Data Aggregation for Road Functionality Detection Based on Machine Leaning and VANET

YiRen Shi[1*], Hao Wu[1]

[1] State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing, P. R. China
{12211092, hwu}@bjtu.edu.cn

Abstract. Vehicular Ad hoc Network (VANET) is superior to traditional Intelligent Transportation Systems (ITS) for its vehicle-to-x communication. VANET is established on the underlying information exchanges and broadcast by vehicles and infrastructures in the network. The driving status of vehicles on road can be got from the periodically broadcasts messages and the particular data can be stored to be exploited for special service. These obtained data are richer and more representative than those data collected by fixed sensors such as ground induction coil, floating car data or video analysis, which can be utilized to get the real-time road traffic knowledge in a way similar to big data technology. And the existing relevant approaches about traffic detection and prediction face the following disadvantages: (1) insufficient data utilized; (2) hand engineered in features; and (3) suboptimal learning method. As the big data about the real time traffic can be obtained through VANETs conveniently, this work thus proposed a road functionality detection model based on data aggregation in VANETs applying machine learning algorithm. The performance of the scheme is evaluated through extensive simulations. The simulation results show that the approach of fusing the information of vehicles on the road can accurately detect the dysfunctional road state and further locate on the specific malfunctioning position particularly for high dimensions.

Keywords: data aggregation, machine leaning, road functionality detection, VANETs

1 Introduction

Vehicular Ad hoc Network (VANET) can achieve more abundant services than the traditional intelligent transportation systems since vehicles in VANET can communicate with each other as well as with roadside base stations. The main contribution of network-based VANET is its ability of communication between vehicles in the mode of vehicle-to-vehicle (V2V), vehicles-to-roadside units (V2R) or other available infrastructures (V2I) by the support of underlying infrastructure communication platforms and protocols [8, 17]. The traditional method of getting traffic parameters of the recent time can only be relying on the technique of costly infrastructure construction, such as ground induction coil, floating car data or video analysis, some fixed facilities, etc. VANET can get more rich intuitive data of traffic situation with low computational load and easier maintenance compared to the traditional way. Especially the real time traffic station of their ongoing roads is practically needed for the vast number of drivers in the modern time. Even for some emergency situation in traditional transport system can only be manually input to corresponding system to realize information dissemination [2-3]. As the scenario shown in Fig. 1, when the road fails or is in a closure, the data (velocity, position, acceleration, etc.) of vehicles in the road will be affected immediately and continuously by the abnormal situation. And the data is broadcasted periodically in the safety messages of each vehicle, therefore these data can be aggregated to identify this situation and locate the fault location owing to VANET.
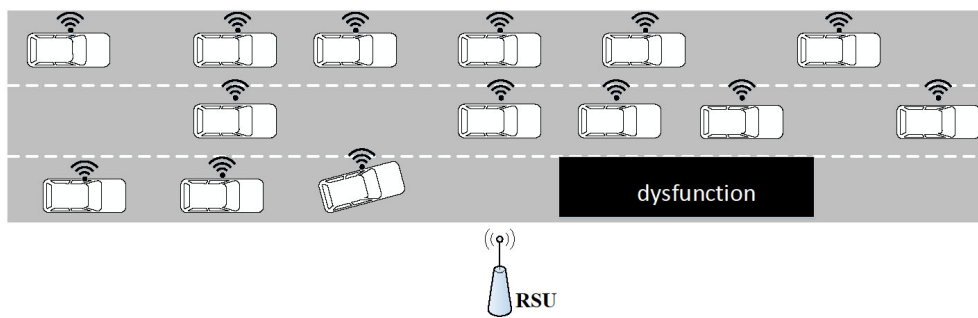
---

* Corresponding Author

Fig. 1. A road with dysfunctional part in VANET

The equipped On Board Unit (OBU) with each vehicle periodically broadcasts real-time messages about its identity, position, time, speed and other information reflecting the current traffic. Thus the related information of vehicles on the road can be got from the periodically broadcasts messages and the particular data can be stored to be exploited for special service. Since extra extracted information about the traffic state can be deployed from the massive data using machine learning rather than the hand engineered feature from the raw data. A corresponding survey of the current information collecting and data aggregation capabilities is presented in [8], it proposed an overall VANET framework with cognitive data and information fusion component, and an information processing unit in order to enhance the efficiency of the transportation system. Based on the mentioned framework, Golestan et al. proposed a scheme of integrating the sharing data with data fusion to obtain more accurate localization information of the vehicles [9]. An ensemble machine learning method is presented to classify misbehaviours in VANET [10]. A deep architecture with deep belief network (DBN) was employed in [14] to predict traffic flow, and the result showed that this method is promising in transportation research.

As the well rendered performance result of big data in the extensive application, the big data scenario formed by VANETs is going to be deployed to offer special services for the numerous drivers. In this paper, a road functionality detection approach is proposed based on data aggregation in VANETs applying machine learning algorithm. The massive data used is gathered by RSU composing the position, speed and acceleration and other related information of the vehicles on the road. Two algorithms, support vector machine (SVM) and deep neural network (DNN), are compared to detect the dysfunctional road state. And a satisfied result has been got while taking full advantage of the massive data, even an accuracy closer to 100%. As the better performance of deep learning in pattern recognition and machine learning [20], DNN is further utilized to determine the specific start and end position of the malfunctioning road. A mean accuracy MA closer to 0.9 is got which is practical useful to the traffic service in real life. Although this method may take a long time in the training process, but once the system is established, the corresponding states will be obtained almost simultaneously with the giving feature variables. We only need to update the database regularly to reconstruct the system parameters so that the system can be suitable for the latest road conditions and get more accurate results. Abundant experiments have been made in the scenario of road functionality detection, and the simulation results show the proposed approach is an efficient way to identify a fault and locate the fault location with good performance.

