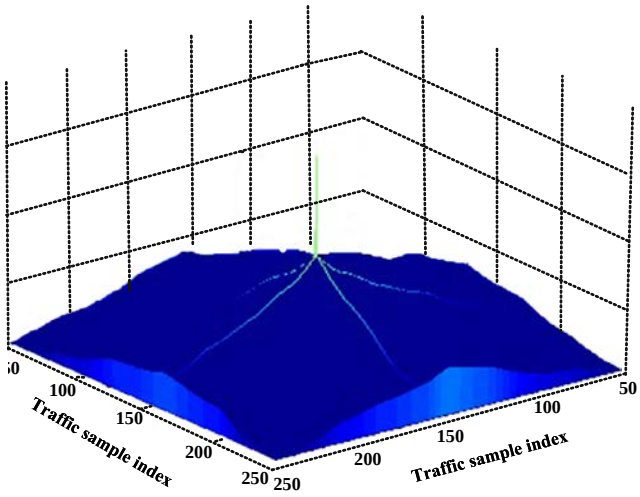

Fig. 5. Pearson's correlation matrix

The histogram of Pearson correlation coefficient is shown in **Fig. 6**.

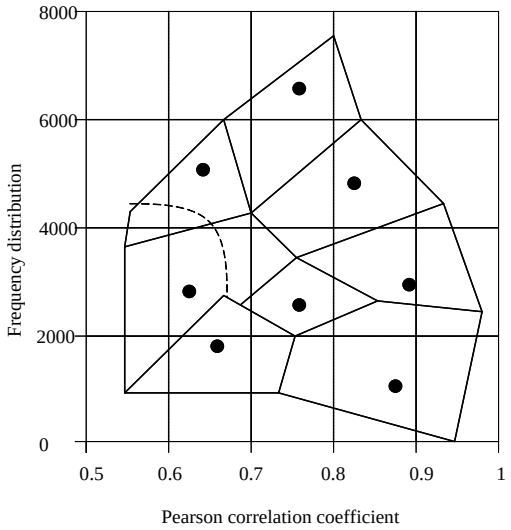

Fig. 6. Pearson correlation coefficient frequency between samples

According to the test results of **Fig. 5** and **Fig. 6**, the traffic flow data of different detectors and weekdays show strong positive correlation. This is mainly due to the regular travel behavior of commuters on weekdays, which also lays a solid foundation for the application of low rank matrix completion model in the field of traffic data. However, an important fact ignored by current research is that the Pearson correlation coefficient between traffic flow samples is actually distributed in a large range [0.4,1]. This shows that the correlation strength of traffic flow samples fluctuates greatly with different detectors and weekdays.

In order to further illustrate the correlation between the samples, a traffic sample is randomly selected as the test sample, and the Pearson correlation coefficient is used as the evaluation standard to find the most similar and least similar samples with the test sample. The results are shown in **Fig. 7** and **Fig. 8**.

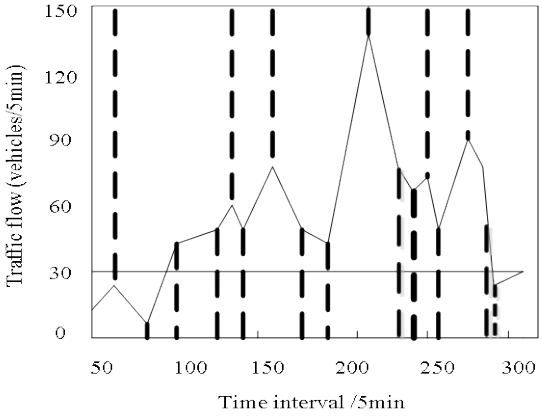

Fig. 7. Test sample of traffic flow

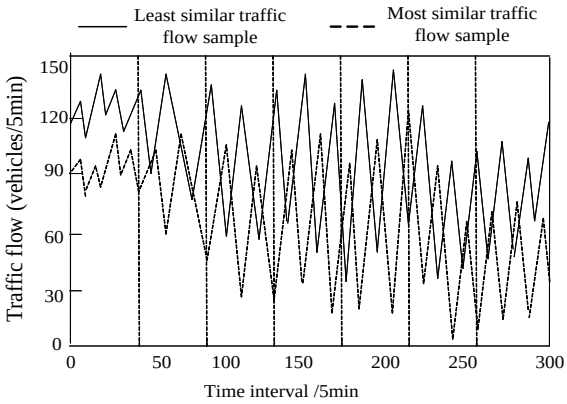

Fig. 8. Test sample comparison

The Pearson correlation coefficients between the test sample and the most similar and the most dissimilar samples are 0.9683 and 0.5100, respectively. In **Fig. 7**, for the test sample, the trend of the most similar sample and the least similar sample is very different. The results show that the most similar samples can provide more reliable information when the missing values are recovered. On the contrary, if all samples are used without distinguishing their correlations, dissimilar samples may provide unreliable information, which can adversely affect interpolation performance.

