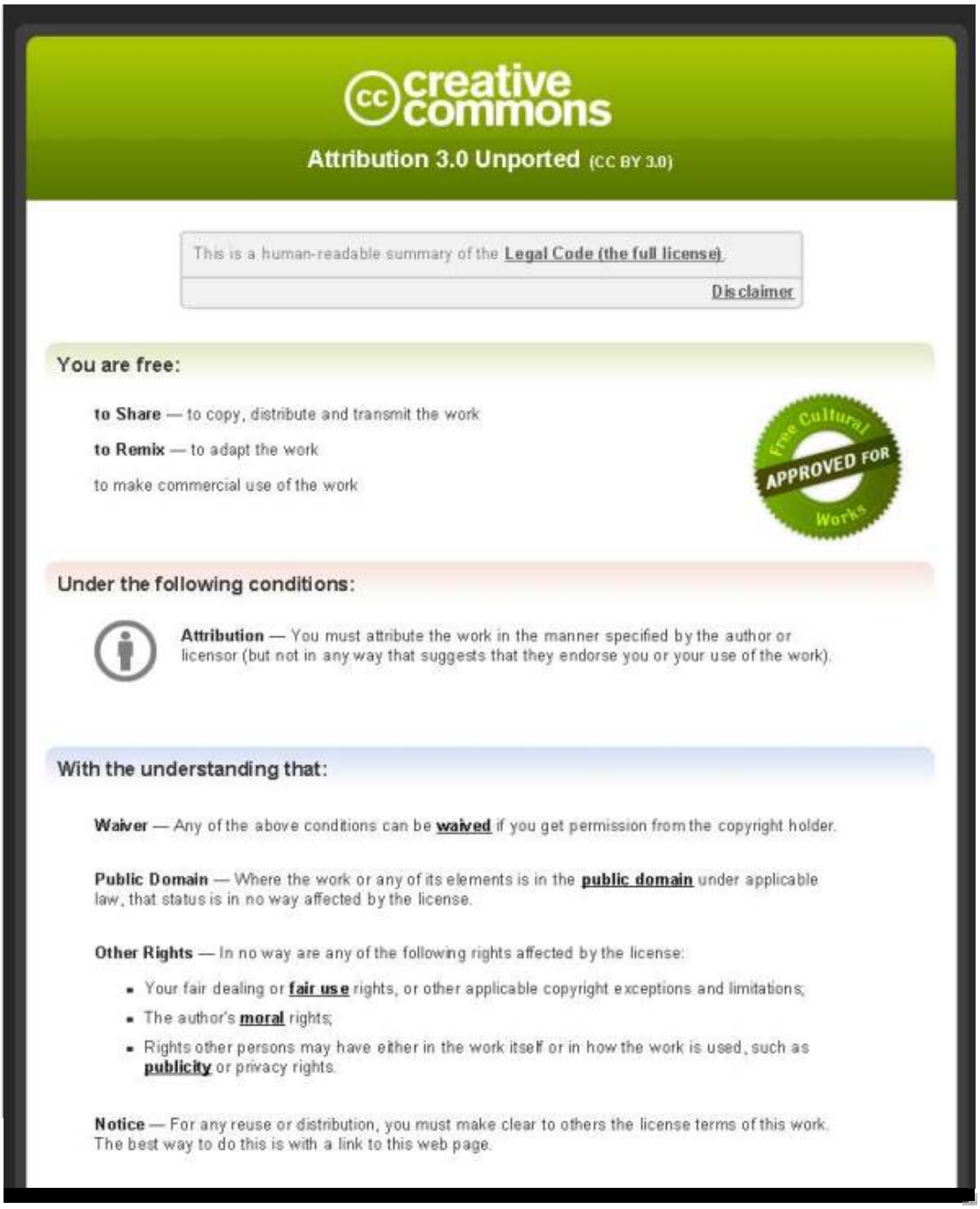


Accident Analysis and Prevention 65 (2014) 105–113

Contents lists available at ScienceDirect

Accident Analysis and Prevention

journal homepage: www.elsevier.com/locate/aap

Network-level accident-mapping: Distance based pattern matching using artificial neural network☆

Lipika Deka*, Mohammed Quddus[1]

School of Civil and Building Engineering, Loughborough University, Loughborough LE11 3TU, United Kingdom

ARTICLE INFO

Article history:
Received 14 October 2013
Received in revised form 2 December 2013
Accepted 5 December 2013

Keywords:
Accident-mapping
Pattern-matching
Artificial neural network

ABSTRACT

The objective of an accident-mapping algorithm is to snap traffic accidents onto the correct road segments. Assigning accidents onto the correct segments facilitate to robustly carry out some key analyses in accident research including the identification of accident hot-spots, network-level risk mapping and segment-level accident risk modelling. Existing risk mapping algorithms have some severe limitations: (i) they are not easily 'transferable' as the algorithms are specific to given accident datasets; (ii) they do not perform well in all road-network environments such as in areas of dense road network; and (iii) the methods used do not perform well in addressing inaccuracies inherent in and type of road environment. The purpose of this paper is to develop a new accident mapping algorithm based on the common variables observed in most accident databases (e.g. road name and type, direction of vehicle movement before the accident and recorded accident location). The challenges here are to: (i) develop a method that takes into account uncertainties inherent to the recorded traffic accident data and the underlying digital road network data, (ii) accurately determine the type and proportion of inaccuracies, and (iii) develop a robust algorithm that can be adapted for any accident set and road network of varying complexity. In order to overcome these challenges, a distance based pattern-matching approach is used to identify the correct road segment. This is based on vectors containing feature values that are common in the accident data and the network data. Since each feature does not contribute equally towards the identification of the correct road segments, an ANN approach using the single-layer perceptron is used to assist in "learning" the relative importance of each feature in the distance calculation and hence the correct link identification. The performance of the developed algorithm was evaluated based on a reference accident dataset from the UK confirming that the accuracy is much better than other methods.

1. Introduction

In 2012 Great Britain saw 1754 deaths, 23,039 seriously injured and a total of 195,723 casualties in reported road accidents (DoT, 2013). World Health Organisation estimates over 1 million deaths world-wide as a result of road accidents (WHO, 2013). To make roads safer and save life and money, understanding the safety performance of the underlying road network and identifying link-level accident hot-spots so as to design engineering countermeasures are critical. Hence, accurate assigning of accidents to the correct road segments where the accidents occurred is a vital precursor for safety related applications such as accident risk modelling, risk mapping and accident hot-spot identification. Accident risk modelling and link feature identification are also essential for the design and manoeuvring of intelligent, self-driving vehicles of the future.

