International Journal of scientific research and management (IJSRM)
||Volume||2||Issue||12||Pages||1856-1861||2014||
Website: www.ijsrm.in ISSN (e): 2321-3418

"Travel Time Prediction Methods for Dynamic Incident Handling"

Sanjay Singh**, **Dr. Alka Singh
Research Scholar
Shri Venkateshwara University
Research Supervisor
Asst. Professor,Shri Ram Swaroop University , Lucknow.

Abstract:-

Effective travel time prediction is of great importance for efficient real-time management of freight deliveries, especially in urban networks. This is due to the need for dynamic handling of unexpected events, which is an important factor for successful completion of a delivery schedule in a predefined time period. This chapter discusses the prediction results generated by two travel time estimation

Methods that use historical and real-time data respectively. These methods are incorporated by the real-time fleet management system for the prediction of arrival time to the remaining customers. The first method follows the k-nn model, which relies on the non-parametric regression method that was analyzed, whereas the second one relies on an interpolation scheme which is employed during the transmission of real-time data in fixed intervals. The study focuses on exploring the interaction of factors that affect prediction accuracy by modeling both prediction methods. The data employed are provided by real-life scenarios of a freight carrier and the experiments follow a 2-level full factorial design approach.

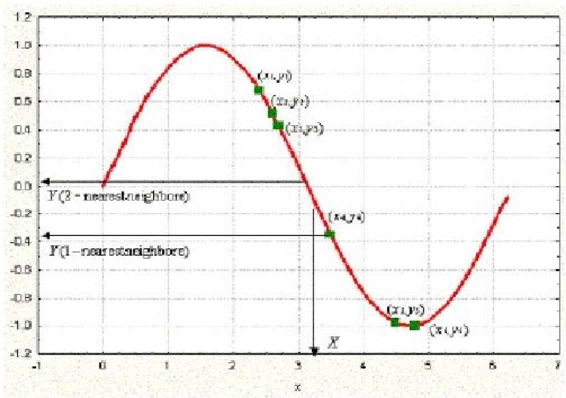
Background theory on non-parametric regression

The k nearest neighbor (k-nn) model-

The proposed method that uses historical data for travel time prediction adopts a non parametric regression technique described as a k nearest neighbor (k-nn) model (Clark, 2003). It has been shown that this technique provides accurate predictions when a large amount of historical data is available (Yun et al., 1998; Vlahogiani et al., 2003; Clark, 2003). This model is best described as a pattern matching exercise, where recent observations are matched with those contained in a database of historical observations. From all the matches, either the k nearest matches or all the matches below a given distance threshold are located. The successive observations from these ''best'' matches are then averaged, usually by taking the arithmetic mean, to obtain the forecasts. The only parameters in the model are the number of observations to match with and the number of best matches to retain, or the distance threshold. Once a sequence of recent observations has been matched and forecasts made, the recent observations can then be added to the historical matching database for use in subsequent matching operations.

As mentioned previously the k-nearest neighbor model can be implemented in non- parametric regression problems. The latter are concerned with predicting the outcome of a dependent variable given a set of independent variables. To start with, we consider the figure shown below, where a set of points (green squares) are drawn from the relationship between the independent variable x and the dependent variable y (red curve). Given the set of green objects (known as examples) we use the k-nearest neighbor method to predict the outcome of X (also known as query point) given the example set (green squares).

Figure- The use of the k-nn model in non parametric regression problems



To begin with, let's consider the 1-nearest neighbor method as an example. In this case we search the example set (green squares), and locate the one closest to the query point X. For this particular case, this happens to be x4. The outcome of x4 (i.e., y4) is thus then taken to be the answer for the outcome of X (i.e., Y). Thus for 1- nearest neighbor we can write:

Y = y4

For the next step, let's consider the 2-nearest neighbor method. In this case, we locate the first two closest points to X, which happen to be y3 and y4. By taking the average of their outcome, we can define the solution for Y.

The above discussion can be extended to an arbitrary number of nearest neighbor k. To summarize, in a k-nearest neighbor method, the outcome Y of the query point X is taken to be the average of the outcomes of its k nearest neighbors. Thus, given a query point, k-nn makes predictions based on the outcome of the k neighbors closest to that point. Therefore, to make predictions with k-nn, we need to define a metric for measuring the distance between the query point and cases from the examples sample. One of the most popular choices to measure this distance is known as Euclidean method.

$$\mathbf{D\ (\ x,\ p\ ) = \sqrt{\ (\ x - p\ )2}}$$

Where x and p are the query point and a case from the examples sample, respectively. After selecting the value of k, we can make predictions based on the k-nn examples.

<u>**Implementation challenges**</u>

There are four fundamental challenges when applying non-parametric regression (Smith et al., 2002): a) definition of an appropriate state space, b) definition of a distance metric to determine nearness of historical observation to the current conditions, c) selection of a forecast generation method given a collection of nearest neighbors, and d) management of the database.