5.4 Parameter Adjustment of LRMC Integrated Learning Method Based on Data Correlation

When the weighted Pearson correlation coefficient is used to describe the similarity between two samples, the weight determines the contribution of the first round interpolation value to the weighted Pearson correlation coefficient. The larger the value is, the greater the contribution of the first round interpolation value to the weighted Pearson correlation coefficient is. In order to determine the appropriate value, after preliminary debugging, the threshold value of the adaptive K-nearest neighbor search algorithm is fixed, so that the data missing rate is and the missing mode is MCAR. Then, the interpolation errors RMSE and MAE of the interpolation methods CLRMC and STREAM will change with the weight. The results are shown in **Fig. 9**.

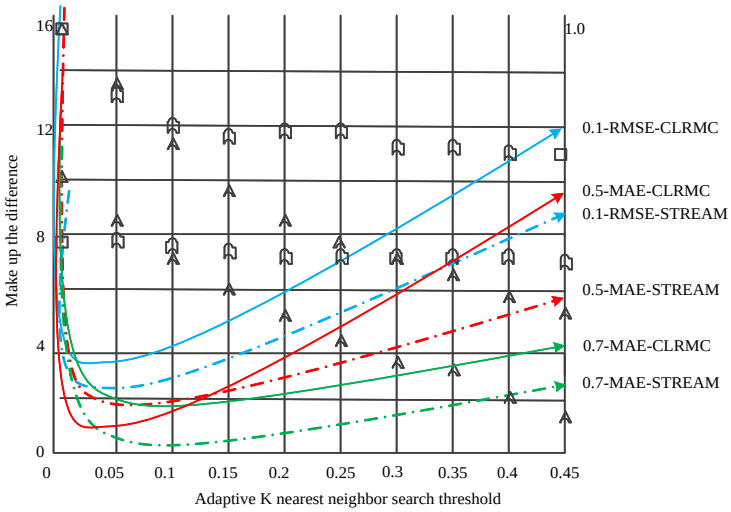

Fig. 9. Fluctuation of interpolation error

In **Fig. 9**, when the data missing rate is relatively low, the change of weight a has no significant effect on the interpolation error. This is because when the missing values are very small, the weighted Pearson correlation coefficient is mainly based on the observed values. Moreover, in this case, the first round interpolation of the whole matrix with low rank matrix completion is more accurate. With the increase of the missing rate δ, the influence of the weight a on the interpolation error becomes larger, and when a exceeds a certain critical value, the larger the α is, the worse the interpolation performance is. This is mainly because the number of observations is too small, which makes the interpolation result inaccurate. Moreover, if the weight a is large, the interpolation value contributes too much to the weighted Pearson correlation coefficient, which will mislead the results of the adaptive K-nearest neighbor search algorithm in step 3. Finally, according to the simulation results, the weight $a = 0.1$ is selected.

5.5 Example Verification Results

(1) The interpolation performance of TNAI and TNI is much worse than other methods. This is because both methods rely entirely on time dependence. In addition, TNAI performs better than TNI in the complete miss mode, but in the other two miss modes, the performance of TNAI's missing value interpolation is not as good as TNI.
(2) The interpolation performance of PPCA and LRMC is much better than that of TNAI and TNI, which shows the superiority of these two methods. In addition, PPCA had better results than LRMC in the case of relatively low miss rate. On the contrary, LRMC has better interpolation performance than PPCA when the miss rate is high. This may be because LRMC is based on the whole data, so LRMC can take advantage of more global information at a higher miss rate.
(3) As expected, CLRMC is superior to PPCA and LRMCo in different miss rates and modes, which indicates that it is very important to distinguish samples according to their intrinsic correlation degree for the missing value interpolation based on the low-rank matrix completion model.
(4) The proposed integrated learning method STREAM further reduces the interpolation error on the basis of the CLRMC interpolation method. This further suggests that multiple

interpolation results do convey useful information that should be integrated. This also proves the effectiveness of ensemble learning in matrix interpolation, which is less studied in missing value interpolation.
(5) In all interpolation methods, the interpolation error in the case of random missing is larger than that in the other two missing modes, which indicates that the continuous missing of data tests the interpolation performance of the algorithm.