The rest of this paper is structured as follows. The background knowledge about VANET and machine learning is given in Section 2. Section 3 presents the proposed road functionality detection model and the detailed related machine learning algorithm used in the approach will be described here. Section 4 gives the experimental results of this approach and the analysis of these results. Section 5 draws the conclusion.

2 Background

2.1 Overview of VANET

Recent years, VANETs is widely accepted and studied as an emerging and promising technology in intelligent transportation systems and academia field due to its ability to improve the efficiency and safety of the traffic systems in our real life [18]. One good example is abnormal traffic situation detection. The VANETs have a very great application prospect in intelligent transportation system (ITS). The

explicit application scenario of VANETs in this paper is shown in Fig. 1 and Fig. 4. In VANETs, vehicle nodes on the roads and RSU constructed on the roadsides are the two members involved in the wireless communication process. Compared with the method used in [21] which collects the effective variables for urban traffic incident detection by the driver's smartphones as mobile sensors and help collect traffic data, the promising VANET system will be more sufficient and convenient for the drivers to make their vehicles join to the process of gathering the relevant data automatically.

The V2V protocol was devised to guaranteed the communication among the vehicle nodes in VANETs [24]. And the communication mode V2I is utilized in the communication process of vehicles and RSUs [4]. The vehicles are installed with some corresponding sensors to get information about the driving status, such as velocity, GPS (Global Positioning System), acceleration and the data in the Controller Area Network (CAN) of the vehicle, etc. Then the information mentioned will be gathered to RSU by V2V and V2I communication for aggregation. The integrated information in RSU can provide broad applications (real-time navigation, safe warning of intersections, discovery of free parking places, etc.) [11] to make driving safer, more efficient and comfortable.

Gathering data is a very important step which gathers massive real time traffic data for ITS applications. The sensed data from vehicles is always streamy and huge, so it's a key research point to decrease the communication bandwidth load of the data gathering. Many researches and implementation efforts have been done to make data gathering better. Back off based Per hop Forwarding (BPF), a broadcast based receiver oriented protocol is proposed in [26], it is available to gather real time data, like traffic conditions or road surrounding environmental parameters with a low communication overhead. Delot and Ilarri [6] outlined the development of VESPA and it's a system they were developing to make vehicles in the ad hoc network able to share the essential information (e.g., information on available parking spaces, road traffic accidents or obstructions, emergency braking, emergency vehicle enquiry rights, etc.).

2.2 Communication Overhead in VANETs

As the intuitive character of VANET that lacking centralized infrastructure and high mobility of the numerous vehicle nodes on the road, the communication in it will face the following technology difficulties: (1) communication collisions; (2) complex routing process; and (3) security and scalability of the whole system to ensure. Groups are so proposed in several researches revising the exchanged packets generated in the communication process among vehicles for the aim of less loads. The vehicles nodes must adopt different communication behaviour as the different vehicles communication states. So the group management and information processing within a group are the essential work to be done. A group in a VANET is composed of sets of location nearly vehicles and its formulation is affected by the mobility pattern of vehicles. A vehicle node called "leader of the group" is selected to manage the group. A secret key will be shared among the group members to work for the security of the wireless connection with the group leader [1]. Several clustering techniques for VANET have been proposed in the literature to form stable clusters.

The corresponding algorithms proposed to reduce the communication overhead in relevant work [7, 19, 25] almost form and maintain the clusters with two procedures: setup and maintenance procedure. The setup process in [7] is based on the following indicators: (1) Movement direction (MD), (2) Leadership duration (LD), the period of a node has become a leader since the last role changed, (3) The projection distance variation (PD) of all neighbouring nodes over time, which estimates the extent of its local topology changes. First, the MD is used to remove out moving away neighbouring nodes within transmission range to reduce numerous irrelevant nodes for a more stable cluster. And then a new node selects a cluster by the value of LD, PD and ID of the multiple neighbouring cluster heads to join in. Every vehicle node has a distinct identifier, ID. Any one of the following three events can trigger the maintenance procedure: (1) find a new node, (2) loss a neighbour, and (3) establish a neighbour.

Theoretical analysis have given in [7] that clustering technology is able to increase redundant messages dramatically in VANETs with an efficient hierarchical network structure and the overhead caused by groups is bound to a constant at each time step for each node, avoiding costly re-clustering chain response.

2.3   Machine Learning Algorithm

Machine learning involves extensive knowledge including pattern recognition and computational optimization theory in artificial intelligence. It dedicates to work up the study and construction of learning algorithms from the data and then make corresponding predictions on it. The algorithm model is formulated with a training set data composing input observations and output results. The data driven prediction is made by such algorithms rather through hand engineered feature selection program. It can be used to devise complex models and algorithms that lend themselves to prediction. These obtained learning models can help researchers, data scientists, engineers, and analysts to produce reliable, repeatable decisions and results and deploy hidden insights of the observations data through the process of learning from provided historical training data sets. Two of the most widely adopted machine learning methods are supervised learning and unsupervised learning. And the following part will introduce two kinds of widely used machine learning method in which SVM is supervised learning and DNN is semi-supervised learning algorithm.

