Weather-dependent road travel time forecasting using a neural network

G. Ghiani, D. Gullì, F. Mari, R. Simino & R. Trunfio
Centro di Supercalcolo per l'Ingegneria Computazionale, Nec Italia S.r.l., Italy

Abstract

The estimation and prediction of link travel times in a road traffic network are critical for many intelligent transportation system (ITS) applications, such as the route guidance system (RGS), advanced traveller information system (ATIS) and freeway traffic management system (FTMS). These systems are adopted to help individual drivers to identify optimal routes based on real-time information on current traffic conditions. The identification of the optimal routes is particularly important for trips where the travel time is relatively long and where it is unlikely that the current travel time will remain stable. In addition, weather conditions and road type are both parameters that influence the travel time of a specific link, and they need to be included in the forecasting process. The aim of our study is to propose a new system for travel time forecasting based on a multilayer feedforward neural network. Both historical and real-time data (which can be provided by loop detectors and sensors positioned along the roads) are inputs for the neural network that returns the short-term travel time needed to traverse the road section that it is related to. Data used to train and test the neural network have been generated using a simulator that is influenced by deterministic (*e.g.* road type) and stochastic (*e.g.* weather, visibility) parameters.
Keywords: travel time forecasting, neural network.

1 Introduction

The problem of travel time prediction has received over the years more and more attention from researchers and many methods have been proposed. The main idea of travel time forecasting is based on the fact that traffic behaviours possess both partially deterministic and partially chaotic properties. Good forecasting

WIT Transactions on The Built Environment, Vol 101, © 2008 WIT Press
www.witpress.com, ISSN 1743-3509 (on-line)
doi:10.2495/UT080491

results can be obtained by reconstructing the deterministic traffic motion and predicting the random behaviours caused by unanticipated factors. Numerous studies have focused on how to obtain accurate travel time predictions. The problem of knowing in advance link travel times in a road traffic network is crucial and critical for many *intelligent transportation system* (ITS) applications such as the *route guidance system* (RGS), *advanced traveller information system* (ATIS) and *freeway traffic management system* (FTMS). The common objective of these systems is to provide information necessary to help individual drivers to identify optimal routes based on real-time information on current traffic conditions. To identify these optimal routes, the selection algorithms should base their link travel times on when the driver actually arrives at a given link rather than the current link travel time. This would be particularly important for trips where the travel time is relatively long (*e.g.* greater than 30 minutes) and where it is unlikely that the current travel time will remain stable. In reality, most drivers base their routes on the estimated travel time when they arrive at a particular link; the ATIS projects should also have the same capabilities. The current travel time information can be obtained from a number of sources including loop detectors, probe vehicles, AVI systems and GPS systems etc.

Our aim is to propose a new travel time forecasting system which accepts as inputs both historical data and current information collected by appropriate sensors and weather stations positioned along the roads. The system is based on a *multi-layer feedforward neural network*.

The paper is organized as follows. Next section is dedicated to a short literature review of the problem we have addressed. After this, we detail our approach and introduce different possible configuration of our system. In the section devoted to the performance analysis we demonstrate how the effects of collecting and using the weather and visibility conditions data as inputs for our forecasting system might sensibly influence its performances.

2 Literature review: previous traffic prediction efforts

With the advent of route guidance systems (RGS), the prediction of short-term link travel times has become increasingly important. In order for RGS to be successful, the calculated routes should be based on anticipatory link travel time information, forecasted using historical information along with current traffic information. Most of the link travel time research has focused on link travel time estimation rather than on link travel time forecasting, taking the input data from loop detectors, probe vehicles, and simulation programs.

The approaches used for traffic prediction are largely dictated by the fact that traffic conditions are time dependent and often follow fairly well-defined patterns. Previous traffic prediction efforts can be classified as i) historical, database algorithms; ii) simulation models; iii) time-series and Kalman filtering models, and iv) neural network models.

All the approaches present interesting results, brilliant idea and unfortunately some weakness due to their assumption.

Historical, data-based algorithms are based on the assumption that traffic patterns are cyclical. A major weakness of this methodology is that it implicitly assumes that the projection ration will remain constant (Kaysi *et al.* [1]).

Simulation models provide predictive capability because they demonstrate how the system is likely to react to varying conditions and control strategies. "An effective on-line simulation model would enable the advanced transportation management system (ATMS) control centre to project promptly future traffic patterns considering any previously implemented strategies in a real-time operating environment" (Junchaya *et al.* [2]). Unfortunately, at this time, the real-time application of traffic simulation is not feasible because existing mode/algorithm constructs cannot support real time application.

