ATLANTIS PRESS

Advances in Intelligent Systems Research, volume 132

2nd International Conference on Modelling, Simulation and Applied Mathematics (MSAM 2017)

Short-term Traffic Flow Prediction Method Based on Balanced Binary Tree and K-Nearest Neighbor Nonparametric Regression

Dongfang Fan[1,2] and Xiaoli Zhang[2]

[1]Transportation Management College, Dalian Maritime University, Linghai Road, Dalian, Liaoning 116026, China

[2]China Academy of Transportation Sciences (CATS), No.240 Huixinlin, Chaoyang District Beijing 100029, China

***Abstract*—Real-time and accurate short-term traffic flow prediction is a key issue and difficult in traffic control and guidance. Using data mining and large data-driven principle, nonparametric regression is a better method to resolve short-term traffic flow prediction. But there are two main obstacles that case base is difficult to be generated and search is slow. For this reason, this paper presents a short-term traffic flow prediction method based on balanced binary tree and K-NEAREST NEIGHBOR NONPARAMETRIC REGRESSION (KNN2NPR). Case base is generated through clustering method and balance binary tree structure. K-nearest neighbor nonparametric regression improves accuracy of prediction and fulfills the real-time requirement. The prediction example in this paper demonstrates that this method is effective.**

Keywords-short-term traffic flow prediction; nonparametric regression; clustering; balanced binary tree

I. INTRODUCTION

Real-time and accurate short-term traffic flow prediction is essential in traffic control and guidance. Prediction results of different requirements and different intervals provide traffic managers and travelers with macro and relatively microscopic road traffic flow conditions. However, the complexity of road network and the random behavior of vehicles in intersections, vehicular entrances, ramps and others are very uncertain for short-term traffic flow prediction. How to take these random factors into full consideration and make a relatively accurate prediction is a key and difficult issue [1].

$$Y = f(X) + U$$

Where U is a random error term reflecting the combined effect of other factors affecting Y. In order to explain this method in brief, firstly, take reference of the working principle in the case of a single intersection. First select the X vector. Usually X refers to the relevant flow of the studied sections. As shown in Figure 1, sections a_1, a_2 and a_3 are network segment, and related to the studied section b.

As previously described, the uncertainties of traffic flow conditions of a_1, a_2 and a_3 will cause great difficulty to find regression law f[2] and great difference in the estimation in different periods f. Even in the same period, there will be great uncertain for the variables which impact Y, which means the importance of a_1, a_2 and a_3 in the impact of Y's flow status also changes at any time.

Because looking for the mapping relationship f between X and y will be very complex and not unique, uncertainty of f will be well resolved if there is another method which can fully describe the correspondence between X and Y and can execute dynamic adjustment under the premise of avoiding looking for f. Nonparametric regression is a very good solution. In nonparametric estimation, the form of function $f(X)$ is not assumed or fixed, and the parameter is not set. The value of the function relation at each point is determined by the data. In other words, the completion of mapping relationship f isn't relying on complex analytic expression of the function, but through the characteristics of the data itself and rely on large data-driving and mining. Specifically, the problem to be solved by nonparametric regression estimation is to find the nonparametric approximation function f' of function f according to the training input and output samples (X_i, Y_i). It is a model-free forecasting method based on the "pattern recognition". This method attempts to identify a set of past cases with the same input and status as the system at the time of prediction, or to find a "cluster" of historical data that is similar to the current input and then determine the predicted value based on the output of this "cluster" of historical data. The key of this method is how to determine the "cluster" historical data which is most similar to the system status at the time of prediction. In general, the nearest neighbor is composed of K points closest to the point of input state [3].

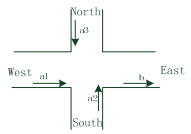

FIGURE I. THE INTERRELATED LINKS IN SINGLE INTERSECTION

The advantage of this method is that it does not need a priori knowledge and a large number of parameter identification, only a sufficient number of historical databases is needed. Therefore, the nonparametric regression model has a strong ability to deal with emergencies, and the prediction accuracy and error distribution are good. The principle is clear and robust, and it is suitable for nonlinear and uncertain system prediction[4]. Smith [5] applied it to single-point short-term traffic flow prediction, but it was not used in practice because of its slow search speed and parameter adjustment method by trial and error. Then the scholars put forward a lot of improvements. Gong Xiaoyan et al.[6] proposed that forecast

accuracy and speed would be improved by using the density to generate the sample database and optimizing the structure based on the historical data of the hash table. But it cannot fulfill the requirement of real-time prediction when the capacity of data base is very huge. At present, the use of non-parametric regression method to predict the short-term traffic flow has not entered into the stage of practical application yet, following obstacles exist [7].

