International Journal of Engineering Technology, Management and Applied Sciences

IJETMAS

www.ijetmas.com December 2017, Volume 5, Issue 12, ISSN 2349-4476

Machine Learning Algorithm Performance in Incident Detection

Kostandina Veljanovska

Department of Intelligent Systems, Faculty of ICT,

University “St. KlimentOhridski” - Bitola

ABSTRACT

Machine learning as a part of artificial intelligence is becoming everyday tool in various aspects of life. Neural network as one of the most used techniques is elegant way of solving non-linear problemsand it is able to respond to variable real-time traffic conditions. In this research, the algorithm of neural network is compared to one conventional algorithm. The effectiveness of the algorithms were measured by several measures. The results are promising, proving the ability of neural network algorithm to perform incident detection on the freeway corridor.

Keywords

Artificial intelligence (AI), machine learning, intelligent system (IS), incident detection, neural network

INTRODUCTION

Implementation of machine learning techniques in traffic and transportation system in Republic of Macedonia, is at the starting point in terms of practical functioning. There are several theoretical attempts of research in laboratory regarding designing intelligent transportation system (ITS) and its implementation in traffic management centers. Incident management system as a part of traffic management has not been tackled at all in terms of ITS. This was our motivation for this research, to help provide scientific support in establishing a part of freeway incident management system.

The research is an attempt to prove that automatic incident detection algorithm as the key element of incident management system is feasible if it is applied as machine learning algorithm. This paper contains a performance comparison of conventional algorithm which uses McMaster methodology and one advanced algorithm in the field of artificial intelligence which uses Multi - Layer Feed - Forward Neural Network with Back Propagation methodology. There are many research in the field of intelligent transportation system and incident management system[1, 2, 3, 4]. Some researchers implement different types of neural networks [5, 6,1, 7] in order to help build safer roads. The ability of artificial intelligence to create simple algorithms in order to solve complex problems can be used inmodern freeway traffic monitoring system [7].

The power of the artificial intelligence techniques to learn and adapt itself to the changing environment was used in this research in order to start building automatic incident management system in the Republic of Macedonia.

AUTOMATIC INCIDENT MANAGEMENT SYSTEM

Incidents are a major cause of traffic delays on the freeways. Incident detection on time is crucial for saving life, lowering the damage done in the infrastructure and vehicles, lowering the impact on an environment, saving fuel, lowering pollution. Fast and reliable incident detection is clearly a vital traffic management objective. Automated incident detection systems are the basic elements of a freeway traffic management systems that rely on Automated Incident Detection Algorithms (AIDA). The aim of this paper was to develop automatic incident detection (AID) algorithms using Multi Layer Multi Feed Forward with Back Propagation Neural Network. The algorithm's efficiency in detecting incidents was compared to one conventional algorithm - McMaster Algorithm.

The model of an environment was provided with dataset as part of a large collection from different locations and time slots. The core components of the data were the occupancy and volume count data with reference to the time in thirty second intervals. The location where the incident was detected by the loop detectors was also noted.

Dataset was divided into two subsets. One subset of training data while the other subset consisted of validating data. Both incident detection algorithms were trained on the training data and the control components obtained were tested on the validating data which provided the outputs.

A. Modelling McMaster Incident Detection Algorithm

In order to model the McMaster Incident Detection Algorithm first the template was prepared, and then the program was developed in Java.The logic of the McMaster algorithm is that traffic downstream of a permanent bottleneck is different from that downstream of an incident – caused bottleneckor temporary bottleneck. Traffic operations are classified into one of four possible traffic states based on two variables: volume and occupancy. These variables were obtained from electronic loop detectors, located along the freeway. Occupancy is the percentage of time a detector is occupied by a vehicle during the reporting interval. This algorithm detects incidents in two distinct phases: detect the existence of traffic congestion, and determine the cause of congestion.

Congestion can be detected when values of occupancy and volume rise above established thresholds. The cause of congestion is then determined based on values from the adjacent downstream detector. Actually, the cause of congestion is defined to be a capacity reducing incident if the volume and occupancy values from the downstream detector are sufficiently low. Otherwise, the congestion is defined to be of the recurrent variety, which appears when traffic demand exceeds freeway capacity.