Support Vector Machines (SVMs). An SVM classifier can be learned given a set of training data sets with input features data and corresponding belonged classification, and the SVM model build can distinguish a new observation from the features data into its largest possible classification. Therefore it's a non-probabilistic binary linear classifier. The SVM model has the input data feature data as points in space, and the mapped plot is appeared to be divided into two separate parts. The main duty of the formulation process is to figure out a clear gap between the two parts that is as wide as possible. New observations to be classified are then mapped into the plot with the gap and assigned to corresponding classification depending on the side of the gap the mapped points fall on. The SVM classifier tries to calculate a hyperplane in high dimension space which maximizes the margin. The ideal hyperplane for a SVM is got by training with samples of two classifications. The support vectors are the mapped points on the margin. Fig. 2 depicted a plot mentioned above, the circled spot is called support vector.
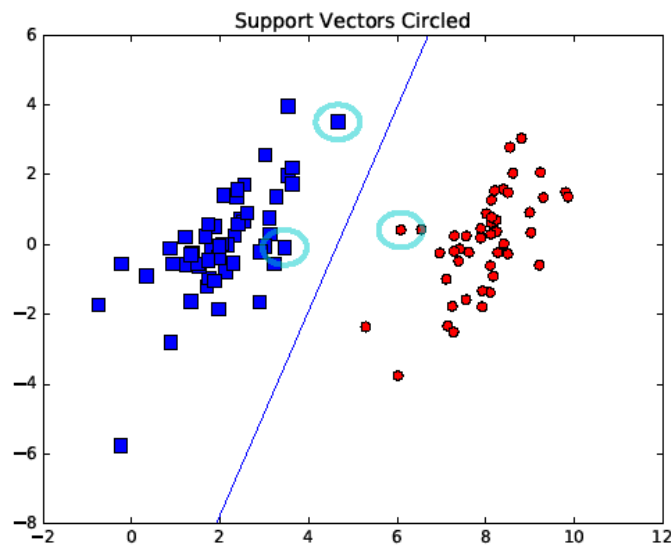

Fig. 2. A linear classifier learned with SVM

SVM is an advanced learning algorithm due to the outstanding study of statistical learning theory. SVMs perform a prominent result in image recognition, classification and a wide range of applications. There even will be a dedicated package of it in relevant deployment language system.

A context aware security framework for VANETs has been constructed in [23], and the SVM algorithm is utilized to discriminate the malicious nodes and the normal ones automatically. The proposed framework is more rested on context changes that are easy got in VANETs compared to traditional security solutions for VANETs. The context changes are caused by malicious nodes changing their attacking modes over time or changing the environmental factors (motion speed and transmission range, etc.) rapidly.

SVMs can also deliver a well performance in non-linear classification with the trick called kernel

function which implicitly maps the inputs into high dimensional feature spaces to guarantee the well done gap appeared. As the high dimensional feature of the data in the aggregation process in the detection work, the SVM algorithm has been applied in this paper. The details of computing the SVM classifier are depicted in Section 3.

Deep Neural Network (DNN). It needs two steps to construct a deep learning system, firstly, a large amounts of unlabelled data is necessary to unsupervised pre-train the multilayer NN (such as DBN) to get a feature presentation and, then, using the labelled data is used for supervised fine tuning the learned features to optimize the final system [14]. That's why DNN is called a semi-supervised machine learning algorithm. Recent work done by DL indicates that the layerwisely trained deep sigmoidal network is able to deliver well performance for many tasks such as the most widely applied in image and audio classification [13, 15].

In March 2016, Google DeepMind's computer program AlphaGo beats a 9-dan professional Go player Lee Sedol in a best-of-five competition without handicaps, a landmark program guided by "value network" and "policy network," both implemented with deep neural network technology. Monte Carlo tree search is used in AlphaGo's algorithm to find its moves based on previously learned knowledge by machine learning with an artificial neural network by extensive training including human and computer self-play. Silver et al. [22] describes the technical details of this new algorithm in computer Go which combines Monte Carlo tree search with deep neural networks (DNN) trained before by supervised learning, from historical human expert games and reinforcement learning from theirs self-play games. DNN is so applied in this paper to propose the high-dimensional data about the current traffic state. The results of extensive simulations show that the new method leads to with accuracy rates, particularly for high dimensions.

Hinton et al. [13] derived a fast and greedy algorithm that can learn deep directed belief networks layerwise, provided the top two layers form an undirected associative memory. The weights are initialized using a contrastive version of the wake-sleep algorithm in this algorithm to be better fine-tunned in the latter learning process. DBN is the most effective way among all deep learning models and also the most commonly used model. As depicted in Fig. 3, DBN is composed of stacked RBMs, each having one hidden and visible layer. In the stack, activations of learning units from the front RBM are transformed as input visible values for the next RBM. Hinton et al. proposed a fast greedy learning DBN method that learns one layer each time.
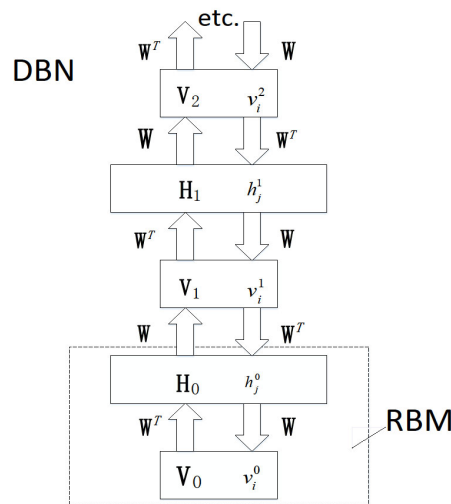

Fig. 3. Deep belief network construction

An RBM consists of two layers of units, a set of hidden units h and another visible units v. The matrix $\mathbf{W}$ is the connect weights between the two layers. The negative log probability v, h by energy function $E(v, h; \theta)$ and the correspondent conditional probability distributions are written as bellow, it's a binary RBM here:

$$-\log P(\mathbf{v},\mathbf{h}) \propto E(\mathbf{v},\mathbf{h};\boldsymbol{\theta})$$
$$= -\sum_{i=1}^{|\mathbf{V}|}\sum_{j=1}^{|\mathbf{H}|} w_{ij}v_i h_j - \sum_{i=1}^{|\mathbf{V}|} b_i v_i - \sum_{j=1}^{|\mathbf{H}|} a_j h_j \tag{1}$$