Data and information on traffic accidents (such as their geographical references in terms of road name, district name, accident location denoted as *x*- and *y*-coordinates, number of casualties and their characteristics, number of vehicles/types involved) in most countries are recorded by the police by either visiting the place of accident or by conducting remote inquiries. Due to reasons such as the situation at the accident site, accuracy issues related to positioning methods/instruments such as GPS or national grid reference (Quddus et al., 2007), mistakes on part of the police, etc., errors exist in the police recorded accident data (Shinar et al., 1983; Levine et al., 1995; Austin, 1995; Aptel et al., 1999; Loo, 2006; Tarko et al., 2009; Khan et al., 2004). For example, in the UK (Austin, 1995) as well as in Abu Dhabi (Khan et al., 2004), location of the accident has been identified as the most inaccurately recorded data item. Shinar et al. (1983) reported that in the United States, highway feature data such as gradient, speed limit, surface composition and curvature were the most inaccurately reported information, whereas accident location, date, passenger and vehicle information were the

* Corresponding author. Tel.: +44 01509223401.
E-mail addresses: l.deka@lboro.ac.uk (L. Deka), m.a.quddus@lboro.ac.uk (M. Quddus).
[1] Tel.: +44 01509228545.

0001-4575/$ – see front matter.
http://dx.doi.org/10.1016/j.aap.2013.12.001

most reliable data. More recently Tarko et al. (2009) reports that large number of missing accident data entries, spelling mistakes in road names, presence of alternate road names poses issues in the US.

Such data inaccuracies invalidate and significantly affect any analysis rooted in it. Hence, accident records must be validated and accidents should be mapped on to the correct links before being analysed for the purpose of enhanced road safety applications. The objective of an accident-mapping algorithm is to snap traffic accidents onto the correct road segments and correct position on the selected segment, given inaccurately recorded location information.

Much of the accident-mapping research effort in the past has been towards identification of mistakes in road accident data records (Shinar et al., 1983; Levine et al., 1995; Austin, 1995). This was initially done manually and through the use of computer validation techniques (Shinar et al., 1983) and later through the use of Geographical Information System (Austin, 1995). Natural language understanding techniques have also been used to retrieve information from accident reports written in free format plain English and used to validate the accompanied records in pre-defined formats (Wu and Heydecker, 1998). In the near past, significant progress has been made in not only identifying mistakes in police reported accident records, but also in correcting those mistakes to identify correct road segments where the accident took place. The underlying concept of most of these accident-mapping endeavours has been towards integration of the accident database of police accident reports with the road network database. These accident positioning attempts include GIS-based approaches of snapping accidents to nearest road segment or junction and then iteratively validating and correcting associated variables such as district and road name (Loo, 2006). Dutta et al. (2007) used the information of accident location, direction and distance from the accident location to develop an approach of mapping the accident position on the correct intersection or local road segment. Probabilistic record linkage methods have also been used to link erroneous accident record database with the road-network database, thereby positioning accidents on the correct road segments (Tarko et al., 2009). Although these efforts have greatly improved the quality of accident location data, but positioning accuracy is compromised in complex road network scenarios such as presence of multiple parallel roads, roundabouts and other types of junctions. Such compromises are due to the limitation in the heuristic techniques (Loo, 2006) and probabilistic formulas (Tarko et al., 2009) used as well as due to the limitations in the type of data available.

The aim of this paper is therefore to develop a new accident mapping algorithm based on the common variables observed in most accident databases (e.g. road name and type, direction of vehicle movement before the accident and recorded accident location). Our approach involves representing an accident as well as all road links as feature vectors of these variables and a distance based pattern matching technique is employed to map an accident to the link with which its "pattern" or feature vector matches most closely. Since, each feature do not contribute equally towards the identifications of the correct link (Quddus et al., 2007; Tarko et al., 2009; Velaga et al., 2009; Greenfeld, 2002), an artificial neural network is employed to "learn" the relative significance of the above stated features. Once the correct link has been identified, the accident position on the link is determined through perpendicular projection of the accident location on the selected link. In the case where the perpendicular projection of the accident location falls outside the link, the closest end point of the link is fixed as the point of accident.

The performance of approach was evaluated against a reference accident dataset that was compiled through manual mapping of accidents onto links through the use of GIS software. Additional variables such as vehicle position at time of accident, second road name (in the case of junction accidents) were used during manual mapping apart from the variables used in our algorithm. In the absence of correct reference data on accident location, any form of mapping must be treated with caution. Manual mapping is extensively labour intensive and hence we evaluated our results against only a subset of 560 accidents from the accident data set on UK's strategic road network for the year 2012.

The remaining sections present the detailed description of the approach (Section 4) and evaluation (Section 5) result by first presenting a brief literature review (Section 2) and overview of the existing challenges (Section 3).

2. Related work

The police department of different countries collect data on traffic accidents with subtle difference in the type of data from country to country. For example, an accident location in Wisconsin is recorded as the direction and distance from a junction (Dutta et al., 2007) whereas an accident location in the UK is recorded in terms of its geographic co-ordinates (Austin, 1995). This section presents the existing accident-mapping techniques that utilise the location specific available accident data.

Loo (2006) developed a GIS-based spatial data validation methodology to map accident locations in Hong Kong to a precise road section. The methodology snaps an accident to the nearest junction if the accident occurred at a junction else snaps it to the nearest road. The approach then checks to validate the road and district name of the mapped accident location to that of the original recorded accident. In the case of a mismatch in the district name, the algorithm amends the incorrect field with the correct data associated with the accident-mapped location. In the case where the road name does not match, the algorithm maps the accident location to the next nearest road or junction, this time amending the accident record with the current mapped road name in case the original road name still does not match.