First step in theprocess of modelling the algorithm was to prepare a template according to whichit can classify data. The procedure for creating the algorithm is classical procedure of McMaster algorithm. The pick period was taken to be from 6:00 to 9:30 am. The maximum un-congested occupancy, (**OCMAX)** was found to be **0.2**. As expected,it was found that, volume – occupancy pairs for the un-congested period tend to cluster tightly about the LUD (lower bound un-congested data) line. The lower bound of the cluster can be established as:

$$\boldsymbol{Flow = a*(occupancy)^2+b*(occupancy)+c}$$

Parameters for the LUD line which best suits the template were adjusted by manual inspection and were found to be as follows:A = - 151; B = 146; C = - 1. Minimum discharge volume**(V_{crit})** downstream of a road during peak incident free hour was found to be **14**.

B. Modelling Neural Network Incident Detection Algorithm

Supervised neural networks require prior inputs and outputs in order to configure the system or force it to learn the pattern within the data for validating other data based on the same logic. For our model, we constructed a neural network with three layers. In the first layer it has four inputs i.e. volume and occupancy at upstream loop detector and volume and occupancy at the downstream loop detector. The second layer is one hidden layer of two neurons. The third layer is one output layer consisting of one output neuron indicating whether incident was detected or not.

In order to skip the "black boxes" with virtually no information with regards to the actual working of the network it was decided to use Excel to form our own customized version of a neural network. Final version of the network that was validated was trained in Excel, although a number of commercially available software were consulted and primary training of the network was done on them.

The complete network can be seen in Fig.1.

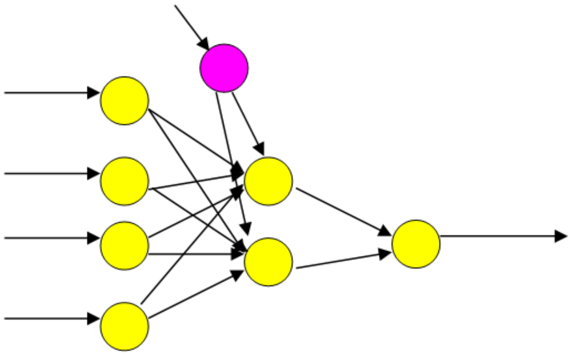

Fig.1 Neural Network created for the Algorithm

The input data is normalized to reduce the effect of local pull and after the outputs have been finalized, these outputs are again de-normalized (scaled back). Themodel of neural networkwas created as follows: four inputs were taken which included volume and occupancy values for both upstream and downstream loop detectors. These inputs were normalized using:

$$2*((x_i-x_{min})/(x_{max}-x_{min}))-1$$

There are many rules for setting the number of neurons in hidden layer: number of neurons in hidden layer need to be half of sum of input and output values; number of neurons in hidden layer need to be 2/3 of number of input values plus output values; number of neurons in hidden layer needs to be smaller than half of input value; number of neurons in hidden layer doesn't have to be too large, because of overcrowding, which reduced accuracy. The average of input and output neuron was taken to get a tentative value for the hidden layer's number of neurons. (in this case (4+1)/2=2.5 or take 2).

Afterwards these neurons were connected together and add a Bias neuron. Random weights were generated to initialize the system by putting these on all the links between inputs, bias and between hidden and output layers. Aninitial set of randomly generated weights were produced to start the iteration process. Weight randomness was between 0 and 1.

Thebias node input and bias node weight arexb and wb, respectively. Bias input was taken to be constantly as -1. A bias node was added to initialize a threshold function at 1.The next step was to calculate activation function values for the hidden layer neurons:

$$X_j=1/(1+\exp(-Y_j))$$

In order to perform back propagation, output value for activation function Y for hidden neurons was calculated:

$$Y_j=\sum wij * xi + wb * xb$$

Then Y_k was calculated: $Y_k=\sum xj * w_{jk} + x_b * w_b$and also compute $X_k=1/(1+\exp(-Y_k))$.