Where θ —a parameter set composed by w

The corresponding conditional probability distributions, when $\mathbf{v}$ or $\mathbf{h}$ is computed, as

$$p(h_j \mid \mathbf{v};\boldsymbol{\theta}) = sigm\left(\sum_{i=1}^{|\mathbf{V}|} w_{ij}v_i + a_j\right)$$
$$p(v_i \mid \mathbf{h};\boldsymbol{\theta}) = sigm\left(\sum_{j=1}^{|\mathbf{H}|} w_{ij}h_j + b_i\right) \tag{2}$$

Where $sigm(x) = \left(1/\left(1 + e^{-x}\right)\right)$. Then, the most likely probability of generating visible variables can be written as follow:

$$p(\mathbf{v}) = \sum_{\mathbf{h}} p(h \mid \boldsymbol{\theta}) p(v \mid h,\boldsymbol{\theta}) \tag{3}$$

When θ is learned from an RBM, $p(v \mid h,\boldsymbol{\theta})$ can be computed. The final step is to fine tuning the whole system from the top two layers with labelled data. This is the supervised part mentioned above. And then the DNN attained is completed to get into work for good prediction performance.

3   Proposed Data Aggregation Approach

A specific data aggregation approach has been proposed in this work based on VANET framework of information gathering and data fusion. In order to avoid data collisions, the group communication mode is used to loose the communication load. The communication between vehicles and RSUs can be completed by group, in which the leader vehicle stores all relevant data of the rest vehicles in this group and then forwards to the RSU. The utilization of group in VANET has been demonstrated in [1], and it is a good solution to decrease the number of communications in the network. The proposed road functionality detection model consists of two phases: (1) the collection of specific data and (2) the application of machine learning on the collected data.

3.1   Data Collection

As depicted in Fig. 4, groups are exploited to assist RSU to collect the necessary data for detection. Vehicles will be formed into groups according to dynamic cells where the leader has initiated the group or that has the greatest number of neighbours when the previous leader falls below an established threshold for group formation. The leader vehicle is responsible to collect the periodically broadcast safety messages of the group members and then assemble these data to a packet sent to RSU. Thus the number of communication links between vehicles and RSUs can be greatly decreased without missing any useful information. This is very significant to reduce the network overhead and RSU can collect more data of the road traffic.

Specific data (velocity/v, position/(x, y), acceleration/a, etc.) should be extracted from the received raw data. And then the extracted data will be sent to the machine learning module to complete pattern recognition in a time interval of t which can be set according to the traffic flow and the importance of the road. Since the abnormal situation needs to be recognized timely to minimize the impact on the whole traffic taking into account the communication and computational load. For example, t should be smaller if the road is regularly crowded and this road is very important for the whole transport system, vice versa. Moreover, the data should be rearranged and regularized in the formal format to be the input of recognition. If the dysfunctional road state is detected, the format data will be further turned to the malfunctioning position determining algorithm to predict the fault location.
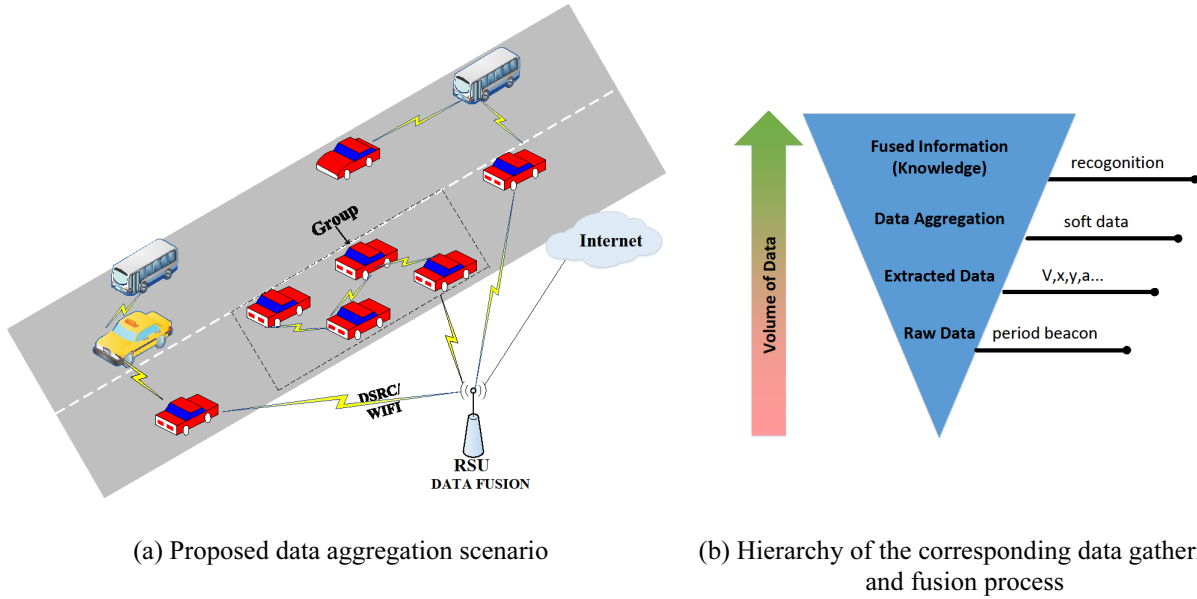
(a) Proposed data aggregation scenario   (b) Hierarchy of the corresponding data gathering and fusion process

Fig. 4.

The training process of machine leaning has exploited the intuitive relation of these data instead of the prior knowledge of specific domains for feature extraction which is necessary in traditional method. Thus tedious and error-prone hand-engineered features are not needed in this machine leaning approach.

