1 Introduction

The growth in car mobility has lead to more uncertainty in travel times. As a result cardrivers have an increasing demand for information on these travel times (see e.g. Huisken and Van Berkum, 2000). Travel times can be measured using automated vehicle identification (AVI) techniques (floating car data, automated license plate recognition, etc.). However, these techniques are rarely used since they need large investments in roadside equipment. Currently travel times are estimated using data from inductive loop detectors. Since loop detectors yield spot measurements of flow and speed, travel times can only be estimated, not actually measured. Furthermore, when cardrivers are provided with information on travel times, these travel times should ideally be the times that they will encounter. Therefore we need to predict travel times, based on previous measurements. Currently two methods are being used, i.e. Static Travel Time Estimations (STTE) and Dynamic Travel Time Estimations (DTTE) (see e.g. Hounsell and Ishtiaq, 1997; Van Arem et al, 1997; and Zee, 2001). This research proposes a new travel time prediction method using an Artificial Neural Network (ANN). The three methods STTE, DTTE, and ANN methods were applied on the A13 motorway from The Hague to Rotterdam and their performance was compared.

2 Data Acquisition

This section is about the data acquisition site and the data itself. An overview of the geographical site is given along with a quantitative description of the data sets.

2.1 The geographical site

The chosen geographical location (figure 1) is the motorway A13 from The Hague to Rotterdam – one direction only. This motorway has a length of 11.4 km, 5 on- and off-ramps, a speed limit of 100 km/h, and has one gas station (approximately halfway). The section of the motorway possesses 21 locations where dual induction loop detectors are situated.
2.2 Data sets description

In this contribution two main issues will be addressed. First a method is selected that yields the best travel time estimate using inductive loop data. Second, three methods to predict travel times are being compared. For the first part two data sets have been acquired: one licence plate recognition set (Delft-Noord/Nootdorp [km 7.3 post] – Rotterdam-Overschie [km 17.55 post]) and one through the dual induction loop detectors with the MARI [More Applicatie Routekeuze Informatie] system. The licence plate recognition set was acquired by time and licence plate registration of passing red vehicles at the starting and re-identification point and subsequent subtraction produced travel times. The collection took place on October 8th and 9th 1996: 07:00 – 09:30 and 15:30 – 18:30 and on October 11th 1996: 15:00 – 18:45. During this period, also inductive loop data for the 21 locations were collected, containing flow and speed on a one-minute basis.

For the second part, i.e. the comparison of the three prediction methods, loop data was collected from November 11th 1996 – February 23rd 1997. The set also contains 1-minute aggregated data from loop detectors containing information on flow and speed.

3 METHODS AND MODEL DEVELOPMENT

3.1 Estimation of travel times using inductive loop data

From Bovy and Thijs (2000) five algorithms (RT0 - RT4) were selected and tested for the estimation of travel times using loop detector data (table 1). The estimates these methods yield were compared with the actual measured travel times using license plate recognition \textit{RT\_M\_AVG} (figure 2).

<table>
<thead>
<tr>
<th>Link method</th>
<th>Route method</th>
<th>Smoothing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>Mass balance</td>
<td>Static</td>
</tr>
<tr>
<td>RT0</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>RT1</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>RT2</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>RT3</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>RT4</td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

Table 1. Travel time estimation methods.
3.2 Prediction using static travel time estimation

STTE uses last-known link travel times and sums them. Links (L) are defined as the distance between two consecutive dual induction loops. Suppose a vehicle enters a specific route (number of links = k) at time $t = T_0$. The most recent recorded loop speeds $v_n(T_0)$, ($n = 1..k$), are assigned to half of the link before and after the specific loop location. STTE becomes:

$$STTE_{T_0} = \frac{L_2 - L_1}{2 \cdot v_1(T_0)} + \sum_{n=2}^{k-1} \left( \frac{L_{n+1} - L_{n-1}}{2 \cdot v_n(T_0)} \right) + \frac{L_k - L_{k-1}}{2 \cdot v_k(T_0)}$$

(1)

So, link travel times are assumed fixed as of the moment that the vehicle enters the route ($T_0$).