Tarko et al. (2009) employed the concept of probabilistic record-linkage using the Fellegi–Sunter model (Fellegi and Sunter, 1969) with the Expectation-Maximization (EM) method to map accidents to road segments. The features used for accident-mapping were County ID, Township ID, City ID, main road name, reference road name, shoulder type, median presence and junction type. The Fellegi–Sunter model estimates the probability of the occurrence of an accident on a road through pair-wise matching of features in records from respective datasets. Each feature is adjusted with weights, where weights determine an attributes relative contribution towards the final decision of match/no-match. Probabilities above a certain threshold will translate to a match, below a second threshold translates to a no-match and any probability in-between suggests human intervention for the final decision of a match/no-match. The EM method is used to estimate the respective feature weights. The method was evaluated using a test sample of 137 real and simulated accident data from the state of Indiana in the USA and saw that even though almost all accidents were mapped, 80% of accidents were mapped to more than one link and mapping to intersections were not very efficient.

Dutta et al. (2007) developed a tool to digitally plot Wisconsin's local road accidents on a GIS map integrating it with complete information on the mapped accident. Two primary data sources namely the Wisconsin Accident Database and the Wisconsin's Information System for Local Roads were used and accident mapping for intersection accidents and segment accidents were done separately. Accident mapping methodology mainly involved parsing portions of street names (i.e. prefix, name type and suffix component) of each accident record and matching them against records in the road

network data set to identify intersections. Accident positions were then fixed on links using the given information on direction and distance from the intersection.

Though, considerable progress has been made in improving accident-mapping accuracy, limitations exit. Correct accident mapping as opposed to map-matching (Quddus et al., 2007; Velaga et al., 2009; Greenfeld, 2002) depends on only the police recorded data and there is no historical data such as trajectory information of the involved vehicles to augment the accuracy of accident-mapping. The approach developed by Loo (2006) considers only the two closest roads/junctions and this approach may fail in regions of high road density, especially at a complex fly-over. Moreover, failing to consider the direction in which the vehicle(s) involved in the accident were travelling may lead to a final accident mapping on a road whose heading is opposite to that of the vehicle. It is seen from the method developed by Tarko et al. (2009) that the dataset must include sufficient information for more precise mapping and even though the probabilistic method used could identify probable links on which the accident actually occurred, it is believed that methods such as neural networks can achieve much higher levels of accuracy (Wilson, 2011). Wisconsin's accident mapping method (Dutta et al., 2007) was relatively simple due to the type of data (direction and distance from an intersection). This method cannot be easily 'transferable' as it was developed for a very specific accident dataset. Existing literature shows that there is a need for higher accuracy accident mapping approaches that takes into consideration both simple and complex road networks and can also be adapted to different datasets.

3. Raw data and its limitations

In the United Kingdom the police records details of the accident in a form referred to as STATS 19, following instructions given in a manual called STATS 20 and the recorded police data is post validated for accuracy employing techniques described in STATS 21. Data covering all aspects of an accident is reported within variables divided into three groups namely attendance circumstances variable, vehicle variable and casualty variables in the STATS 19 form. Some of the locational variables from the STATS 19 form include:

- *Location*: The location of the accident is coded within the British grid coordinates easting and northing.
- *Road class*: This variable gives the code number for the class of the road on which the accident location in being recorded. The classes of road being motorway, A(M), A, B, C and unclassified.
- *Road number*: The road number of the road whose class has been recoded as above is entered within this variable. In our algorithm, we concatenate the road class and number and term it as road name. For example if the road class is M and number 25, the road name is M25.
- *Road type*: Appropriate code of the road type on which accident has occurred is given within this variable. The road types include roundabout, one way street, dual carriageway, slip road and unknown.
- *Junction detail*: If an accident occurs within 20m of a junction the junction detail is entered in this variable.
- *Vehicle movement compass point*: The direction of travel of each vehicle involved in an accident is recorded in two variable termed 'to' and 'from'. The codes entered in these variable include the true compass directions north, north east, east, south east, south, south west, west and north west.

Other variable include 2nd road name, speed limit, manoeuvres, vehicle location at time of accident, weather, road surface condition, etc.

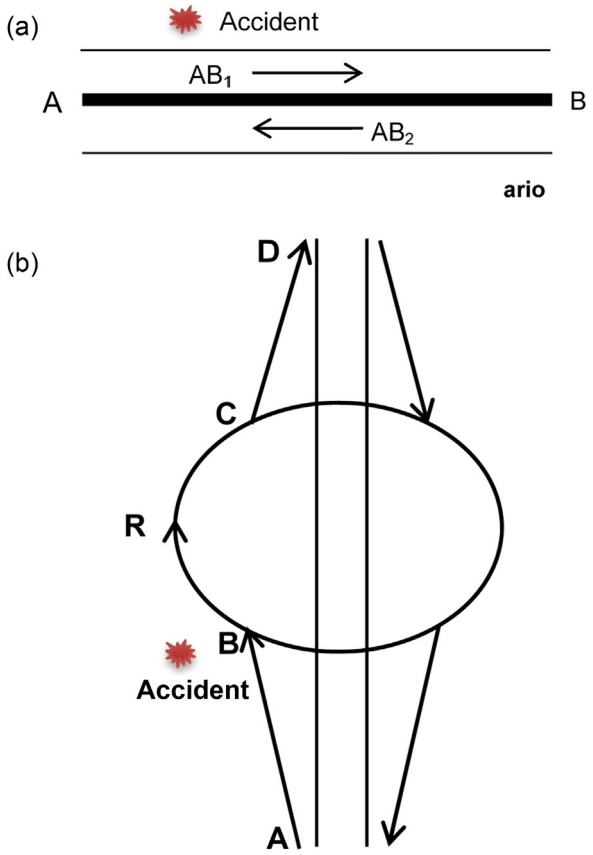


Fig. 1. (a) Challenging crash-mapping scenario and (b) challenging crash-mapping scenario. (For interpretation of the references to colour in the text, the reader is referred to the web version of the article.)

Inaccuracies in any of the recorded variables lead to challenging positioning scenarios. For example:

- In Fig. 1(a) the avilable accident data tells us that the road name is AB. The location coordinates of the accident is as seen in the figure (the red asterix) and the vehicle movement is from East to West. AB is a dual carriageway with the directions of the two sections as indicated by the arrows. We can see that both sections, AB_1 and AB_2 of the dual carriageway is AB and the location of the accident tempts us to map the accident on section AB_1. But, the accident actually occurred on AB_2 and this can only be determined from the vehicle movement compass point data along with the road name.
- Consider the road layout in Fig. 1(b). The location coordinates of the accident is detoned by the red asterix and the vehicle movement is given to be from south to north. We can see both the slip road (i.e. section AB) and the roundabout (i.e. section named R) matches the description above and accident location is located at almost equal distance from each. The accident could be correctly mapped to AB using the road type information which was given as slip road.

