Received December 7, 2020, accepted December 21, 2020, date of publication December 24, 2020, date of current version January 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3047340

A Hybrid Method for Traffic Incident Detection Using Random Forest-Recursive Feature Elimination and Long Short-Term Memory Network With Bayesian Optimization Algorithm

QIANG SHANG[1], LINLIN FENG[2], AND SONG GAO[1]
[1]School of Transportation and Vehicle Engineering, Shandong University of Technology, Zibo 255000, China
[2]School of Marxism, Shandong University of Technology, Zibo 255000, China

Corresponding author: Song Gao (gs6510@163.com)

This work was supported in part by the Shandong Provincial Natural Science Foundation under Grant ZR2018BF024, in part by the Key Research and Development Program of Shandong Province through the Soft Science Project under Grant 2020RKB01793, and in part by the Ministry of Education (MOE) in China Project of Humanities and Social Sciences under Grant 18YJC190003.

ABSTRACT Automatic Incident Detection (AID) is an important part of Intelligent Transportation Systems (ITS). A hybrid AID method using Random Forest-Recursive Feature Elimination (RF-RFE) algorithm and Long-Short Term Memory (LSTM) network optimized by Bayesian Optimization Algorithm (BOA) is proposed in this article. Firstly, a relatively comprehensive set of initial variables is constructed using basic traffic variables and their combinations. Secondly, feature variables are selected from the initial variables using the RF-RFE algorithm. Then, the feature variables are used for training the LSTM network, and the hyper-parameters of the LSTM network are optimized by BOA. In addition, Synthetic Minority Over-Sampling Technique (SMOTE) is employed to solve the problem of imbalance between incident sample size and non-incident sample size. Finally, experiments are conducted using real-world data to test performance of the proposed method and compare with several state-of-the-art AID methods on multiple evaluation criteria. The experimental results illustrate that the proposed method achieved superior performance with respect to almost all the evaluation criteria. It also shows that the proposed method is promising for dealing with the problems of imbalance and small sample size of traffic incident data.

INDEX TERMS Traffic incident detection, feature selection, long short-term memory network, Bayesian optimization algorithm.

I. INTRODUCTION

Traffic incidents such as accidents, vehicle breakdowns and spilled loads may lead to severe problems such as congestion, traffic delays, increased emissions, and secondary accidents. In the United States, more than half of the congestion on freeways is caused by incidents [1]. Early detection of traffic incidents can largely minimize traffic delays, wasted fuel, emissions, and economic losses, and also reduce the likelihood of secondary accidents [2]. Thus, AID is a crucial technology in ITS and has attracted the interest of many researchers, particularly in terms of reducing congestion on freeways. In related work, traffic incidents usually refer to any non-recurring events that may disrupt normal traffic flow and reduce the traffic capacity of a road [3]. Planned traffic incidents, such as routine maintenance of roads and the traffic control for sports games and concerts, are not in the scope of AID research.

In the past few decades, many AID methods have been proposed. Data used for AID are usually collected from three sources: static detectors, dynamic detectors, and traffic cameras. Although traffic cameras belong to static detectors, they are listed separately because of their unique characteristics. Static detectors mainly include loop detectors,

The associate editor coordinating the review of this manuscript and approving it for publication was Li He.

radar detectors, infrared detectors and magnetic detectors, which can provide numerical traffic variables, such as volume, speed, and occupancy. Dynamic detectors (such as GPS sensors) are usually installed on probe vehicles to provide a continuous stream of measurements such as instantaneous speed, latitude and longitude of the probe vehicles when they pass by the roads. Due to the limitations of probe vehicles proportion, coverage, sampling frequency and accuracy, in other words, there are high requirements for the quality of probe vehicles data [4]–[6], the data of dynamic detectors used for AID often requires additional data as a supplement [2]. Video cameras are usually installed to monitor the traffic state of the roads and provide video clips of the roads [7]. Videos collected from cameras can be either converted into numerical variables or directly used for detecting incidents with the help of computer vision techniques [8], [9]. Although video can provide more information, they are susceptible to the weather and cost a lot to cover entire roads. In this study, we focus on the use of numerical traffic variables rather than videos for AID.