6. Conclusion

Based on the real-time traffic flow data, this paper describes in detail the theoretical basis and algorithm flow of the traffic discrimination algorithm proposed in this section. The main conclusions are as follows:
(1) At the same time, depending on the real-time traffic flow data of expressway, the model is constructed.
(2) The validation experiment is set up to discuss the correctness of the algorithm model.
In the experimental process, the interpolation performance of the deep learning algorithm studied in this paper is always better than that of the CLRMC algorithm, and the STREAM algorithm is regarded as a more effective interpolation method. The interpolation error of this algorithm is significantly lower than that of other methods, which fully verifies the effectiveness of the interpolation algorithm based on deep learning.
In the next step, the network topology construction method and more reasonable node and path definition methods will be studied. The current road network topology is relatively elementary. There are fewer nodes and paths defined in the network topology, and the structural relationship is relatively simple. In the follow-up study, more new objects can be added to further optimize the network topology.

References

[1] Vassiliou, and Georgiou, “Markov and Semi-Markov Chains, Processes, Systems, and Emerging Related Fields,” *Mathematics*, vol. 9, no. 19, pp. 2490-2490, 2021. Article (CrossRef Link)
[2] Lipovetsky Stan, “Statistical and Machine-Learning Data Mining: Techniques for Better Predictive Modeling and Analysis of Big Data,” *Technometrics*, vol. 64, no. 1, pp. 145-148, 2022. Article (CrossRef Link)
[3] Sumeer Gul, Shohar Bano, and Taseen Shah, “Exploring Data Mining: Facets and Emerging Trends,” *Digital Library Perspectives*, vol. 37, no. 4, pp. 429-448, 2021. Article (CrossRef Link)
[4] Bachhal, Ahuja, and Gargrish. "Educational Data Mining: A Review,” *Journal of Physics: Conference Series*, vol. 1950, no. 1, pp. 012022-012022, 2021. Article (CrossRef Link)
[5] Accorsi Luca, Lodi Andrea, and Vigo Daniele, “Guidelines for The Computational Testing of Machine Learning Approaches to Vehicle Routing Problems,” *Operations Research Letters*, vol. 50, no. 2, pp. 229-234, 2022. Article (CrossRef Link)
[6] Shang Qiang, Feng Linlin, and Gao Song, “A Hybrid Method for Traffic Incident Detection Using Random Forest-Recursive Feature Elimination and Long Short-Term Memory Network With Bayesian Optimization Algorithm,” *IEEE Access*, vol. 9, no. 9, pp. 140538-140538, 2021. Article (CrossRef Link)
[7] Belezamo Baloka, Eken Süleyman, and Avcı Cafer, “Special Issue on Big Data in Transportation,” *Expert Systems*, vol. 39, no. 2, p. 12931, 2022. Article (CrossRef Link)
[8] Xinlan Sun, and Shiwei Lu, “Research on Road Traffic Intelligence Based on Big Data Analysis,” *International Journal of Frontiers in Engineering Technology*, vol. 3, no. 3, pp. 113-117, 2021. Article (CrossRef Link)