3.3 Prediction using dynamic travel time estimation

DTTE at time $T_0$ can only be estimated by historical reconstruction and is done iteratively. Suppose at $T_0$ vehicle 2 enters the motorway and vehicle 1 is the last vehicle that left the motorway. From the recorded speed data the travel time that vehicle 1 did encounter can be reconstructed. Now DTTE uses this travel time as a prediction for vehicle 2.
3.4 Prediction using artificial neural networks

ANNs are based upon biological neural networks by mimicking their architectural structure and information processing in a simplified manner. They both consist of processing elements called neurons that are highly interconnected making the networks parallel information processing systems. They are capable of tasks such as pattern recognition, perception and motor control that are considered poorly performed by conventional linear processing. These parallel systems are also known to be robust and to have the capability to capture highly non-linear mappings between input and output. ANN applications in transport can be found in e.g. Dougherty (1995) and Huiskens (1998 and 2001).

Here Multi Layer Feedforward (MLF) neural networks were used. The MLF is a supervised learning network meaning that during the training phase all inputs are mapped on desired outputs. The error, i.e. the difference between the actual and the desired output, is a criterion that is used to adjust the weights of the neurons iteratively so that the total error of all input-output pairs is minimised. The algorithm responsible for this method is called the learning rule. More comprehensive information on MLF neural networks can be found in any textbook on ANNs.

MLF networks will be trained with the most accurate travel times as targets. The number of input variables is: 21 (induction loops) * 2 (quantities: speed and intensity) + 1 (time of day) = 43. This high number of inputs resulted in time consuming training phases so pre-processing was used to cut the input down to 12 inputs. The data set was divided into 3 equally sized subsets (for cross-validation purposes) where one subset was used as test set to find the optimum number of hidden neurons and epochs (training cycles) and to prevent the MLF network from overfitting.

4 RESULTS & CONCLUSIONS

The predictions of the three methods were compared with the travel times determined by RT4. Travel time prediction becomes interesting when free flow conditions no longer hold. Therefore prediction was only executed when travel times exceeded 500 seconds (i.e. average speed under 82 km/h). To compare the method’s performances several measures of error were determined (see formula 2): the Mean Relative Error (MRE) [%], the Mean Absolute Relative Error (MARE) [%], the Mean Time Error (MTE) [seconds], the Mean Absolute Time Error (MATE) [seconds], and R-squared.

\[
\begin{align*}
MRE &= \sum_{n} \frac{\hat{t}_n - t_n}{n \cdot \hat{t}_n} \\
MARE &= \sum_{n} \left| \frac{\hat{t}_n - t_n}{n \cdot \hat{t}_n} \right| \\
MTE &= \sum_{n} \frac{\hat{t}_n - t_n}{n} \\
MATE &= \sum_{n} \left| \frac{\hat{t}_n - t_n}{n} \right|
\end{align*}
\]

(2)

where \( n \) is the number of cases, \( \hat{t}_n \) is the travel time to be predicted (target value from RT4), and \( t_n \) is the travel time generated by the model.
The results are given in table 2 that shows that MLF significantly outperforms DTTE, which in turn significantly outperforms STTE. The MRE values are also classified into 5%-error intervals and from this can be concludes that roughly two-thirds, a half, and one-third of the prediction cases fall in the [-5%, 5%] error domain (between the 2 vertical lines) for MLF, DTTE, and STTE, respectively (figure 4).

<table>
<thead>
<tr>
<th>Method</th>
<th>MRE [%]</th>
<th>MARE [%]</th>
<th>MTE [sec]</th>
<th>MATE [sec]</th>
<th>R-squared</th>
</tr>
</thead>
<tbody>
<tr>
<td>STTE</td>
<td>1.06</td>
<td>10.7</td>
<td>-2.30</td>
<td>79.5</td>
<td>0.816</td>
</tr>
<tr>
<td>DTTE</td>
<td>-1.71</td>
<td>6.91</td>
<td>-7.75</td>
<td>55.0</td>
<td>0.874</td>
</tr>
<tr>
<td>MLF</td>
<td>-0.249</td>
<td>4.61</td>
<td>-0.107</td>
<td>35.1</td>
<td>0.957</td>
</tr>
</tbody>
</table>

Figure 4. Distribution of MRE.

REFERENCES