The previous AID methods mainly include California algorithm [10], Standard Normal Deviation (SND) algorithm [11], McMaster algorithm based on catastrophe theory [12], Minnesota algorithm [13] and Double Exponential Smoothing (DES) algorithm [14]. The California algorithm constructs three variables using upstream occupancy and downstream occupancy and its combination. If the three variables exceed the preset threshold, it is determined that the traffic incident occurs on the road between the two detectors used to collect traffic data. The SND algorithm uses the observed and predicted values of traffic variables to give a formula for calculating the standard deviation. If the two standard deviations exceed the preset threshold, it is determined that the traffic incident occurs. Both McMaster algorithm and low-pass filtering algorithm use a certain theory to analyze the degree of change in traffic data to determine whether a traffic incident occurs. For DES algorithm, traffic variable is predicted using the double exponential smoothing method. If the difference between the actual value and the predicted value exceeds the preset threshold, it is determined that there is a traffic incident. For these previous methods, it is important to determine the threshold used to detect incident. Threshold determination still attracts attention. Recently, an AID method using spatiotemporally denoised robust thresholds has been proposed, and results show that these robust thresholds can improve incident detection performance significantly compared to traditional threshold determination [15].

Since the 1990s, there have been some AID methods based on machine learning, such as Artificial Neural Network (ANN) [16]–[18], Support Vector Machine (SVM) [19]–[23], decision trees [24] and Bayesian classifiers [25]. For these AID methods, traffic incident detection is regarded as a pattern classification problem, more precisely, a binary-classication problem (incident or non-incident). The machine learning-based AID methods automatically generate judgment rules by learning traffic data collected from upstream and downstream of the incident location, which can more effectively mine the implicit information of traffic data. Therefore, machine learning-based AID methods are more promising than the traditional AID methods.

As an important branch of machine learning, deep learning has been rapidly developed in recent years and has been successfully applied to supervised/unsupervised classification and anomaly detection problems. Against this background, the state-of-the-art technologies of deep learning are gradually being used for AID. Because of the excellent performance of deep learning in computer vision, the AID method based on deep learning usually uses video data [26], [27]. Based on spatio-temporal traffic data, Fuzzy Deep Learning (FLD) [28], Deep Extreme Learning Machine (DELM) [29], Convolutional Neural Network (CNN) [30], and Rrestricted Boltzmann Machines (RBM) [31] have been used for traffic incident detection or duration prediction. Experimental results show that the performance of AID methods based on deep learning is encouraging.

In recent years, hybrid methods based on two or more methods/models have gradually attracted widespread attention. It is worth noting that most of these hybrid methods use machine learning. Wang *et al.* [32] proposed a hybrid AID method based on time series analysis and SVM. Time series analysis is used to predict the traffic variables in the normal state, and the predicted traffic variables and the measured traffic variables and their differences are used as the input of SVM. Agarwal *et al.* [33] built a hybrid model using logistic regression with a wavelet-based feature extraction for detecting traffic incidents. Xiao [34] proposed an AID method using SVM and k-Nearest Neighbor (KNN) ensemble learning, and the experimental results indicate that the proposed method obtains the best performance with better robustness than the individual model (SVM or KNN). Lin *et al.* [35] proposed an AID method based on Generative Adversarial Network (GAN) and SVM. The experimental results show that this method is effective in dealing with imbalanced and small training samples.

From the literature review, we can see that although the performance of the previous AID methods is often unsatisfactory, they provide an important reference for determining the data input form and incident judgment criteria of later AID methods. In recent years, machine learning-based AID methods have received widespread attention because of their outstanding advantages, and a large number of machine learning-based AID methods and their combination with other algorithms have been proposed. At present, research on the AID method mainly focuses on the use of new algorithms and the combination, ensemble and optimization of existing algorithms. However, the construction of AID variable sets and the selection of feature variables are often not considered. That is to say, the machine learning-based AID algorithm usually takes the original traffic variables (measured traffic volume, speed and occupancy) as input variables. As we all know, in terms of classification problems using machine learning, feature selection can retain feature variables and remove redundant variables, thus reducing computational complexity and improving classification accuracy effectively [36].