State space

In the case of a time series, a state vector comprises a set of consecutive measurements at time t, t-1,…, t-d where d is an appropriate number of lags. For example, a state vector x(t) of measurements collected in a consecutive time manner with d = 3 can be written as:

x(t) = [V (t), V (t – 1), V (t – 2), V (t – 3)]

Where V(t) is the rate during the current time interval, V(t-1) is the rate during the previous interval, and so on. Note that an infinite number of possible state vectors exist. Furthermore, they are not restricted to incorporating only lagged values but may also be supplemented with aggregate measures such as historical averages.

Distance Metric

A common approach to measuring ‘‘nearness’’ in non-parametric regression is to use Euclidean distance. Such an approach, weights the value of each measurement equally, which means that the information content of each measurement is considered to be of equal worthiness. In many cases, knowledge of the problem domain may make such an assumption unreasonable. In that case, a weighed distance metric, in which the ‘‘dimension’’ of variables with higher information content would be weighed more heavily may be appropriate. While this makes intuitive sense, it is clear that such an approach is heuristic in nature, and requires careful consideration by the modeler.

Forecast generation

The most straightforward approach to generating the travel time forecast is to compute a simple average of measurement values of the neighbors that have fallen within the non-parametric regression neighborhood. The weakness of such an approach is that it ignores all information provided by the distance metric. In other words, it is logical to assume that past cases ‘‘closer’’ to the current case have higher information content and should play a larger role in generating the forecast.

To address this concern, a number of weighting schemes have been proposed for use within non-parametric regression. In general, these weights are proportional to the distance between the neighbor and the current condition. An alternative to the use of weights is to fit a linear or nonlinear model to the cases in the neighborhood, and then use the model to forecast the value of the dependent variable.

Database issues

As stated in Chapter 2, the effectiveness of non-parametric regression is directly dependent on the quality of the database of potential neighbors. Clearly, it is desirable to possess a large database of cases that span the likely conditions that a system is expected to face. However, while as large a database as possible is desirable for increasing the accuracy of the model, the size of the database has

Significant implications on the model execution time.

When considering how non-parametric models work, one will see that the majority of effort at runtime is spent in searching the database for neighbors. As the database grows, this search process grows accordingly. For real-time applications, such as travel time prediction, this is a significant issue (Smith et al., 2002). Steps must be taken to keep the database size manageable, while ensuring that the database has the depth and breadth necessary to support forecasting.

Again, there are various approaches to accomplish this. One approach would be to cluster the database, and only search those cases in the cluster in which the current state falls. Another approach would be to periodically delete records from the database. This process would involve searching for multiple records that are nearly identical. Such an approach would require the use of a distance metric such as those discussed earlier. While examining this issue fully is beyond the scope of this thesis, it is important to realize that the management of the database is a critical issue, particularly in real-time applications of non-parametric regression.

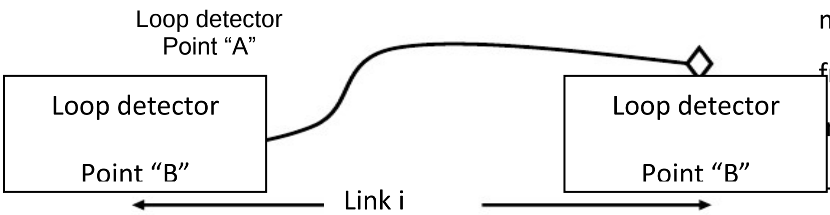
<u>Travel time prediction methods for dynamic incident handling</u>

Fundamentals of travel time prediction

In order to calculate the travel time between two points of interest, the road is usually segmented into sections (or links) with consecutive loop detectors. The later obtain values such as velocity, flow and occupancy that are reported at a certain period of time (usually every few seconds). Figure 3.2, shows a section with two loop detectors being in the extreme points of the link (denoted as points "A" and "B").

$$T = \sum t(i)$$

Figure -Travel time predictions in a certain link



Therefore, the travel time along a link i, t(i) can be written as follows:

$$t(i) = 2li/v A + Vb$$

Where li denotes link length, v A and v B are the measured velocities at the extremities (point A and B) of link i. The total travel time (T) along the road f

Link i

links is then calculated by summing the estimates for each segment:

The above equations are the fundamental elements of most travel time estimation and prediction models. Below we present two novel methods for travel time estimation. The first method uses historical data (i.e. historical mean travel times between customers) that were collected from previous visits of the vehicle to the specific customers whereas the second uses real-time data (i.e. actual mean velocity and vehicle's position during delivery execution) for estimating the arrival time in the remaining customers. The combination of customer delivery restrictions (i.e. time windows) with accurate knowledge of vehicle's arrival to customer sites allows for effective detection of time deviations from the initial delivery plan.