3.2 Data Aggregation Process

The parameters of machine learning for detecting the dysfunctional road are obtained by the training data collected previously which is provided to the machine learning component for learning. Given the training data $\mathbf{T} = \{(\mathbf{X_1,Y_1}),(\mathbf{X_2,Y_2}),\dots(\mathbf{X_m,Y_m})\}$. m is the number of instance in training set, $\mathbf{X_i} = (v_1, x_1, y_1, a_1; v_2, x_2, y_2, a_2;\dots; v_n, x_n, y_n, a_n)$, n is the number of vehicles of one instance. The goal of the learning system is to form a classifier to detect the abnormal scenario and a regression function of the fault position. In the classifier system, $\mathbf{Y}=\{+1,-1\}$, $+1$ indicates the class of normal situation and -1 otherwise. The output of the regression system is $\mathbf{Y}=(s,e)$, $s,e \in [0,1]$ are the regularized start and end position of the malfunctioning road respectively. SVM and DNN are used to detect the dysfunctional road state, DNN is further utilized to determine the specific start and end position of the malfunctioning road in the simulation.

The SVM classifier tries to find a hyperplane in high dimension space which maximizes the margin: the minimum distance between the hyperplane and a data point of any class. Given the two classes, the principal object of SVM is to maximize the margin between the two classes. Haven the training data sets $(\mathbf{X_1, Y_1})$, …, $(\mathbf{X_m, Y_m})$, the objective is to find the hyperplane that hold a maximum margin, as follow ($\mathbf{W}$ is a weight vector and b is a constant threshold):

$$\mathbf{w} \cdot \mathbf{x} + b = 0 \tag{4}$$

The problem of finding the optimal hyperplane can be turned into a convex optimization problem:

$$\min_{w,b}\left\{\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\varepsilon_i\right\}$$
$$s.t. \quad y_i\left(\mathbf{w}\cdot\mathbf{x}_i + b\right) \geq 1-\varepsilon_i, \quad i=1,2,\cdots m \tag{5}$$
$$\varepsilon_i \geq 0, \quad i=1,2,\cdots m$$

Where ε_i is the relaxation factor on the learning vectors, and c is a constant responsible for complete the tradeoff between maximizing the margin and minimizing the number of misclassifications. By solving the convex optimization problem, the final decision function can be got as bellow:

$$f(\mathbf{x},b)=\{1\}=\text{sgn}(\mathbf{w}\cdot\mathbf{x}+b) \tag{6}$$

And the most commonly used measurement for classification is the Accuracy Rate (*AR*):

$$AR=100\%*\frac{number\ of\ correctly\ classified\ instances}{Total\ number\ of\ test\ instances}$$

It has been demonstrated in recent researches that the deep learning network can be trained layer wise with a good performance on hierarchical representations of high-dimensional data [5]. High-dimensional data can be converted to low-dimensional codes by training a multilayer neural network to reconstruct the high-dimensional input vectors, but the initial weights of the network are needed to be close to a good solution. And this can be done by the deep belief network (DBN) which is composed of stacked restricted Boltzmann machine (RBM), then gradient descent can be used for fine-tuning the weights [16]. There are two parts of DNN: the DBN is utilized to pre-train the multilayer NN with a good initial weights in an unsupervised way; and the weights will be fine-tuned by the labelled data to get a better prediction.

The mathematical communication process of the raw input data into final result is present in Fig. 5. An RBM consists of two layers of units, a set of hidden units *h* and another visible units *v*. DBN is stacks of RBMs. The outputs of the last RBM are taken as the inputs of next RBM. The training phase tries to fine tune **W** for good performance avoiding local optimization. The DBN is used to deal with the traffic abnormal status recognition problem (classification problem) as depicted in Fig. 5(a) and further utilized to locate the fault start (y_1) and end (y_2) position (regression problem) as depicted in Fig. 5(b).

(a) the traffic abnormal status recognition problem (classification)

(b) the fault start (y1) and end (y2) position prediction problem (regression)
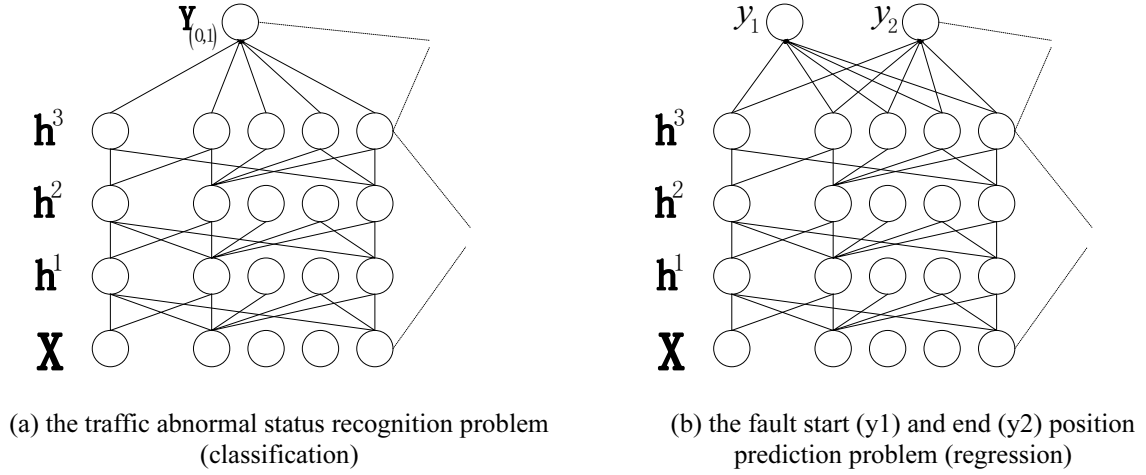
Fig. 5. DBN