The basic idea of the *time-series analysis techniques* is to forecast the condition $X(t+d)$, given $X(t)$, $X(t-d)$, $X(t+2d)$, and so on while d is the prediction time interval. The Box and Jenkins technique is a widely used approach to specifying a variety of time-series models. It has been shown to yield accurate forecasting results in a number of application areas. These methodologies are usually combined using Kalman filtering models. *Kalman filtering* is a useful technique based on the idea to use information about how measurements of a particular aspect of a system are correlated to the actual state of the system.

However, these studies are limited with respect to RGS in that they are concerned only with one-time-period-ahead prediction. This assessment is supported by comparison study of filtering model with neural network model made by Park *et al.* [3].

2.1 Neural networks

Over the past several years, both in research and in practical applications, neural networks have proven to be a very powerful method of mathematical modelling. In particular, neural networks are well suited for pattern recognition, offer efficient execution, and model nonlinear relationships effectively. Clearly, neural networks are well worth exploring as a tool for the short-term prediction of traffic. A *neural network* (NN) may be defined as "an information processing technology inspired by studies of the brain and nervous system" (Klimasauskas [4]). This inspiration obviously led to the use of the word neural. However, NNs in no way attempt to produce biological clones; rather, they are simply models with a rigorous mathematical basis. Although NNs are typically associated with the field of artificial intelligence, they function as a sophisticated form of regression.

The use of NNs has been proven successful in a number of applications for the following reasons: i) NNs can perform highly nonlinear mappings between input and output spaces; ii) the parallel structure of NNs lends them to implementation on parallel computers, which offers the potential for extremely fast processing; and iii) the NN approach is nonparametric. Therefore, one need not make any assumptions about the functional from of the underlying distribution of the data.

These characteristics have attracted the attention of transportation researchers. In the past decade, a large number of research projects have been done with NN technology in the area of incident detection, incident delay, signal control system, freeway travel time forecasting and short-term traffic flow prediction. The most notable works could be referred to Smith and Demetsky [5], Florio and Mussone [6], Park and Rilett [7] and Innamaa [8].

Sharda and Patil [9] concluded from their work that the simple NNs could forecast about as well as the Box-Jenkins forecasting technique. Tang *et al.* [10] in their comparative study of the performance of NNs and conventional statistical techniques concluded that for short-term memory series, NNs appear to be superior to the Box-Jenkins model.

All these results showed that such NN prediction models hold considerable potential when used in real time ITS applications.

3 The proposed system

We have developed a new system for travel time forecasting based on a multilayer feed forward NN. The output of the NN is the prediction of the short-term travel time needed to traverse a link *(i,j)* of the graph *G* representing the application field. The NN is trained periodically using historical data, while it will predict the short-term travel time using real-time data (both data are provided by detectors positioned along the way). Collected data concern to *deterministic* (*e.g.* road type) and *stochastic* (*e.g.* gap between vehicles) parameters.

3.1 The input data of the neural network

The innovation in our technique lies not in the fact that we use a NN for the travel time forecasting, as we have shown many NN based forecasting system are available in literature, but in the input data that we give to the NN.

Transportation engineers have developed during the past years a number of functions to calculate travel times basing on different parameters. Those functions have been proved to calculate travel times well only in static environments. This is far from a real life situation in which travel times are slightly dependent on the traffic conditions that may vary many times and in many different manners in a single day. Otherwise, the idea to use streets conditions, data representing the weather conditions and traffic flow data to calculate the travel time prediction seemed to us still to be a good idea. Moreover, the progress in technology development during the last decade can support this idea, therefore we have decided to consider the following data as input for our forecasting system.

The input data for our system have been divided into three macro-categories: i) *weather conditions*, ii) *street conditions* and ii) *traffic conditions*. A total of 8 input neurons belonging from the three categories provide to the prediction system the current traffic data information of a particular arc *(i,j)* of the graph representing the observed territory.

To the weather conditions data category belong two neurons: one expressing the meteorological situations (*dry*, *rainy*, *snowy*, *foggy*) and one expressing the visibility conditions (*high*, *average* and *low* visibility). This data are usually collected using a weather station.

To the second category belong three different inputs: the first one expresses the number of available lanes, the second one is associated to the length (in kilometres) to cover and the last neuron is related to the street type (*level*, *rolling* and *mountainous*).

To the traffic conditions input category are associated three different neurons: *time mean speed* (kilometres per hour), *flow rate* (vehicles on time slot) and the *average gap* (in seconds). Every one of them are values measured over the data collected in the last 10-minutes for the link *(i,j)*. Using video and loop detectors is possible to collect for each vehicle *velocity* and *entry time*.