From the two examples which are representative of real data from the 2011 accident data set, it is clear that no single field in the police accident report but only a combination of fields can accurately determine the exact accident road segment. Moreover, inaccuracies can occur in any variable and the proportion of inaccuracy is not fixed. For example, the distance from an accident (given by the geographic coordinates) to the correct accident-mapped road segment is about 20 m whereas the distance of another

accident from its correctly mapped segment is 50 m. Hence, the challenges here are to: (i) develop a method that takes into account uncertainties inherent to the recorded traffic accident data and the underlying digital road network data, (ii) accurately determine the type and proportion of inaccuracies, and (iii) develop a robust algorithm that can be adapted for any accident set and road network of varying complexity.

4. Artificial neural network based accident mapping methodology

4.1. Artificial neural network (ANN)

Artificial neural networks are mathematical methods inspired by biological neural networks and are utilised for solving problems in pattern recognition, speech recognition, prediction, optimisation etc. (Jain and Mao, 1996; Hanspal et al., 2013). An ANN model can be viewed as a function that transform's a set of input variables to a set of one or more output variables.

$$f : X \rightarrow Y$$

For example, in the case of traffic signal recognition, the input features activating the network can be the pixel values of the traffic light image and the output is a three bit binary number indicating red, amber and green.

The architecture of an ANN model is similar to weighted directed graphs where the nodes are the artificial neurons of a neural network. Neurons are arranged in layers, with a simple 3 layer ANN as seen in Fig. 2(a) being composed of the first layer of input neurons, a second hidden layer and the third layer of the output neurons. The weighted directed edges of the ANN graph are interconnections between neuron outputs of one layer and neuron inputs of the next layer. Depending upon the directions of connection, ANNs can either be feed-forward (i.e. graphs with no loops) or feedback networks (i.e. graphs with loops). Each neuron is a computational unit that takes in weighted inputs and produces outputs according to its associated activation function. The different activation functions in use are threshold, piecewise linear, sigmoid, Gaussian, etc. (Jain and Mao, 1996). Fig. 2b presents McCulloch–Pitts model of a neuron with a threshold activation function. The neuron computes a weighted sum of the inputs and outputs a 1 if the sum h is above the threshold u and 0 otherwise.

What makes an ANN "intelligent" and hence an attractive problem solving tool, is its ability to learn. It is because of this ability that ANNs have been widely used to solve problems which are difficult to solve with rule based programming approaches. An ANN learns from examples through the process of updating its architecture and connection weights to perform the required task. Learning can either be *supervised learning* (each input is provided with the correct output in the training process and the network adjusts parameters to produce the correct output), *unsupervised learning* (no correct output is provided, but the model explores the solution space and discovers correlations and patterns) and *hybrid learning* (a combination of supervised and unsupervised learning). Connection weights are updated according to learning rules such as error-correcting, Boltzmann, Hebbian and Competitive learning. For example, the basic principal behind the error-correcting rule which is mainly associated with the supervised learning paradigm is to use the difference between the desired output (obtained from the given training set) and the actual output iteratively to gradually reduce the error. Learning rules are associated with learning algorithms which specify how learning rules are applied to adjust connection weights. Examples of learning algorithms being the perceptron learning algorithm, back-propagation algorithm, Boltzmann learning algorithm, linear discriminant algorithm, etc. Each learning rule as well as learning algorithm is associated with particular network architecture. For example, the single or multiple layer perceptron is associated with the supervised learning paradigm, error-correction learning rule and perceptron learning algorithm. The reader is directed to the tutorial by Jain and Mao (1996) for a comprehensive overview of ANN concepts.

4.2. Method

In the accident-mapping algorithm described in this paper the variables from accident records used in the accident-mapping process are: the *road name*, *road type*, the *position* of the accident in terms of the geographic Cartesian coordinates, easting and northing and the accident angle. The accident angle is derived from the vehicle compass point variables "to" and "from", for example, if "from" is East and "to" is West, the accident angle is considered to be 270°. The angle is measured with respect to the north being taken as 0°. When more than one vehicle is involved in an accident (especially at a junction) the average of individual accident angles is considered in the algorithm. For the purpose of encoding the solution space, a road segment is treated as a poly-link of one or more straight lines/links and each link is described by its features namely the *road name*, *road type*, *heading* and *position* which is the geographic coordinates on the link closest to the currently mapped accident. It must be noted here that the description of the same link differs in the *position* value in the context of different accidents.

Firstly, for each accident the algorithm selects a set of candidate links that fall within (either inside or touching) an error bubble as suggested by Velaga (2010) for the case of matching GPS fixes on to a link. The radius of the error bubble has been determined empirically and fixed at 150 m. Each candidate link is coded as a feature vector consisting of the above stated features and it is basically a description of the state of a vehicle travelling on the link. Similarly, an accident is also represented as a feature vector of the above features to represent the location of the accident as recorded in the police report. For example in Fig. 3(a), P_{ACC} is the accident point and AB and AC are the two candidate links. To obtain the feature vector of AB corresponding to accident P_{ACC} we determine:

- the co-ordinates (X_{AB}, Y_{AB}) of the perpendicular projection of the accident on AB, thus obtaining the closest position of the accident on the link. In case the perpendicular projection of an accident lies outside the candidate link, the closest end point on the link is taken as (X_{AB}, Y_{AB}).
- Candidate link heading from the start and end coordinates of the link with respect to the northerly direction taken as 0°.

Hence, the feature vector of accident P_{ACC} is: [(X_{ACC}, Y_{ACC}), 90°, AC, carriageway] and of link AB and AC with respect to P_{ACC} is: [(X_{AB}, Y_{AB}), 45°, AB, slip road] and [(X_{AC}, Y_{AC}), 90°, AC, carriageway] respectively.