1) The generation and update of case database. Even in the case of a single intersection, the traffic situation is very different. There are a lot of impact factors, such us traffic signal release period, uncertainty of traffic absorption point, weather condition and unexpected events, which may affect the traffic flow. Therefore, the size of the case database which store traffic pattern may be very large. Traffic patterns will change over time and the size of the database will increase accordingly. For example, the Hampton Roads Smart Traffic Center (HRSTC) builds 203 points and 634 detectors for an approximately 30 km freeway in Virginia. The data size reaches 17 GB and increases with a rate of 650 MB per month[8]. Nonparametric regression, in essence, is still an intelligent method based on pattern recognition. The predictive performance depends largely on the completeness and typicality of various traffic flow patterns in the case database. Therefore, the difficulty of this method is the choice of neighbor (including the determination of the number of neighbor K) and the size of the database.

2) Search speed. Case database is a large collection of historical characteristics of data, has large amount of data. Online forecasting requires accurate prediction and real-time requirements. At present, most the case databases use a simple linear structure. The search method is slow, cannot meet the real-time requirements. However, in order to improve the efficient of search, it is necessary to establish a complex structure which has high maintenance costs. It is an inevitable contradiction.

In the view of existing problem, this paper proposes a nonparametric regression method for traffic flow predication based on balanced binary tree. Find out the clustering center in the original data, keep clustering center point and K neighboring points of the clustering as the representative points of the cluster centers, and remove other dependent points of the cluster centers. The compacted database retains the necessary feature data, and eliminates unnecessary redundant data. The structure of the case database uses a balanced binary search tree structure to achieve search with the minimum time in a large number of case databases, to ensure the time complexity of search is $O(\log_2 n)$, which greatly improves the search speed.

II. ALGORITHM FRAMEWORK

Figure 2 is the traffic flow prediction algorithm framework, based on *K*-neighborhood nonparametric regression. The case database uses balanced binary tree. The flow is that, first, the original data of the traffic flow detected by the detector is clustered to find out the clustering center of the original data and add it to the case database. Then search the state vector *X(t)* of the system at time t in case database. If the matching data is searched, forecast according to the K historical data a in the cases database and current stat data; if the matching data is not searched, add current vector *X(t)* into the database as a new situation which hasn't been considered, modify the case databases of the binary tree to maintain the binary tree balance structure.

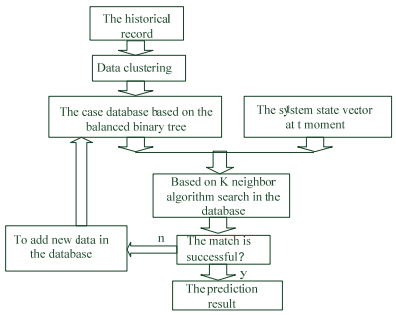

FIGURE II. THE FRAME OF FORECASTING ALGORITHM BASED ON K-NEAREST NEIGHBORS NON-PARAMETRIC REGRESSION

A. Clustering Analysis of Historical Data

Suppose the data saved in the database is S historical traffic of *M* road segments which is closest to the traffic *v(t)* of the studied road segment.

$$v_m(t-1), v_m(t-2), \ldots, v_m(t-s)$$

In this paper, clustering algorithm is used to find clustering centers, to eliminate purpose of redundant data. There are a variety of clustering algorithms to choose from, such as DBSCAN (density-based spatial clustering), SC (subtractive clustering), and CURD (based on reference points and density clustering). Each method has its own advantages and disadvantages. DBSCAN requires that the number of data objects within a given radius eps for each core object in a cluster must be greater than a given value (expressed in minpts), with the advantage of being able to discover clusters of arbitrary shape. However, in this paper, the precondition minpts cannot be determined in advance, but can only be determined by trial and error. SC assumes that each data point is a potential cluster center, and then calculates the measure of each possible cluster center according to the density of nearby points. The radius parameter, radii, is a vector of 0 to 1, which is used to specify the extent of influence of the cluster center on each data dimension. The smaller the radii, the more clustering can be found; CURD adopts a certain number of reference points to represent a clustering region and shape effectively, shields abnormal data (noise) on the impact of the algorithm by density method. Various clustering methods are compared in the prediction example. *K* points nearest from the cluster center are recognized as nearest neighbor points. The distance between the each data point and the center point is calculated by Euclidean distance formula. Formula (2),

$$L_t = \sqrt{\sum_{m=1}^{M}\sum_{s=1}^{s}[v_m^c(t-s) - v_m(t-s)]^2}$$

B. Establish a Database

Establish a database based on binary balance tree balanced binary tree is AVL tree which is a special binary search tree. In this paper, the binary balance tree is mainly used to achieve

dynamic balance, and can keep the minimum height of the tree. Properties are as follows:

1) If the left subtree is not empty, the key value of all the nodes in the left subtree is less than the key value of the root node;

2) If the right subtree is not empty, the key value of all the nodes in the right subtree is greater than the key value of the root node;

3) The absolute value of the difference between the height of the left and right subtrees of the root is not more than 1;

4) The left and right subtrees of the root are binary balance trees.