In the next figure are summarize the input data of the NN.

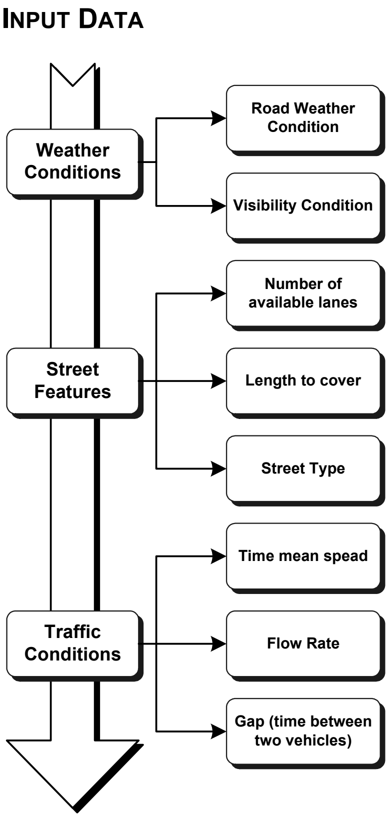

Figure 1: Input data of the neural network.

3.2 The data set

The territory in which the system has been tested is the area of a small town of South Italy (Cosenza, Calabria). In order to face the necessity to test our system with realistic input data, an ad hoc microsimulator has been developed to generate the data set.

In the past, Ritchie and Cheu [11] investigated the capability of multilayer feed-forward neural networks to recognize spatial and temporal traffic patterns using data simulated with INTRAS simulation model.

Our simulator combines mathematical relations (*e.g.* safety distance between vehicles, speed reduction in function of weather and visibility conditions) and stochastic events (*e.g.* weather, vehicles generation from a centroid).

In particular, the number of vehicles that move from one point to another one in a time interval (for us an hour) has been generated using a set of origin-destination matrix constructed in collaboration with the local administrations of the area of Cosenza (Calabria, Italy).

The entry speed of a vehicle for the link *(i,j)* is computed as function of the link speed limit, number of available lanes, street type, weather and traffic conditions (Godwin [12], Kyte *et al.* [13]).

The zoning of the target territory is showed in figure 2. We classified each area considering the maximum number of incoming and outgoing vehicles as *yellow* (less than 400 vehicles per hour), *orange* (between 400 and 600 vehicles per hour) and *red* (more than 600 vehicles per hour). The suburban areas numbered from 1 to 5 are classified as yellow, while the areas 6 and 7 are orange. At last, the areas from 8 to 12 are classified as red.

In figure 2 are shown the links between the 12 centroids of the selected territory. In our study, each link is not an abstraction of the real link between areas. For example, the area number 8 (Quattromiglia di Rende) is linked with the 12 (Cosenza Centro) by mean of a freeway (A3). Thus, we added the edge between centroids 8 and 12 whose features matches with the real freeway.

As we stated that the NN has 8 input neurons, it has one output neuron that provides the travel time for the link *(i,j)*. So, each pattern of a data set contains the simulated data regarding the 8 input neurons and the measured output (the travel time) for a fixed link *(i,j)*. A data set contains data for 1440 time periods of 10-minutes. For each time period, one pattern per each link of the complete graph shown in figure 2 is collected.

A data set has been divided into three subsets: i) *training set*, ii) *validation set*, and *iii) test set*. The first one, covering six days of observation, is used to train the NN. The validation set covers other two days of observation and is used to test the quality of the generalization capability of the system every *NL* learning epochs. Finally, the test set is used to evaluate the final performances of the NN, in terms of quality of the predictions on the last two days of observation.

3.3 System comparison and experimental setup

Collecting data about weather is not easy to do, therefore we have tested two different NN configurations. The first one (*SimpleNN* or *SNN*) has no neurons associated to data about weather conditions (*i.e.* only street conditions and traffic flow conditions data), the second one (*WeatherNN* or *WNN*) instead accepts inputs telling the system about meteorological situation and visibility conditions. Our aim is to compare these two different approaches in terms of quality of the predictions.