The goal of the algorithm is then to perform approximate pattern matching of the feature vector of an accident and each of its candidate link feature vector and filter out the candidate links into the category "match" or "no match". Due to the inaccuracies existing in the accident data as discussed above, exact pattern matching approach is not successful in accident-mapping. Hence, we employ a form of approximate or distance based pattern matching, where the distance is defined as the amount of alteration that must be done to one vector (i.e. the accident vector in this case) to match exactly the other vector (i.e. the candidate link feature vector). This form of distance calculation is termed as Lavenshtein distance or edit distance (Navarro, 2001). Considering that an accidents "pattern" is a corrupted form of one of the candidate links "pattern", the aim of the approximate pattern matching algorithm is hence to select the link with the least distance as the accident-mapped link.

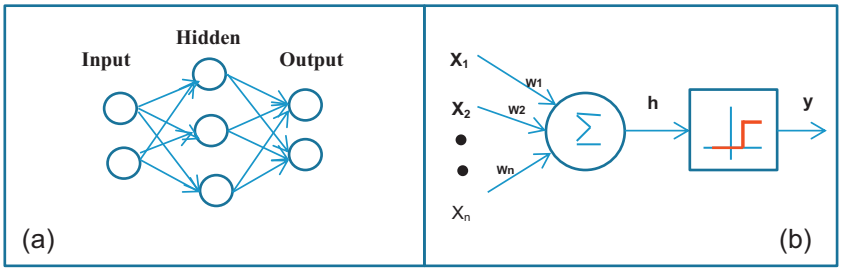


Fig. 2. ANN examples: (a) a typical 3 layer ANN architecture and (b): McCulloch–Pitts model of a neuron.

Edit distance (ED) is calculated from individual features edit distances. In the case of *position* the edit distance is the Euclidean distance between the accident position and the projected position of the accident on the link. For the accident angle, the edit distance, θ is the numerical difference between the calculated heading of the link and the accident angle. Whereas, edit distance of non-numerical feature values (such as the road name and road type) is 1 if the two do not match and 0 otherwise. Calculation of edit distance (ED) is illustrated by an example in Fig. 3(b). In principle, lesser the edit distance, closer is the match.

Accident-mapping (Tarko et al., 2009) as well as map-matching approaches (Velaga, 2010) have suggested that each feature do not have the same relative significance in the accident-mapping or map-mapping process. Hence, the distance measure of each feature is multiplied by a weight representing its relative significance. Higher the weight, greater is the respective features contribution in the final link selection. In order to be consistent with the respective weight and distance values, we transform the individual edit distances such that lower the distance, the higher is its value. In the case of actual distance of the accident coordinates from the candidate link, the Euclidean distance Δd is transformed into a functional form $f(\Delta d)$ as follows:

$$f(\Delta d) = \frac{D - \Delta d}{D}$$

The value of D is taken as 80 as suggested by Velaga (2010). The heading difference, $(\Delta\theta)$, between the candidate link and the accident angle is transformed by considering it as the cosine function of $\Delta\theta$ as suggested by Velaga (2010) and Greenfeld (2002). Doing so ensures that if $\Delta\theta$ is small the final weight will be large and vice versa. Similarly, a "match" is denoted by 1 (i.e. $I = 1$) and "no-match" by 0 (i.e. $I = 0$) in case of road name and road type. The weighted sum of the transformed individual feature distance is then used to measure similarity between an accident and candidate link vector. We term the weighted sum of distances as the *weighted edit distance* (WED) as illustrated in Fig. 3(b). Higher the value of WED, greater is the possibility of a "match" with the corresponding candidate link.

Tarko et al. (2009) estimated the different weights probabilistically, whereas Velaga (2010) derived them empirically. Inaccuracies in the data do not follow any definite rule. While in some accident records, errors may have been in the road name or road types while in another record the error is location information of the accident. Moreover, the magnitude of inaccuracies in the numerical values location data is also indefinite. As such it is hard to determine a function aggregating the different variables and determining the correct accident-mapped link. Because of an ANN's inherent ability to "learn" from observations, the accident-mapping approach presented in this paper employs an ANN to estimate the weights of each variable in the feature vector.

4.3. ANN model

Accident-mapping complexity is higher when an accident occurs near a junction specially a roundabout due to the intersection of a number of roads at close proximity. Such differences in complexity in the road network have also been indicated in map-matching problems by Quddus et al. (2007) and Winter and Taylor (2006). Hence, we divided our training set into accidents that occurred at a roundabout and those that did not and generated separate ANN models for each set. Winter and Taylor (2006) suggested the use of a modular neural network with sub-networks solving sub-tasks. Such amalgamation of individual intelligent units into a single unit improves the generality of the method rather than its predictive accuracy.

The ANN model used in this study is Rosenblatt's single-layer, feed-forward perceptron with threshold activation function and perceptron learning algorithm as discussed below. A single-layer perceptron is the simplest form of neural network used to classify patterns that are linearly separable. An initial data analysis on accident mapping results compiled manually showed that the "match" and "no match" candidate links within each group ("roundabout" and "no roundabout") are linearly separable when plotted against the four features discussed above. It has also been shown by Wilson (2011) that a single layer perceptron provides a higher accuracy as compared to probabilistic techniques for the purpose of record

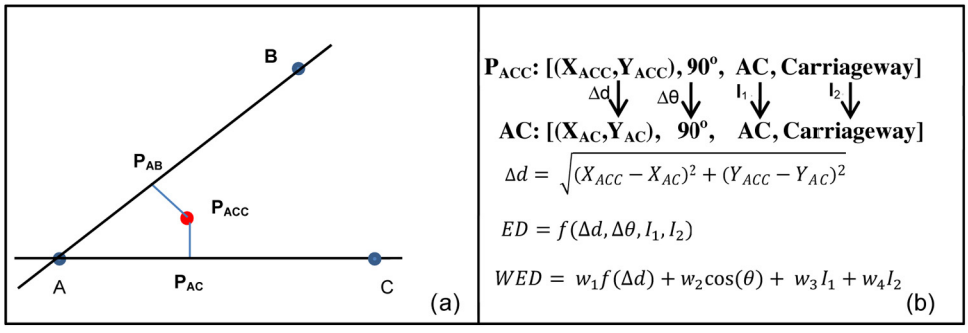


Fig. 3. (a) Feature vector and (b) edit distance between feature vectors.

linkage, a technique similar to pattern matching used by Tarko et al. (2009) for the purpose of accident-mapping. Single-layer perceptron's simplicity is both its strength and limitation and our study relied on its simplicity to provide us with an initial proof of concept of the technique and the results so far have been promising.