[9] Li Li, Rui Jiang, Zhengbing He, Xiqun Chen, and Xuesong Zhou, "Trajectory Data-based Traffic Flow Studies: A Revisit," *Transportation Research Part C: Emerging Technologies*, vol. 114, no. 3, pp. 225-240, 2020. Article (CrossRef Link)
[10] Tu Botao, Zhao Yu, Yin Guanxiang, Jiang Nan, Li Guanghui, and Zhang Yuejin, "Research on Intelligent Calculation Method of Intelligent Traffic Flow Index Based on Big Data Mining," *International Journal of Intelligent Systems*, vol. 37, no. 2, pp. 1186-1203, 2022. Article (CrossRef Link)
[11] Li Dai, and Hou Zhongsheng, "Data-driven Urban Traffic Model-free Adaptive Iterative Learning Control with Traffic Data Dropout Compensation," *IET Control Theory & Applications*, vol. 15, no. 11, pp. 1533-1544, 2021. Article (CrossRef Link)
[12] Mu Pengyu, Zhang Wenhao, and Mo Yuhong, "Research on Spatio-temporal Patterns of Traffic Operation Index Hotspots Based on Big Data Mining Technology," *Basic & Clinical Pharmacology & Toxicology*, vol. 128, no. 1, pp. 185-185, 2021. Article (CrossRef Link)
[13] Chen Bi Yu, and Kwan Mei Po, "Special Issue on Spatiotemporal Big Data Analytics for Transportation Applications," *Transportmetrica A: Transport Science*, vol. 16, no. 1, pp. 1-4, 2020. Article (CrossRef Link)
[14] Wei Dong, and Wei Sun, "Traffic Flow Prediction based on Bi LSTM and Attention," *International Core Journal of Engineering*, vol. 8, no. 3, pp. 439-444, 2022. Article (CrossRef Link)
[15] Wang Yi, and Jing Changfeng, "Spatiotemporal Graph Convolutional Network for Multi-Scale Traffic Forecasting," *ISPRS International Journal of Geo-Information*, vol. 11, no. 2, pp. 102-102, 2022. Article (CrossRef Link)
[16] Reshma Ramchandra Nazirkar, and Rajabhushanam, "Machine Learning Algorithms Performance Evaluation in Traffic Flow Prediction," *Materials Today: Proceedings*, vol. 51, no. p1, pp. 1046-1050, 2022. Article (CrossRef Link)
[17] Luo Xiaoyi, Peng Jiaheng, and Liang Jun, "Directed Hypergraph Attention Network for Traffic Forecasting," *IET Intelligent Transport Systems*, vol. 16, no. 1, pp. 85-98, 2022. Article (CrossRef Link)
[18] Nagy Attila, and Simon Vilmos, "Improving Traffic Prediction Using Congestion Propagation Patterns in Smart Cities," *Advanced Engineering Informatics*, vol. 50, no. 1, p. 101343, 2021. Article (CrossRef Link)
[19] Khac-Hoai Nam Bui, Jiho Cho, and Hongsuk Yi, "Spatial-temporal Graph Neural Network for Traffic Forecasting: An Overview and Open Research Issues," *Applied Intelligence*, vol. 52, no. 5, pp. 2763-2774, 2022. Article (CrossRef Link)
[20] Agafonov, "Short-Term Traffic Data Forecasting: A Deep Learning Approach," *Optical Memory and Neural Networks*, vol. 30, no. 1, pp. 1-10, 2021. Article (CrossRef Link)
[21] Hoque Jawad Mahmud, Erhardt Gregory D, Schmitt David, Chen Mei, and Wachs Martin, "Estimating the Uncertainty of Traffic Forecasts from their Historical Accuracy," *Transportation Research Part A: Policy and Practice*, vol. 147, no. 1, pp. 339-349, 2021. Article (CrossRef Link)
[22] Abduljabbar Rusul L, Dia Hussein, Tsai PeiWei, and Liyanage Sohani, "Short-Term Traffic Forecasting: An LSTM Network for Spatial-Temporal Speed Prediction," *Future Transportation*, vol. 1, no. 1, pp. 21-37, 2021. Article (CrossRef Link)
[23] Pavlyuk Dmitry, "Spatiotemporal Cross-validation of Urban Traffic Forecasting Models," *Transportation Research Procedia*, vol. 52, no. 1, pp. 179-186, 2021. Article (CrossRef Link)
[24] Anjomani Ardeshir, "An Integrated Land-use/transportation Forecasting and Planning Model," *Journal of Transport and Land Use*, vol. 14, no. 1, pp. 65-86, 2021. Article (CrossRef Link)
[25] Lu Huakang, Huang Dongmin, Song Youyi, Jiang Dazhi, Zhou Teng, and Qin Jing, "ST-TrafficNet: A Spatial-Temporal Deep Learning Network for Traffic Forecasting," *Electronics*, vol. 9, no. 9, pp. 1474-1474, 2020. Article (CrossRef Link)
[26] Jingjuan Wang, and Qingkui Chen, "A Traffic Prediction Model Based on Multiple Factors," *The Journal of Supercomputing*, vol. 77, no. 3, pp. 2928-2960, 2021. Article (CrossRef Link)
[27] Azzedine Boukerche, and Jiahao Wang, "Machine Learning-based traffic prediction models for Intelligent Transportation Systems," *Computer Networks*, vol. 181, no. 9, pp. 170530, 2020. Article (CrossRef Link)

[28] Weerasekera Rivindu, Sridharan Mohan, and Ranjitkar Prakash, "Implications of Spatiotemporal Data Aggregation on Short-Term Traffic Prediction Using Machine Learning Algorithms," *Journal of Advanced Transportation*, vol. 2020, no. 1, pp. 7057519, 2020. Article (CrossRef Link)
[29] Nadarajan Parthasarathy, Botsch Michael, and Sardina Sebastian, "Machine Learning Architectures for the Estimation of Predicted Occupancy Grids in Road Traffic," *Journal of Advances in Information Technology*, vol. 9, no. 1, pp. 1-9, 2018. Article (CrossRef Link)

Shuai zhang is a Ph.D student in Yanshan University, majoring in computer science and technology. His research interests mainly include data mining and urban computing.

Caiyan Pei is currently a lecturer at Hebei Normal University of Science and Technology, focusing on computer science education.

Wenyuan Liu is a professor and doctoral supervisor at the School of Information Science and Engineering, Yanshan University. His research interests include wireless sensor networks, mobile computing, distributed computing and data mining.