Travel time prediction using historical data

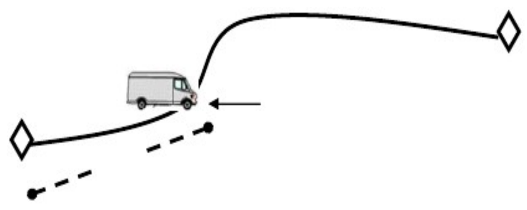
This travel time estimation method uses historical data in order to predict the arrival time of a vehicle to a customer. This historical data is retrieved from a database that stores information concerning the time that took to a vehicle to travel from one point to another and provides mean travel times from any point of interest to any other.

$$t_{CB} = \frac{D_{AB} - D_{AC}}{D_{AB}} \times \overline{T}_{AB}$$

Let us assume a route between two points of interest A and B and a vehicle travelling towards point B (Figure 3.3). The distance between these points is noted as DAB. The vehicle has already travelled a certain distance, say DAC. When the vehicle is travelling towards point B, the travelling time (tCB) from point C to point B is estimated by:

Where DAB and DAC are the total and travelled distance respectively and TAB is the mean historical travel time between these two customers which is revealed from the database that stores data for each delivery schedule.

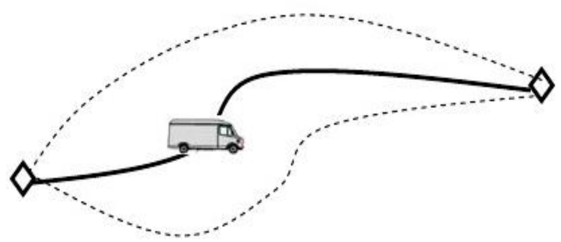
Figure -Travel time prediction using historical data



As it can be seen, the predictions made by this method depend on how accurate the calculation of TAB and on the homogeneity of the historical data.

As mentioned above, TAB is derived from previous travel times between points A and B that the vehicle has carried out. However, the main problem is that when historical data concerning travel times between these two points are extracted, the latter include not only one arc which is direct from point A to point B but a sequence of arcs where other points are inserted between points A and B.

Figure - Three different links of getting from Point A to Point B



The latter causes inaccuracies to travel time prediction as the mean travel time TAB is increased. In order to determine which historical travel times should be considered for deriving the mean time TAB, we apply an outlier removal method (1st stage) where a selection of historical data is made based on time and the k-nn algorithm (2^{nd} stage) where a refinement a data is made based on distance. Here is a step by step demonstration on how to identify from the database the routes with certain travel times:

1) Determine from the database the link that the vehicle has followed in order to travel from point A to point B with minimum travel time.

2) Identify whether the vehicle has used the same link previously more than once.

a) If there is no such case, use only the specific data (i.e. historical travel times retrieved from this single trip) in order to estimate the arrival in the following customer

b) If other delivery schedules are identified where the vehicle has used more than once the specific link, then:

i) Identify these travel times

ii) Remove outliers but choosing the ones that are equal or less than a certain percentage of the minimum travel time (≤25%)1

3) Determine parameter K (i.e. number of nearest neighbors)

4) Calculate the distance between the query-instance and all the training samples

5) Sort the distance and determine nearest neighbors' based on the K-th minimum distance

6) Use their arithmetic mean in order to calculate TAB

It must be mentioned though that we have assumed that the minimum historical travel time represents a direct route between the two customers (i.e. not including any intermediate stops). The probability of this being the case is dependent on the availability of historical data between the two customers in our database. In other words, the more historical data we have the more likely it is that the minimum travel

Time contained in this data will represent a direct link between any two customers. In our tests, we have chosen customer pairs that represent consecutive customers in the route and we have visually inspected the resulting route in the map to verify that no intermediate stops are included. Further to the

aforementioned tests, we have chosen randomly, 100 cases from the database in order to investigate the percentage of cases where the minimum historical travel times represented a direct route between two customers. The results were very encouraging
as in 97 cases out of 100 a direct route was observed.

Applying the k-nn algorithm: An example

In order to fully understand the procedure explained above, we will demonstrate a test case. Let us assume

Abdulhai, B., Porwal, H. and Recker, W. (2003) “Short-term Freeway Traffic Flow Prediction Using Genetically-optimized Time-delay-based Neural Networks” UCB, UCB-ITS-PWP–99-1 (Berkeley, CA: Institute of Transportation Studies, University of California, Berkeley)

Aronson L. D., and Van der Krogt, R. P. J. (2003), Incident Management in Transport Planning, TRAIL Research School, Faculty of Information technology and Systems, Delft University of Technology, Nederland, White paper.

Attwood Nigel 1992 Logistics of a Distribution System. 4th edition. Billing and Sons Ltd. Worcester

that for point A to point B we have retrieved a link with travel time from point A to B to be 8 minutes (minimum travel time). We then retrieved 6 other cases where the truck has used the same link. Four of them had travel times which were less or equal to 25% of the minimum travel time.

REFERENCES

Campbell, A., Clarke, L., Kleywergt, A., Savelsbergh, M. (1998) “The inventory routing problem”, In Laporte G., Crainic, T.G. (Eds.) Fleet Management & Logistics,Kluwer, Boston, US

Chien, S. I. J. and Kuchipudi, C. M. (2005) “Dynamic travel time prediction with real- time and historical data”, in: Proceedings of the Transportation Research Board 81st Annual Meeting, Washington, DC.

Cooper James 1994 Logistics and distribution planning. 2nd edition. Great Britain by Biddles Ltd.

Fleischmann, B., Gietz, M., Grutzmann, S. (2006), “Time-varying Travel Times in Vehicle Routing”, Transportation Science 38 (2).