The perceptron model used for performing accident-mapping as depicted in Fig. 4 is built around McCulloch–Pitts model of a neuron (Jain and Mao, 1996). It takes as input an input vector X, which may consist of both binary and non-binary values and another input called the *bias*, computes a weighted sum of all the inputs which then is subjected to a threshold activation function. Accordingly, if the weighted sum is above a certain threshold u the perceptron outputs a 1 else 0.

In our application, the input vector is the vector containing the transformed distances as discussed above between respective features of an accident vector and a candidate link vector. Hence, the input vector consist of two non-binary numerical values $f(\Delta d)$ and $\cos(\Delta\theta)$ and two binary values corresponding to road name and road type with same values in the two vectors depicted by a 1 and 0 otherwise. The output value is a binary value with a 0 indicating a "no match" (i.e. $I_1 = 0$ or $I_2 = 0$) and 1 indicating a "match" (i.e. $I_1 = 1$ or $I_2 = 1$). The value of the weights, bias and the threshold value are "learned" by the perceptron using the perceptron learning algorithm described as:

Perceptron learning algorithm:

Step 1: Initialise the weights and threshold to small random numbers.
Step 2: Present the feature vector $X = [x_1, x_2, \ldots, x_n]$ and evaluate output y according to the threshold function given in Fig. 3.
Step 3: Update the weights according to

$$w_i(t+1) = w_i(t) + \eta(d - y)x_i$$

where d is the desired output, t is the iteration number and η $(0 < \eta < 1)$ is the gain (step size).

The perceptron learning algorithm iterates through a training set of pair-wise transformed distance vectors (of accident and corresponding candidate links) and its associated output, updating the weights after each run of the entire training set. Training stops at the end of a predefined number of iterations or when the desired performance (i.e. the mean absolute error falls below a certain threshold) level has been achieved. This form of learning rule for updating weights is called the error correcting rule. It must be noted that the ANN may require retraining before it can be used on a different type of road network.

The trained perceptron was then tested for its performance and it was seen that some accidents were mapped to more than one candidate link, meaning that the weighted sum of the distances of more than one feature vector pair of an accident was higher than the estimated threshold. Such similar results have also been reported by Tarko et al. (2009). Hence, once the weights have been estimated, the algorithm calculated the weighted edit distance of each accident and candidate-link pair. The candidate link with the highest WED (hence, the lowest ED) and whose road name and road type matched with the recorded accident data was than selected as the final mapped link. In the case where none of the candidate links road type and/or road name matched with that of the accident, the candidate link with the highest WED was selected as the correct link.

5. Implementation and performance evaluation

Almost all subtasks of the proposed algorithm described in the previous section have been implemented in the MATLAB programming environment. We utilised MATLAB's ANN toolbox to implement a single-layer, feed-forward, 4-input perceptron for the purpose of "learning" the relative weights of features used for our accident-mapping application. Input weights and biases were initialised with random numbers and the perceptron learning function *learnp* has been used for learning the weights and bias. MATLAB's *hardlim* transfer function encoding the threshold activation function was used for calculating the output. The performance of the network was measured using the function *mae* (*mean absolute error*) within which the aim is to minimise the mean of the absolute differences of the actual and predicted output for each input vector in the training set. The perceptron used for estimating feature weights in the case of accidents in roundabouts reached its performance goal in 14 iterations whereas the perceptron used for estimating feature weights for rest of the accidents took approximately 100 iterations to converge. The algorithm can match 29 accidents per minutes and this includes the time from training to validation with 90% of the time spent on pre-processing (including candidate link selection). The implementation ran on a 2.8 GHz Pentium CPU and 8192 MB of RAM.

To be able to train the two perceptron's and later reliably evaluate and validate our developed algorithm, the training data set needs to be representative consisting of accidents that took place across the entire road network environment including different road types such as slip roads, carriageway and roundabouts, junction and non-junction accidents. Therefore, to generate the training data set we randomly selected 400 accidents (350 non-roundabout and 50 roundabouts) from the 2011 UK accident data set ensuring that all network environments is proportionately represented. Road network data used in this study has been obtained from the UK's Highways Agency Pavement Maintenance Systems (HAPMS). Each accident was manually mapped onto road segments taking into consideration road name, type, vehicle compass direction and recorded geographical coordinates along with other variables such as second road name, junction details and speed limits. The training set totalling 800 vectors was then compiled to consist of one "match" candidate link and one "no match" candidate link for each accident.

The feature weights estimated by the two trained perceptron's were then used to implement the algorithm using the 10,520 accidents occurred in the year 2012 on the UK Highways Agency's Strategic Road Network (SRN). The algorithm was successful in mapping 100% of the accidents. In order to evaluate the performance of the algorithm, 560 accidents were randomly selected from the 10,520 accidents using the quota sampling. The 560 accidents were then manually mapped by using additional variables such as vehicle position at time of accident, road name and type, junction details, speed limits and second road name and type (in the case of junction accidents). Ideally, the performance of the proposed approach should be evaluated against existing accident-mapping approaches discussed in the literature review section, but due to differences in the type of accident data used in each of the existing methods, we are unable to implement them. An alternative approach has been adopted in the performance evaluation. Two basic commonly used crash mapping algorithms were implemented and the performance of the ANN-based developed algorithm was compared against these two algorithms using the same validation dataset (i.e. 560 accidents). The two algorithms are:

1. Closest-link (CL) algorithm: in which an accident is mapped to the closest road segment.
2. Weight-based (WB) algorithm: in which an accident is mapped to one of the candidate links that fall within a 100 m radius from the accident location. The final segment was calculated based on the heading weight (W_h) and the proximity weight (W_p) as