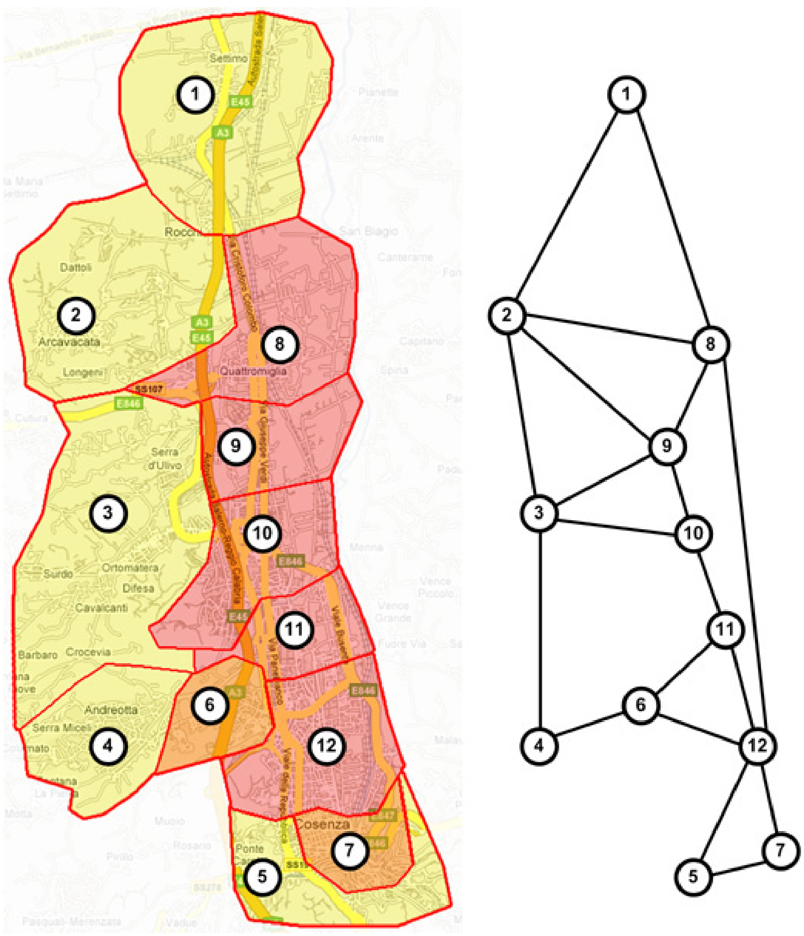

Figure 2: Target territory.

We performed the training of the NN using the *backpropagation* supervised learning technique that uses a sigmoid activation function. It is worth to mention that at this stage of the research it is not our aim to find the best training algorithm. As we have already pointed out we have been focusing on the NN configuration capable of giving good results in terms of quality of the travel time predictions.

Let m and n be respectively the number of input and hidden neurons, we have $m=6$ for the SNN, and $m=8$ for the second WNN. Preliminary tests showed that better results could be obtained using a 0.6 learning rate and $n=m-3$.

For the initialization of the weights of the networks, we used the method suggested by LeCun *et al.* [14]. The weights are obtained from a uniform distribution within the interval $\left[-3/\sqrt{m}, 3/\sqrt{m}\right]$. The biases are initialized in the range $\left[-0.1, 0.1\right]$.

We fixed the number of learning epochs *NL* to 200, so we derived the first exit rule in the training process: "if learning epochs become greater or equal to 10000 epochs stop the training".

We included other two exit rule that stop the training of the NN if *i)* the validation error is not decreasing or *ii)* the validation error is decreasing with a (nearly) const slope.

3.4 Performance analysis

In this paragraph, we compare the computational results of the two NNs (*i.e.* SNN and WNN) obtained using the dataset available at the URL [15]. Our aim is to compare the *total quadratic error* (TQE) related to both *validation* and *testing* of the SNN and WNN.

Each NN has been trained 30-times using different arc weight and bias random initialization. Values obtained during validation and testing are reported in figure 3.

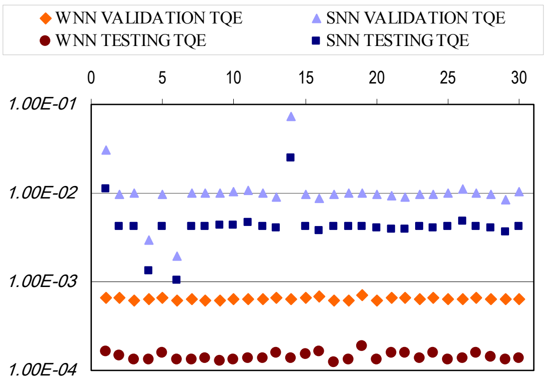

Figure 3: Validation and testing TQE for SNN and WNN.

Table 1 reports *mean*, *standard deviation*, *best case* and *worst case* of the TQE sample depicted in figure 3.

Table 1: Computational results.