Unlike the binary RBM, as introduced in Section 2.2, we replace it with real-valued units that have Gaussian noise to model the traffic flow data. Energy function and conditional probability distributions are given as:

$$-\log P(\mathbf{v},\mathbf{h})\propto E(\mathbf{v},\mathbf{h};\boldsymbol{\theta})$$
$$=-\sum_{i=1}^{|\mathbf{V}|}\frac{(b_i-v_i)^2}{2\sigma_i^2}-\sum_{j=1}^{|\mathbf{H}|}a_jh_j-\sum_{i=1}^{|\mathbf{V}|}\sum_{j=1}^{|\mathbf{H}|}\frac{v_i}{\sigma_i}h_jw_{ij} \tag{7}$$

$$p(h_j\,|\,\mathbf{v};\boldsymbol{\theta})=sigm\left(\sum_{i=1}^{|\mathbf{V}|}w_{ij}v_i+a_j\right)$$

$$p(v_i\,|\,\mathbf{h};\boldsymbol{\theta})=\text{N}\left(b_i+\sigma_i\sum_{j=1}^{|\mathbf{H}|}h_jw_{ij},\sigma_i^2\right) \tag{8}$$

Where σ —the standard deviation vector of Gaussian visible units

The parameter, w_{ij}, b_i and a_j are determined by the maximum likelihood parameter estimation algorithm so as to minimize the energy of states drawn from the data distribution and raise the energy of states that are improbable given the data. Eventually minimize the discrepancy between the original data and its reconstruction. Moreover, the sparsity of the hidden unit activations is usually added as a constraint condition to be the regularized term of the objective function. Fortunately, the contrastive divergence learning algorithm is proposed in [12] which makes an efficient approximation to the gradient of the log likelihood and the contrastive divergence update rule can be applied to the gradient descent during each iteration.

After the pre-training process for good initial weights, the following back propagation learning algorithm will be used to tune the weights for a better regression. Thus the whole detail parameter can be calculated and the formed DNN is potential to have a better performance since the features learned by the DBN are more representative than the raw input data.

4 Experiments and Results

4.1 Experimental Settings

The data in experiment was acquired through PTV VISSIM which is a microscopic multi-modal traffic flow simulation software. Both car following and lane change behaviour in VISSIM are completed with a psycho-physical car following model for longitudinal vehicle movement, and a rule-based algorithm for lateral movements based on Wiedemann. Thus the simulation results are very close to the actual scene. The simulation scenarios are set as Fig. 1 and the abnormal scenarios are constructed with different fault location to increase the reality of this simulation. Data of the intermediate simulation period is extracted in the experiment to increase its authenticity.

The different fault scenarios are simulated in order to fully illustrate the universality of this data fusion method, e.g. the simulated data are generated in different lanes, different fault locations and with different fault lengths. We set the distribution of speed and acceleration values in the high-speed road scenario, the speed value is between 60 and 120km/h. The different traffic scenarios were simulated under normal and abnormal conditions, respectively. Five different normal road conditions were set by changing the vehicle configuration in the road and changing the speed distribution according to the actual driving speed, and 30 kinds of road abnormalities were set with different fault location and fault length.

Taking into account the initialization and other specialities of the data during the establishment and termination of the system, we only extract the data in the stable phase of the simulation as the final collected data. Among them, more than 20,000 sets of information data concerning the position, velocity, and acceleration of vehicle nodes are collected in a time of 600 simulation seconds on a 1000 meters road. But in order to unify the data set specification, we only extract 20,000 of them as the raw data in the data processing phase. Next, according to actual needs, data from a specific dimension is randomly selected from these data as input data to the learning system. It is easy to understand that when the required data dimension is larger, the cycle of data collection in practical applications is smaller and the communication load is greater. For each scenario, only 20,000 arrays in stable time periods are selected as the data pool. This makes up 35 data pools, including data for 5 normal conditions and 30 different abnormal conditions. Randomly extract these data into standard input data with specific dimensions as the training and testing sets to construct learning system. In this paper, 15,000 and 20,000 standard input sample sets were extracted for normal and abnormal states, respectively, for subsequent machine learning algorithm research and analysis.

Input and output data: the input data is the regularized velocity/v, position/ (x, y), acceleration/a of n vehicles in the road. And different dimensions of the input feature space have been made for comparison considering the computational complexity and storage space limitations in practical application. The output of the classifier is $\{\pm1\}$ to detect the abnormal scenario and the output of the predict network is the regularized start/s and end/e position of the malfunctioning part.

Evaluation metrics: the measurement of the detection part is the accuracy rate mentioned above. The mean absolute accuracy (MA) is worked as the evaluation criteria of the prediction part.

$$MA\left(y, y^{'}\right) = 1 - \frac{1}{k}\sum_{i=1}^{k}\left|y - y^{'}\right|,\ k \text{ is the number of instances in the testing set. And the performance of}$$

DL for predicting position have been compared with different architecture.