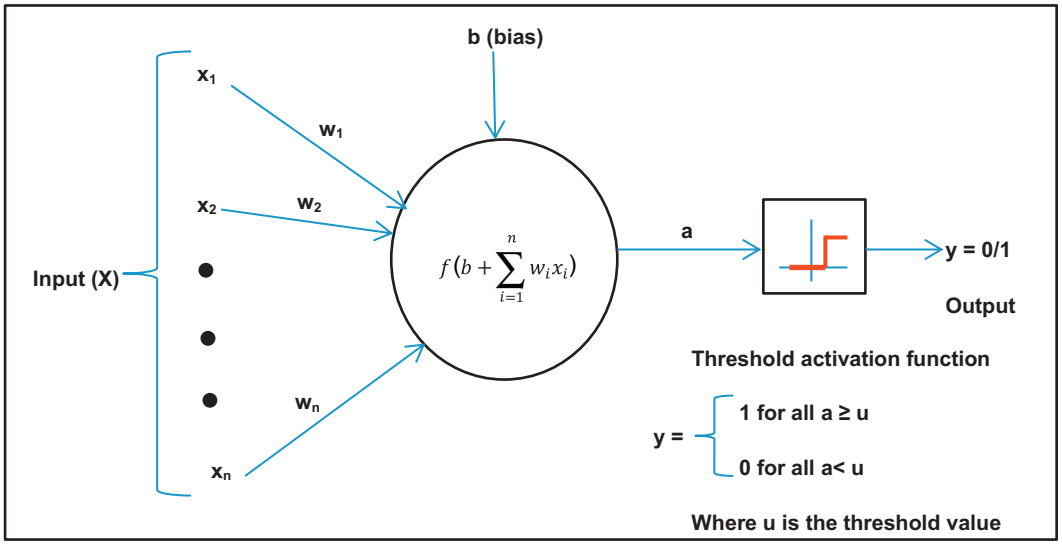


Fig. 4. The perceptron model used for the accident mapping application.

suggested by Velaga (2010) and Greenfeld (2002). The candidate link with the highest total weight score (TWS) is selected as the correct link. The TWS is calculated as follows:

$$\text{TWS} = W_h + W_p$$

$$W_h = H_w f(\theta)$$

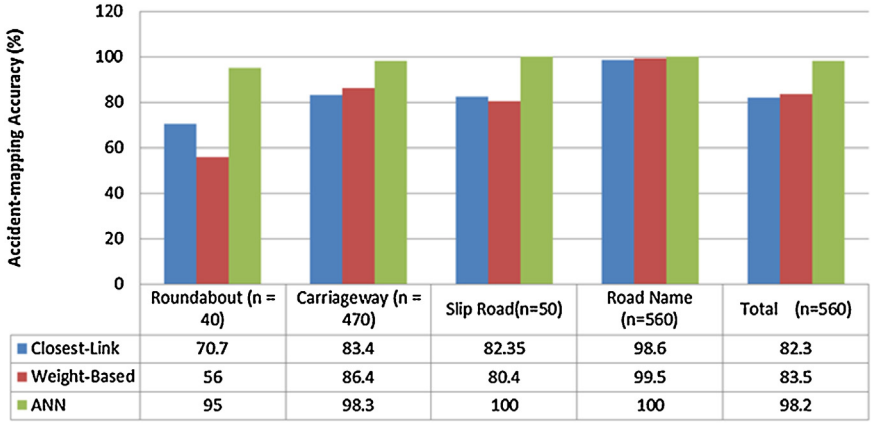
$$W_p = H_p f(\Delta d)$$

H_w and H_p denotes the heading weight coefficient and proximity weight coefficient respectively. $H_w = 0.6$ and $H_p = 0.4$ respectively and $f(\theta) = \cos(\Delta\theta)$ where $\Delta\theta$ is the difference between the heading of a candidate link and the corresponding accident angle and $f(\Delta d) = (80 - \Delta d)/80$ where Δd is the perpendicular distance of the accident point to the candidate link. The values of the weight coefficients were derived empirically for a set of independent dataset.

The accident-mapping accuracy performance of the three algorithms disaggregated by accidents occurred in different operational environments (i.e. roundabout, carriageway and slip road) and road name is given in Fig. 5.

It is clearly seen that the accuracy of ANN-based algorithm is much better than that of either the CL or the WB approach. On analysing the errors produced by the ANN algorithm, it is noticeable that the accident-mapping accuracy is lower for accidents reported to occur at roundabouts s compared to the other road types. This is primarily due to the use of heading data employed in the ANN for the case of accidents involving multiple vehicles in which the average of all vehicle headings was used as the accident heading. Similarly, it has been noticed that more than 50% of the mismatches of accident-mapping on carriageways were due to the incorrect vehicle headings from the accident data (with the potential maximum error of 22.5°). We attribute such errors as the limitations in the available data with regards to the vehicle movement direction. Mismatches were also noticed where the reported accident position is at equal distance from the two adjacent road segments with the similar headings, road name and type. The remaining mismatches were those that occurred within 20m of a junction and can be attributed to the limitation of the developed ANN-based algorithm. We believe such errors can be mitigated by extracting the different type of junction accidents and developing separate perceptron models for each type similar to the approach for roundabout accidents adopted in this research.

One may argue that mapping run-off-road accidents (in which a vehicle leaves the carriageway) onto the correct segments would be more challenging. In order to investigate this, the performance of the developed ANN method was evaluated for run-off-road



	Roundabout (n = 40)	Carriageway (n = 470)	Slip Road(n=50)	Road Name (n=560)	Total (n=560)
Closest-Link	70.7	83.4	82.35	98.6	82.3
Weight-Based	56	86.4	80.4	99.5	83.5
ANN	95	98.3	100	100	98.2

Fig. 5. Accuracy of accident mapping by road type.

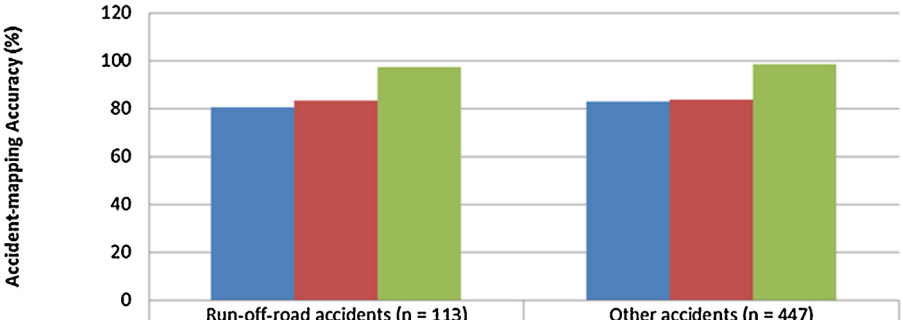


Fig. 6. Accuracy of accident mapping for run-off road vs. other accidents.

accidents (i.e. single vehicle accidents) and the results are presented in Fig. 6.