Therefore, a novel AID method is proposed using RF-RFE algorithm and BOA-optimized LSTM network in this study. LSTM network is chosen as the basic model, because of its excellent performance to capture the long-term dependence of time series and has been successfully applied to traffic variables prediction with an excellent performance [37]–[39]. We have reason to believe that LSTM network has the potential to improve the effect of incident detection based on traffic variables. The main contributions of this article are summarized as follows.

(1) A relatively comprehensive set of initial variables is constructed using basic traffic variables and their combinations, and then feature variables were selected from initial variables based on the RF-RFE algorithm.

(2) To the best of our knowledge, this study is the first to use LSTM network for traffic incident detection, and BOA is used to optimize the hyper-parameters of LSTM network.

(3) The SMOTE method is employed to solve the problem of imbalance between incident sample size and non-incident sample size, which can generate new incident sample rather than simply repeat sampling.

(4) In terms of AID performance evaluation, not only binary-classication performance evaluation criteria such as the ROC curve, confusion matrix, accuracy and precision are used, but also Mean Time to Detection (MTTD), a criterion commonly used to evaluate AID performance.

To give an explanation of the proposed AID method in detail, the rest of this article is organized as follows. Section 2 illustrates the construction of the initial variables set, the balance of the data set based on SMOTE, the selection of feature variables based on RF-RFE algorithm, LSTM network, BOA and the whole process of the proposed method. Section 3 presents the experimental result and discussion. Finally, the conclusions and future work are described in Section 4.

II. METHODOLOGY

A. CONSTRUCTION OF THE INITIAL VARIABLES SET

The significant change of traffic variables caused by a traffic incident is the basis for designing AID algorithms. In order to clearly show the changing trends of traffic variables before and after traffic incidents, traffic variables of upstream and downstream of the incident site are presented in Figure 1 and Figure 2 respectively (the data were collected from the east line of Shanghai North-South Elevated Expressway on September 22, 2008). In normal traffic state conditions, the change of traffic variables (traffic volume, traffic speed and occupancy) is relatively stable. When a traffic incident occurs, the traffic volume and traffic speed from upstream of the incident site decrease rapidly, and the occupancy increases rapidly; then the traffic volume and occupancy from downstream decrease, and the traffic speed from downstream

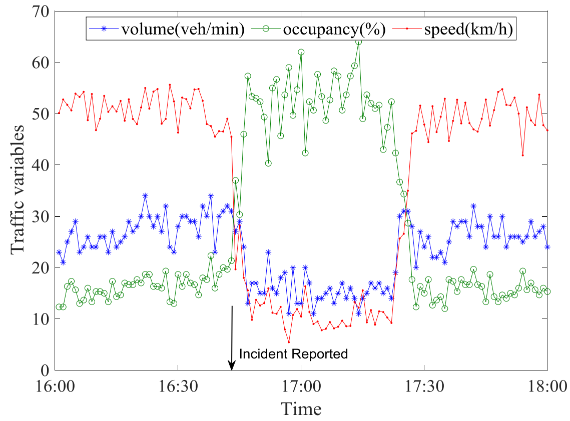

FIGURE 1. Traffic variables of upstream detector before and after the incident.