4.2 Results and Analysis

Velocity and acceleration of vehicles in road are investigated as the feature vector to SVM classifier respectively. Fig. 6 shows the curve of accuracy rate on the testing set as a function of the dimension of the feature vector given different volume of training set. It can be seen from Fig. 6(a) that the recognition accuracy rate can easily reach 99% when the number of dimension is more than 200. It illustrates that this method still works well when the vehicles in road is rare in which the data collection work is difficult to achieve. And the over fitting phenomenon appears if the training set is too large. In addition, the performance under different training sets volume are compared to demonstrate that the SVM can still produce a good performance even if the training set is not quite enough. The Fig. 6(b) shows that the acceleration woks less effective than speed as feature input, the accuracy rate can only reach a maximum of about 84%. Therefore the velocity can be extracted from the received data by RSU as the feature vector to perform the detection program. The false positive rate is shown in Fig. 7 and it is substantially equal to false negative rate with a slightly superiority.
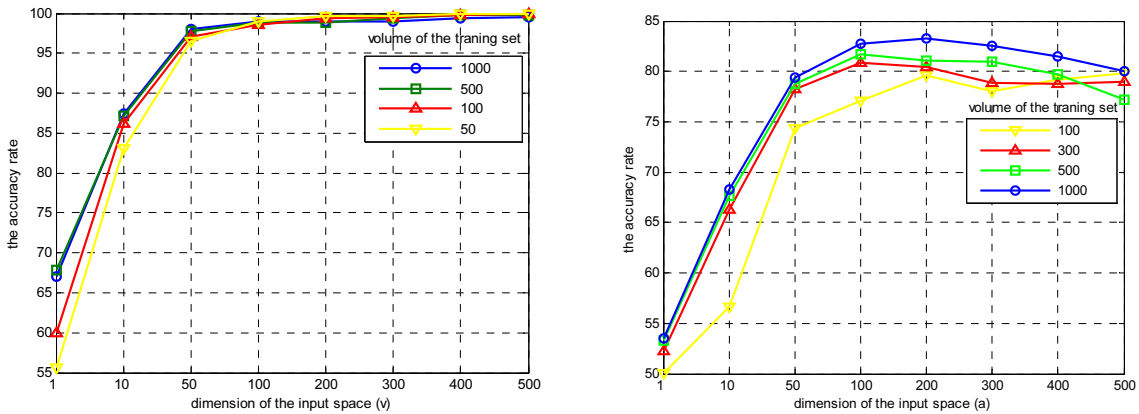

Fig. 6. SVM: the accuracy rate with different feature vector in input space
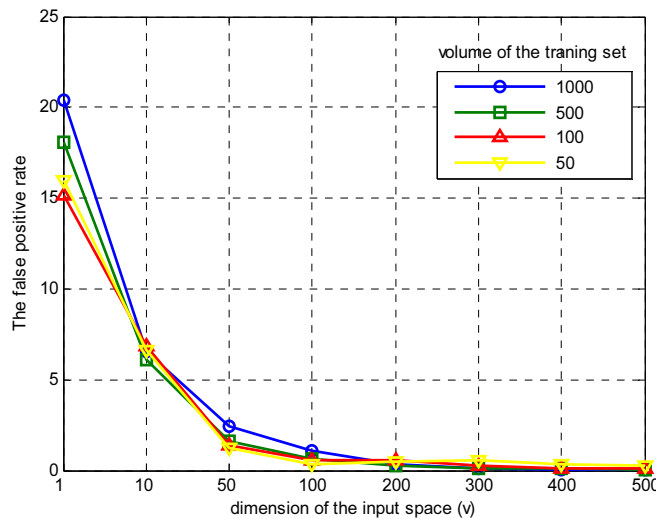

Fig. 7. The false positive rate

Then DNN is also used to recognize the abnormal situation. The simulation result is described in Table 1. It shows that the DNN can still realize a good performance with a very low input dimension. And at a high dimension condition, the demanded training set volume also increases accordingly to achieve better detection accuracy even 100%. Since given a higher dimension input, the number of parameters in the corresponding network increased, more training samples are necessary for tuning these parameters to obtain a more accurate result.

Table 1. DNN: the accuracy rate with velocity as the input feature vector

Dimension		10		30		100		300		500	
		Vol	*AR*	*Vol*	*AR*	*Vol*	*AR*	*Vol*	*AR*	*Vol*	*AR*
Volume of training set (vol)	Accuracy (AR)/%	40	81.1	40	85.1	100	99.1	800	70.2	4000	99.9
		80	82.2	500	92.6	200	99.2	1000	99.4	5000	99.9
		100	83.2	1000	96.1	1000	99.3	5000	99.8	8000	99.9
		10000	**87.1**	10000	**96.2**	10000	**99.3**	10000	**99.9**	10000	**99.9**
best/%		**87.1**		**96.2**		**99.3**		**99.9**		**99.9+**	

The predicted results by DNN with different architecture are given in Table 2. It can be seen that over fitting appears in the complicated architecture with 1000 dimension input. And the best *MA* is 0.8980 while the input feature vector is composed of all the relevant messages (velocity/*v*, position/*x, y*, acceleration/*a*). The correspondent DNN architecture contains two hidden layers with 1000 units in each layer. It can also learned from table 2 that the start position is potential to be predicted more accurately as the feature data of vehicles closer to the start position of the failed road are more presentative than the end one. The equivalent actual length of MA in this experiment is about tens meters which is very helpful to the corresponding traffic service and remedial measure. It can be learned from the prediction result that the better MA can be acquired with more relevant fused data under allowed conditions.

Table 2. DNN: MA through fusing different data under different architecture (the architecture 1000-1000 represents two hidden layers with 1000 units in each layer)

Architecture	1000/v+x		1500/v+x+y		2000/v+x+y+a	
	s	*e*	*s*	*e*	*s*	*e*
1000-1000(*s/e*)	0.8797	0.8851	**0.8847**	0.8831	**0.8980**	0.8889
800-800(*s/e*)	**0.8807**	0.8830	0.8834	0.8821	0.8911	0.8866
best/s	**0.8807**		**0.8847**		**0.8980**	

5 Conclusions

In this paper, a road functionality detection model has been designed to recognize the abnormal traffic state based on VANET and machine learning. The related data to be fused in this model are collected by RSU applying the group communication mode to avoid communication load in VANET. SVM and DNN are compared in the simulation to detect the dysfunctional road state and DNN is further utilized to predict the specific fault position. A deep learning architecture has shown its advantage in high input dimension. The evaluation results indicate that the current road traffic state can be well recognized through fusing various information about the vehicles in the road. Abundant experiments result also illustrate that the proposed approach will work well in actual application scenario through tuning the input dimension according to the traffic amount in the corresponding road. The more data are acquired to perform a better result. Therefore, the extra data such as the parameter of the CAN can be added in the future work to pursuing a better recognition and predicting result. And big data technology can be made greater integrated with traffic information in the future work, such as the massive information data of the vehicle in a certain period of time which reflects the driver's driving behaviour can be gathered by the insurance company. A corresponding traffic insurance policy can be made for every driver from learning these data to facilitate the traffic industry.