It can be seen from Fig. 6 that the performance of the developed ANN accident-mapping for run-off-road accidents is similar to that of other accidents (i.e. the difference in accuracy performance is 1.1%). This can be attributed to the fact that the reported location of an accident is usually the point of first impact or the point where a vehicle leaves the roadway. Therefore, the reported location of an accident does not incur any additional error to the input (i.e. accident location and heading) of the developed ANN method.

6. Conclusion

This paper develops a new machine learning approach for reliable and accurate mapping of traffic accidents onto the correct road segments where the accidents actually occurred. Given that the type of inaccuracy in police recorded accident data and given that the scale of such inaccuracy cannot be scientifically determined, our approach employs an ANN approach to learning the relative importance of each possibly inaccurate feature and then uses a distance based pattern matching approach in accident-mapping where an accident was assigned onto a road link whose "pattern" matches most closely with that of the accident-related data. We implemented the proposed approach using the 10,520 accidents that occurred in the year 2012 on the UK's strategic road network. The approach was able to map 100% of the accidents. A subset of these accidents (i.e. 560 accidents) was employed to evaluate the performance of the algorithm. It was found that the ANN-based accident mapping algorithm developed in this research outperforms other two commonly employed algorithms. More specifically, it was noticed that that the developed ANN algorithm produces a 14.7% more matches compared to the weight based approach and a 15.8% more matches than the closest link approach.

The research presented in this paper is significant as the developed algorithm has already being implemented by the UK Highways Agency in developing network-level risk mapping (from observed accident data) of the UK strategic road network. The developed algorithm is transferable and can be applied to other accident datasets. As part of future work, we intend to develop a single modular ANN model, integrating separate ANN models each address a unique network environment such as slip roads, junctions and roundabouts.

Acknowledgements

This paper is based on a project commissioned by the UK Highways Agency. The authors would like to express their special thanks to Stuart Lovatt and Louisa Cliffe (both from the Highways Agency) and Dr. Ramesh Perera and Chris Page (both from URS, the project partner) for their assistance in this research during the course of the project. The opinions in this paper are those of the authors and do not necessarily reflect those of the UK Highways Agency or URS. The authors remain solely responsible for any errors or omissions.

References

Aptel, I., Salmi, L.R., Masson, F., Bourde, A., Henrion, G., Erny, P., 1999. Road accident statistics: discrepancies between police and hospital data in a French island. Accident Analysis and Prevention 31, 101–108.

Austin, K., 1995. The identification of mistakes in road accident records. Part 1. Location variables. Accident Analysis and Prevention 27 (2), 261–276.

DoT, 2013, June. Reported Road Casualties in Great Britain: Main Results 2012. In: A Department of Transport Statistical Release Release27.

Dutta, A., Parker, S., Qin, X., Qiu, Z., Noyce, D.A, 2007. A system for digitizing Wisconsin accident location information. In: 86th Annual Meeting of the Transportation Research Board, Washington, DC, January, 2007.

Fellegi, I.P., Sunter, A.B., 1969. A theory for record linkage. Journal of the 2nd American Statistical Association 40, 1183–1210.

Greenfeld, J.S., 2002. Matching GPS observations to locations on a digital map. In: 81st Annual Meeting of the Transportation Research Board, Washington, D.C., January.

Hanspal, N.S., Allison, B.A., Deka, L., Das, D.B., 2013. Artificial neural network (ANN) modelling of dynamic effects on two-phase flow in homogeneous porous media. Journal of Hydroinformatics 15 (2), 540–554.

Jain, A.K., Mao, J., 1996. Artificial neural networks: a tutorial. Computer 29 (March (3)), 31–44.

Khan, M.A., Salim, A., Kathairi, A.I., Garib, A.I., 2004. A GIS based traffic accident data collection, referencing and analysis framework for Abu Dhabi. In: Proceedings Codatu XI, Bucharest.

Levine, N., Kim, K.E., Nitz, L.H., 1995. Spatial analysis of Honolulu motor vehicle crashes. I. Spatial patterns. Accident Analysis and Prevention 27 (5), 663–674.

Loo, B.P.Y., 2006. Validating accident locations for quantitative spatial analysis: a GIS-based approach. Accident Analysis and Prevention 38, 879–886.

Navarro, G., 2001. A guided tour to approximate string matching. Journal of ACM Computing Surveys 33 (1), 31–88.

Quddus, M.A., Ochieng, W.Y., Noland, R.B., 2007. Current map-matching algorithms for transport applications: state-of-the art and future research directions. Journal of Transportation Research Part C: Emerging Technologies 15 (5), 312–328.

Shinar, D., Treat, J.R., McDonald, S.T., 1983. The validity of police reported accident data. Accident Analysis and Prevention 15 (3), 175–191.

Tarko, A.P., Thomaz, J., Grant, D., 2009. Probabilistic determination of accident locations in a road network with imperfect data. In: 88th Annual Meeting of the Transportation Research Board, Washington, D.C., January 11–15, 2009.

Velaga, N.R., (Ph.D. thesis) 2010. Development of a weight-based topological map-matching algorithm and an integrity method for location-based. Loughborough University.

Velaga, N.R., Quddus, M.A., Bristow, A.L., 2009. Developing an enhanced weight based topological map-matching algorithm for intelligent transport systems. Journal of Transportation Research Part C: Emerging Technologies 17 (6), 672–683.

WHO, 2013. Global Status Report on Road Safety 2013. Supporting a Decade of Action.

Wilson, R.D., 2011. Beyond probabilities. Probabilities record linkage: using neural networks and complex features to improve genealogical record linkage. In: Proceedings of International Joint Conference on Neural Networks, San Jose, CA, USA, July 31–August 5.

Winter, M., Taylor, G.G., 2006. A modular neural network approach to improve map-matched GPS positioning. In: Proceedings of the 6th International Conference on Web and Wireless Geographical Information Systems.

Wu, J., Heydecker, G.G., 1998. Natural language understanding in road accident data analysis. Advances in Engineering Software 29 (7–9), 599–610.