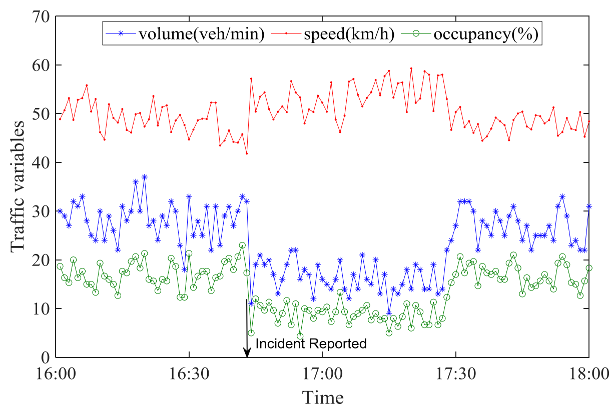

FIGURE 2. Traffic variables of downstream detector before and after the incident.

increases. When the traffic incidents are over, the traffic variables return to a relatively stable state.

Traffic variables in normal traffic state can be used as a criterion for incident detection, so it is necessary to predict traffic variables in the near future for comparing the predicted values with the measured values, such as the SND algorithm. In addition, the combination of traffic variables from upstream and downstream of the incident site has also changed significantly. For example, the California algorithm uses the difference between the upstream occupancy and downstream occupancy as a criterion of the incident.

In view of the above analysis, a comprehensive set of initial variables for AID is constructed using the measured values, predicted values and their combinations of upstream and downstream traffic variables. The initial variables set consists of five parts: first, the measured values of upstream traffic variables; second, the measured values of downstream traffic variables; third, the difference between the measured values and predicted values of upstream traffic variables; fourth, the difference between the measured values and predicted values of downstream traffic variables; fifth, the difference between the measured values of upstream traffic variables and the measured values of downstream traffic variables. The initial

TABLE 1. Initial variables set for traffic incident detection.

No.	Variables Notation	Descriptions
1	*v_up_m*	the measured value of upstream traffic volume
2	*s_up_m*	the measured value of upstream traffic speed
3	*o_up_m*	the measured value of upstream occupancy
4	*v_down_m*	the measured value of downstream traffic volume
5	*s_down_m*	the measured value of downstream traffic speed
6	*o_down_m*	the measured value of downstream occupancy
7	*v_up_m−v_up_p*	the difference between the measured value and the predicted value of the upstream traffic volume
8	*s_up_m−s_up_p*	the difference between the measured value and the predicted value of the upstream traffic speed
9	*o_up_m−o_up_p*	the difference between the measured value and the predicted value of the upstream occupancy
10	*v_down_m−v_down_p*	the difference between the measured value and the predicted value of the downstream traffic volume
11	*s_down_m−s_down_p*	the difference between the measured value and the predicted value of the downstream traffic speed
12	*o_down_m−o_down_p*	the difference between the measured value and the predicted value of the downstream occupancy
13	*v_up_m−v_down_m*	the difference between the measured values of upstream and downstream traffic volume
14	*s_up_m−s_down_m*	the difference between the measured values of upstream and downstream traffic speed
15	*o_up_m−o_down_m*	the difference between the measured values of upstream and downstream occupancy

variables set constructed is given in Table 1. Here, the moving average method is used to obtain the predicted values of traffic variables, and the first four adjacent values are used to predict the fifth values.

B. PROCESSING IMBALANCED DATA USING SMOTE

It is worth noting that the number of traffic incident samples is much smaller than the number of non-incident samples, because it is difficult to obtain the same number of incident samples as non-incident samples. Therefore, traffic incident detection can be seen as a binary-classication problem using imbalanced data. If the imbalanced data is used for training directly, the misclassification rate of minority samples will be much higher than that of majority samples. As long as the majority of samples are classified correctly, we can get a higher classification accuracy. However, the classification accuracy of minority categories is more important, which directly determines the result of traffic incident detection.