Method	*Validation TQE*				*Testing TQE*			
	Mean	*StD*	*Best*	*Worst*	*Mean*	*StD*	*Best*	*Worst*
WNN	6.39E-04	2.07E-05	6.15E-04	6.98E-04	1.43E-04	1.44E-05	1.24E-04	1.88E-04
SNN	1.21E-02	1.21E-02	1.92E-03	7.18E-02	4.89E-03	4.11E-03	1.04E-03	2.51E-02

Through the designed experiment, it is confirmable that the prediction error considering both weather condition neurons provides better short-travel time prediction than not using them.

4 Conclusions and future work

In this study, two different NN models have been proposed to address the problem of the travel time forecasting. The two different NNs differ for the input data that they use to calculate the prediction of the travel times. Both use input referred to the street feature and the current traffic situation, but only one of the NNs accepts also data about the current weather conditions. The two approaches have been tested and compared on the representation of the territory of a small town in South Italy (Cosenza, Calabria). The data sets used to train and test the NNs has been derived using a microsimulator, constructed to replicate the traffic on the chosen area in different environmental situations.

Computational results have shown that better travel time forecasting is obtained using weather conditions as part of the NN input.

Future studies will be carried out to extend our approach to other testing cases. It will be interesting to see if the results obtained will remain similar in different traffic conditions (motorway corridors, highways and so on).

Different configurations of the NNs systems will be also tried and compared. It is certainly worth to test different training algorithms. The idea is to develop parallel computing methods to directly make these comparisons, but how to exploit parallel computing for the NNs is still an open problem that it is worth to better investigate.

References

[1] Kaysi, I., Ben-Akiva, M. & Koutsopoulos, H., Integrated approach to vehicle routing and congestion prediction for real-time driver guidance. *Transportation Research Record*, 1408, pp. 66–74, 1993.

[2] Junchaya, T., Chang, G. & Santiago, A., Advanced traffic management system: Real-time network traffic simulation methodology with a massively parallel computing architecture. *Transportation Research Record*, 1358, pp. 13–21, 1992.

[3] Park, D., Rilett, L.R. & Han, G., Spectral basis neural networks for real-time travel time forecasting. *Journal of Transportation Engineering*, 125(6), pp. 515–523, 1999.

[4] Klimasauskas, C.C., Neural nets tell why. *Dr. Dobb's Journal*, 16(4), pp. 16–26, 1991.

[5] Smith, B. & Demetsky, M., Short-term traffic flow prediction: neural network approach. *Transportation Research Record*, Vol. 1453, pp. 98–104, 1994.

[6] Florio L. & Mussone L., Neural network models for classification and forecasting of freeway traffic flow stability. *Transportation Systems: Theory and Application of advanced technology*, Liu B., Blosseville J.M. (eds), pp. 773–784, 1995.

[7] Park, D. & Rilett, L.R., Forecasting freeway link travel times with a multilayer feedforward neural network. *Computer-Aided Civil and Infrastructure Engineering,* 14(5), pp. 357–367, 1999.

[8] Innamaa, S., Short-term prediction of travel time using neural networks on an interurban highway. *Transportation*, 32(6), pp. 649–669, 2005.

[9] Sharda, R. & Patil, R., Neural networks as forecasting experts: An empirical test. *Proc. of the International Joint Conference on Neural Networks*, 1, pp. 491–494, 1990.

[10] Tang, Z., De Almeyda, C. & Fishwick, P.A., Time series forecasting using neural networks vs. Box-Jenkins methodology. *Simulation*, 57(5), pp. 303–310, 1991.

[11] Ritchie, S.G. & Cheu, R.L., Neural network models for automated detection of non-recurring congestion. *California Partners for Advanced Transit and* Highways (PATH). Research report, repositories.cdlib.org/its/path/reports/UCB-ITS-PRR-93-5, 1993.

[12] Godwin, L.C., Weather impacts on arterial traffic flow. *Mitretek Systems Inc.*, Research report, 2002.

[13] Kyte, M., Khatib, Z., Shannon, P. & Kitchener, F., Effect of environmental factors on free-flow speed. *Proc. Of the 4th International Symposium on Highway Capacity*, eds. Transportation Research Board, pp. 108–119, 2000.

[14] LeCun, Y., Bottou, L., Orr, G.B. & Müller, K.-B., Efficient backprop. *Neural Networks: Tricks of the Trade*, ed. G.B. Orr & K.-R. Müller, Springer-Verlag: Berlin, pp. 9–50, 1998.

[15] Centro di Supercalcolo per l'Ingegneria Computazionale (CESIC) Web Site, papers area, www.cesic.unical.it