The dynamic balance of the tree is mainly due to that the dynamic adjustment of the tree will be carried out every time when add nodes or delete nodes. That is, the tree is always in balance. Therefore, the balance of the binary tree depends on the balance factor of the node, which is defined as the height of the left subtree of the node minus the height of the right subtree. So the balance factor of all the nodes in the binary balance tree can only be -1, 0 and 1. As long as there is a node in the binary tree balance factor which absolute value greater than 1, the tree is not a binary balance tree, must be adjusted.

The height h of the AVL tree with n nodes satisfies:

$$log_2(n+1) \leqslant h \leqslant 1 \quad 4404 log_2(n+2)-0.328$$

The search time of the binary balance tree depends on the height of the tree, so the computational complexity of worst case of searching on a binary-balanced tree is $O(log_2 n)$.

The process of establishing a database is the process of establishing a binary balance tree, which is a process of inserting nodes one by one and adjusting the tree structure to make it re-balanced and sorted. Adjusting the tree structure not only adjusts the relationship between nodes (balance rotation), but also adjust the node balance factors. According to the smallest subtree out of balance to determine how to do balance rotation operation, which has LL type, LR type, RL type and RR type four types. The binary tree structure must be adjusted when insert a new node into the database. Thus, establishing a database is a very time-consuming operation when build a database with a large amount of data. It requires operate offline. When all the nodes are inserted, the database operation is completed.

C. Threaded Binary Tree

If the search dimension is $M \times S$, when searching for the balanced binary tree, it is the sum of the original data, so the search dimension is 1, so that the difference between the different dimensions of the data is erased. Therefore, the balanced binary search tree can only be used as a rough search. In order to locate the best matching node accurately, it is necessary to carry out fine search. The fine search limits the search range to the successful node (error less than ε) and the predecessor and successor nodes of the node (the search range can be extended to improve accuracy). Adding the predecessor and successor node pointers in the node can make the search more efficient. The improved binary tree structure is shown in Table 1.

TABLE I. THE ADVANCED STRUCTURE OF DATABASE NODE

Index value	$\{v_m^z(t-s)\}$	$\{v_m(t-s)\}$	Balance Factor	Predecessor node pointer	Success or node pointer

D. Search Current Traffic Flow Data in the Database

Since the balanced binary tree itself is a search tree, the node values are sorted in the middle sequence in the tree. Suppose matching value x', allowable error ε, search the matching node p_v in the balanced binary tree. Algorithm is as follows.

1) Starting from the root node pg, $p \leftarrow pg$;

2) If p is empty node, return failure. Calculate the difference between the index value D of p and x', $\Delta = | x' - p_D |$. If $\Delta > \varepsilon$, the following three cases are discussed.

(A) If p is a leaf node, return failure;

(B) If $x' > p_D$, search in the node p_R of the right subtree of p, that is, $p = p_R$, and return to 2) iteratively;

(C) If $x' < p_D$, search in the node p_L of the left subtree of p, that is, $p = p_L$ and return to 2) iteratively. If $\Delta < \varepsilon$, search predecessor p and the succeeding node p_h;

3) Calculate the Euclidean distance for p, p_q and ph respectively with x', whichever is the smallest as the matching node p_v.

4) Return to p_v.

E. Prediction

If the current data in the case database if the match is successful, matching node p_v, means that the current situation in the previous similar situation has occurred. Forecast based on the average of traffic in time t of K historical data p_{vk} around p_v in the database. The prediction formula is,

$$v(t+1)^r = \frac{1}{k}\sum_{k=1}^{K} p_{vk}(t)$$

And replace the longest retention one in K historical data matching node with current data. The reason for doing so is considering to update the data in database over time. .

III. PREDICTION EXAMPLE

The traffic flow data adopts traffic flow simulation software[9]. Assuming section b is the flow of the studied section, sections a_1, a_2 and a_3 are related sections, ie M=3, as shown in Fig1.