Acknowledgement

References

[1] C. Caballero-Gil, P. Caballero-Gil, J. Molina-Gil, Using groups to reduce communication overhead in vanets, in: Proc. the 2nd International Conference on Advances in P2P Systems, 2010.

[2] Q. Chai, C. Cheng, C. Liu, H. Chen, Vehicle queue length measurement based on a modified local variance and lbp, in: Proc. International Conference on Intelligent Computing, 2013.

[3] L. Liang, J.B. Kim, S. Jha, K. Sivanesan, G. Li, Spectrum and power allocation for vehicular communications with delayed CSI feedback, IEEE Wireless Communications Letters 6(4)(2017) 458-461.

[4] W. Cho, S.I. Kim, H.K. Choi, H.S. Oh, D.Y. Kwak, Performance evaluation of V2V/V2I communications: the effect of midamble insertion, in: Proc. Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology, 2009.

[5] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, A.Y. Ng, Large scale distributed deep networks, in: Proc. Advances in Neural Information Processing Systems, 2012.

[6] T. Delot, S. Ilarri, Data gathering in vehicular networks: the VESPA experience, in: Proc. 2011 IEEE 36th Conference on Local Computer Networks (LCN), 2011.

[7] P. Fan, P. Sistla, P. Nelson, Theoretical analysis of a directional stability-based clustering algorithm for VANETs, in: Proc. the fifth ACM international workshop on Vehicular Inter-networking, 2008.

[8] K. Golestan, A. Jundi, L. Nassar, F. Sattar, F. Karray, M. Kamel, S. Boumaiza, Vehicular ad-hoc networks (VANETs): capabilities, challenges in information gathering and data fusion, in: Proc. Autonomous and Intelligent Systems, 2012.

[9] K. Golestan, S. Seifzadeh, M. Kamel, F. Karray, F. Sattar, Vehicle localization in vanets using data fusion and v2v communication, in: Proc. the second ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications, 2012.

[10] J. Grover, V. Laxmi, M.S. Gaur, Misbehavior detection based on ensemble learning in vanet, in: Proc. International Conference on Advanced Computing, Networking and Security, 2011.

[11] H. Hartenstein, L.P. Laberteaux, A tutorial survey on vehicular ad hoc networks, IEEE Communications Magazine 46(6)(2008) 164-171.

[12] G.E. Hinton, Training products of experts by minimizing contrastive divergence, Neural Computation 14(8)(2002) 1771-1800.

[13] G.E. Hinton, S. Osindero, Y.W. Teh, A fast learning algorithm for deep belief nets, Neural Computation 18(7)(2006) 1527-1554.

[14] W. Huang, G. Song, H. Hong, K. Xie, Deep architecture for traffic flow prediction: deep belief networks with multitask learning, IEEE Transactions on Intelligent Transportation Systems 15(5)(2014) 2191-2201.

[15] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proc. Advances in Neural Information Processing Systems, 2012.

[16] H. Lee, C. Ekanadham, A.Y. Ng, 2008, Sparse deep belief net model for visual area V2, in: Proc. Advances in Neural Information Processing Systems, 2008.

[17] L. Nassar, A. Jundi, K. Golestan, F. Sattar, F. Karray, M. Kamel, S. Boumaiza, Vehicular ad-hoc networks (VANETs): capabilities, challenges in context-aware processing and communication gateway, in: Proc. Autonomous and Intelligent Systems, 2012.

[18] A. Rangari, P.B. Girish Agrawal, Data aggregation control for VANET-a reviews, International Journal of Engineering Research and Technology 2(2013) 556-569.

[19] Z.Y. Rawashdeh, S.M. Mahmud, A novel algorithm to form stable clusters in vehicular ad hoc networks on highways, EURASIP Journal on Wireless Communications and Networking 2012(1)(2012) 1-13.

[20] J. Schmidhuber, Deep learning in neural networks: An overview, Neural networks 61(2015) 85-117.

[21] C. Siripanpornchana, S. Panichpapiboon, P. Chaovalit, Effective variables for urban traffic incident detection, in: Proc. Vehicular Networking Conference (VNC), 2015.

[22] D. Silver, A. Huang, C.J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, S. Dieleman, Mastering the game of go with deep neural networks and tree search, Nature 529(7587)(2016) 484-489.

[23] O.A. Wahab, A. Mourad, H. Otrok, J. Bentahar, CEAP: SVM-based intelligent detection model for clustered vehicular ad hoc networks, Expert Systems with Applications 50(2016) 40-54.

[24] C.X. Wang, X. Cheng, D.I. Laurenson, Vehicle-to-vehicle channel modeling and measurements: recent advances and future challenges, IEEE Communications Magazine 47(11)(2009) 96-103.

[25] P. Yang, J. Wang, Y. Zhang, Z. Tang, S. Song, Clustering algorithm in VANETs: a survey, in: Proc. 2015 IEEE 9th International Conference on Anti-counterfeiting, Security, and Identification (ASID), 2015.

[26] M.N. Zarmehri, A. Aguiar, Data gathering for sensing applications in vehicular networks (poster), in: Proc. Vehicular Networking Conference (VNC), 2011.