Change in weights or delta was calculated in final output value which shows either incident or non-incident. This is given by:

$$\partial k = xk(1-xk)*(T-xk)$$

where T is the given output or target value. Then ∂j was calculated for all the hidden nodes or neurons by using the formula:

$$\partial j = xj(1-xj)*(\partial k * xj)$$

Finally the change in the weights was calculated and also the adjusted weights were calculated:

$$wij=wij + \alpha*(\partial j * xi)$$

where α is the learning rate for the network. The learning rate was taken to be 0.3 so that a better descent could be achieved on the function surface.

A macro in Excel was developed to run the several iterations to change the weights and plug them into the next set of data to achieve learning. A subset of 4000 lines was used from this training data which was again scaled back and output in the form of incident and non-incident (close to 1 and 0) was compared with the validating dataset. This data was also normalized and final outputs were scaled back to original values. A cutoff value was taken at 0.85 to chart the outputs, based on observing the trend in data. Any value above 0.85 means that an incident was detected and anything lower means not. It was noticed that our learned data followed the same proximity and indeed locate incidents at most cases of data.

DISCUSSION OF THE RESULTS

For the model of McMaster algorithm the best performing set of parameters was: Vcrit=14, Ocrit=0.2, A=-151, B=146, C=-1.It was noticed that there was no big difference between Vcrit =14 and Ocrit=0.2 and Vcrit =14 and Ocrit=0.25. The results are shown in the Table 1.

Table 1.Results of performance McMaster Algorithm

Measurements	Percentage
Correct Classification	70.38%
Non Detected Normal Condition Data	94.12%
Detection Rate	22.09%

Performance of the algorithm with this program is shown in the Table 2. False Alarm Rate (FAR) is defined as number of detected incidents that were actually not incidents.

Table 2. Performance of the McMaster Algorithm

DR – Detection Rate	FAR – False Alarm Rate
12.27	1.65
12.35	2.16
15.41	3.01
16.74	3.79
20.9	5.17
22.09	5.88
26.72	8.64
27.43	8.97

Selection of the parameters is dependent on the goal of the detection. If higher detection rate was selected, FAR also goes up.

As for Back Propagation Neural Network algorithm it can be noticedthat this approach was quite different than that used by the McMaster algorithm. The formation of a template was not needed. The results are summarized in Table 3:

Table 3. Results of performance NN Algorithm

Measurements	Percentage
Correct Classification	66.84%
Non Detected Normal Condition Data	62.48%
Detection Rate	57.98%

As shown the results for NN seem to be on the lower side but comparing to other software, we get similar results. The commercially available software though are more powerful and professional in terms of design and output, but our system was developed in order to create a better learning tool and a good basis for incident detection in incident management system.

CONCLUSIONS

For the conventional automated incident detection algorithms as McMaster algorithm the most important is to have the data for the specific freeway prior to implementation on field. Results show that McMaster algorithm performs excellent false alarm rate and acceptable incident detection rate. NN algorithm on the other side, shows much higher incident detection rate and the data for the specific freeway are not necessary before implementation of the algorithm. In order to speed up the detection process it is possible to model the network prior to implementation.

REFERENCES

[1] L. Weiming, et al., Improved freeway incident detection using neural network based on pulse data of the loop detector, IEEE Xplore**,** 2004

[2] S. Bhonsle, et al., Database Centered Architecture for Traffic Incident Detection, Management and Analysis, 2000 IEEE Intelligent Transportation Systems Conference Proceedings, Dearborn, MI, USA, 2000

[3] K. Veljanovska, Intelligent Infrastructure Functional Aspects of the Traffic Incident Management System Support in the Republic of Macedonia, MSc paper, Technical Faculty Bitola, University St Clement Ohridski, Macedonia, 2002

[4] Q. WangTraffic incident detection based on Artificial Neural Network, Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on, 2011

[5] S. Srinivasan, et al., Freeway incident detection using hybrid fuzzy neural network, IET Intelligent Transport Systems, Volume 1, Issue 4, 2007, p. 249 –259

[6] Weerasuriya, S. A., "An application of artificial neural networks in freeway incident detection" (1998). Graduate Theses and Dissertations

[7] D. Srinivasan, et al.,Evaluation of adaptive neural network models for freeway incident detection, IEEE Transactions on Intelligent Transportation Systems (Volume: 5, Issue: 1, March 2004)