The road length is 800 m, split is 0.5. Historical flow $v(t-1)$, $v(t-2)$ and $v(t-3)$ of related segment are the most closely matched to traffic density of segment b. I.E. S=3. Sampling interval is 2 min. a total of 6,000 sets of data are achieved. The first 5,000 sets of data are adopted to generate the database. Left 1,000 sets of data are for testing. The data of the first 5,000 groups are clustered by different clustering algorithms to get different cluster centers, as shown in Table 3. As shown in table 2, using DBSCAN is difficult to get suitable cluster center through trial-and-error, because there are two parameters of the clustering radius eps and min pts; CURD has three parameters

which are radius distance radius, density threshold t and cycle number iterate. The values of radius and t are consistent with the values of eps and minpts in DBSCAN, iterate is a fixed value of 4. The results show that the clustering center number c and DBSCAN are almost the same. The variation trend of the number of cluster centers with the decrease of cluster radius is shown clearly by SC method, and the variation of radii is 0.13 ~ 0.14. In this paper, the clustering radius radii = 0.134 and the cluster center number c = 2,615 are used to predict. Table 4 shows the prediction errors for different K values. As shown in Table 3, when K = 5, prediction is the best. Table 4 shows the average number of searches, time spent, and forecast error for different database structures. The linear structure is the original data after cluster analysis, directly into the database for matching prediction.

TABLE II. THE CLUSTER RADIUS AND CENTER NUMBER WITH DIFFERENT CLUSTERING METHODS

DBSCAN	Cluster radius eps	0.150	0.140	0.135	0.135	0.135	0.132	0.130
	Neighborhood densitymi npts	12	12	10	5	2	2	2
	Number of cluster centersc	309	635	732	890	2058	2526	2875
SC	Cluster radius radii	0.150	0.140	0.136	0.135	0.134	0.130	0.120
	Number of cluster centersc	328	742	2227	2419	2615	3179	4000
CURD	radius distance radius	0.150	0.140	0.135	0.135	0.135	0.132	0.130
	Density threshold *t*	12	12	10	5	2	2	2
	Cyclesiterate	4	4	4	4	4	4	4
	Number of cluster centersc	288	615	728	879	2140	2574	2916

TABLE III. THE FORECASTING ERRORS WITH DIFFERENT K VALUES

K=2		*K*=3		*K*=4	
Absolute error (Vehicles)	Absolute error (Vehicles)	Absolute error (Vehicles)	Absolute error (Vehicles)	Absolute error (Vehicles)	Absolute error (Vehicles)
5.15	15.17	5.03	14.24	4.78	12.91
K=5		*K*=6			
Absolute error (Vehicles)	Absolute error (Vehicles)	Absolute error (Vehicles)	Absolute error (Vehicles)		
4.33	11.52	4.98	12.63		

As shown in Table 4, the structure of the database has a great impact on the search speed. The linear structure which is commonly used is the worst, followed by binary sort tree, and balanced binary tree is the best one. The improvement of the search speed leads directly to the increase of the prediction speed, so that the real-time prediction can be realized. In the prediction error, the linear is optimal, binary sort tree and balanced binary tree followed, but almost the same.

TABLE IV. THE AVERAGE NUMBER OF SEARCH, NEEDED TIME AND FORECASTING ERRORS WITH DIFFERENT DATABASES

Database structure	Linear	Binary sort tree	balanced binary tree
Average search times	1416	132	15
Average search time(s)	10	2	1
Absolute error (Vehicles)	412	435	433
Relative error(%)	1037	1143	1152

REFERENCES

[1] H. Y. Xiang, H. L. Shu. An Ensemble Model of Short-Term Traffic Flow Forecasting on Freeway. Applied Mechanics and Materials, Vols.744-746, pp.1852-1857, 2015

[2] Xinyu Min, JianmingHu,Member, IEEE and Zuo Zhang, Member, IEEE. Urban Tranffic Network Modeling and Short-term Tranffic Flow Forecasting Based on GSTARIMA Model. 2010 13th International IEEE Annual Conference on Intelligent Transportation System Madeira Island, Portugal, September 19~22,2010

[3] Smith B L, Demetsky M J. Short term traffic flow prediction models—A comparison of neural network and nonparametric regression app roaches [A]. In: Proceedings of IEEE International Conference on Systems [C]. San Antonio, TX, USA:IEEE, 1994. 1706—1709.

[4] He G G. Introduction to ITS System Engineering[M]. Beijing: China Railway Press, 2004.

[5] Smith B L, Demetsky M J. Traffic flow forecasting: Comparison of modeling app roaches[J]. Journal of Transportation Engineering,1997,123(4):261—266.

[6] Gong X Y, Tang SM. Integrated traffic flow forecasting and traffic incident detection algorithm based on non-parametric regression[J]. China Journal of Highway and Transport,2003,16(1):82—86.

[7] Wu, S., Yang, Z.,Zhu, X., and Yu, B.(2014). "Improved k-nn for Short-Term Traffic Forecasting Using Temporal and Spatial Information." J. Transp. Eng., 10.1061(ASCE)TE.1943-5436.0000672, 04014026.

[8] Ma S, Wang T J, Tang SW, et al. A fast clustering algorithm based on reference and density[J]. Journal of Software 2003,14 (6):1089—1095.

[9] Zhang C. Urban Transport Micro-Simulation System Design and Implementation[D]. Tianjin: Tianjin University, 1997.