At present, there are two main methods to deal with the problem of imbalanced data classification, one is to reconstruct the data set by resampling (including undersampling of majority samples and oversampling of minority samples), so as to make it balanced; the other is to improve the model or algorithm without changing the original sample data set, so as to make it applicable to the imbalanced data set. Compared with the two methods, the first method is more convenient and widely used. As a widely used over sampling technology, the SMOTE finds the k minority samples closest to each minority sample based on the k-nearest neighbor method, and then randomly linearly interpolates on the connecting line of the minority sample and the k nearest neighbor minority samples to generate k new samples that are added to minority samples [40]. SMOTE is able to generate new samples that do not exist in the original samples; thus, it can avoid overfitting to some extent. SMOTE is used to balance traffic variable samples in this study, and the specific steps of SMOTE are as follows:

Step 1: For each sample x_i in the incident samples set, use the Euclidean distance as a metric to search for the k samples closest to the sample in the incident samples set.

Step 2: Determine the sampling rate N according to the ratio of the number of non-incident samples to the number of incident samples, and randomly select N samples from the k nearest neighbor samples of each incident sample x_i, denoted as x_{ij}.

Step 3: According to Equation (1), new incident samples are generated by random linear interpolation between the nearest neighbor sample x_{ij} and incident sample x_i.

$$x^{new} = x_i + rand\,(0, 1) \times \left(x_{ij} - x_i\right) \tag{1}$$

where, $rand$ (0, 1) represents a random number belonging to the interval [0, 1].

Step 4: By combining the newly generated incident samples with the original samples, a relatively balanced samples set will be obtained.

C. FEATURE VARIABLES SELECTION USING RF-RFE

This section describes the random forest and its application in variable importance analysis at first, and then illustrates the RF-RFE algorithm for feature variables selection. In this way, not only the dimension of the input data can be reduced, but also the classification accuracy is likely to be improved.

1) RANDOM FOREST AND IMPORTANCE ANALYSIS OF VARIABLES

Random Forest (RF) is an ensemble learning algorithm using decision trees [41], which can be used not only for classification and regression, but also to measure Variable Importance (VI). There are several methods to measure the importance of variables based on random forest, among which the method based on classification accuracy of the "Out of Bag" (OOB) data is the most commonly used one. The basic idea of this method is to add random noise to each variable of the OOB data in turn, and the importance of the variable is determined according to average value of the decrease of classification accuracy of the OOB data of all decision trees. The main steps of the method are as follows.

Step 1: Establish a random forest model.

(1.1) Bootstrap sampling technique is used to extract K' samples randomly from the K original training samples. A decision tree $h_k(k = 1, 2, \cdots, p)$ is constructed using the extracted sample data, and the unselected samples are out-of-bag data K_k^{oob}.

(1.2) Extract the $m_{try}(m_{try} < 15)$ variables from the initial variables set at each node of the decision tree, calculate the amount of information contained in each variable, and select a variable with the best classification ability from the m_{try} variables for node splitting.

(1.3) Each tree splits to its maximum size without pruning throughout the growth of the forest.

(1.4) The above steps are repeated p times to generate a random forest $f = \{s_1, s_2, \cdots, s_p\}$ with p decision trees.

Step 2: For each decision tree in the random forest, OOB data L_k^{oob} is used to calculate the classification accuracy R_k.

Step 3: Each initial variable in the training set is recorded as $\lambda_l(l = 1, 2, \cdots, 15)$, and random noise is added to λ_l of the OOB data L_k^{oob} to obtain a new OOB data $\hat{L}_k^{oob}$, and the classification accuracy $\hat{R}_k$ of each decision tree s_k using the new OOB data $\hat{L}_k^{oob}$ is calculated.

Step 4: The importance of the variable λ_l can be calculated according to the Equation (2).

$$VI = \frac{1}{p}\sum_{i=1}^{p}\left(R_k - \hat{R}_k\right) \tag{2}$$

2) RF-RFE ALGORITHM

Recursive Feature Elimination (RFE) is a variable selection method based on feature variable ranking [42]. The RF-RFE algorithm uses random forests to analyze the importance of variables and rank them according to the importance of variables, and then select important variables by RFE. Flowchart of RF-RFE algorithm is illustrated in Figure 3. First, for the training set of initial variables, random forest is used to calculate the importance of the variables and rank them; then delete the least important variables each time, and use the remaining variables to retest the classification accuracy using random forest. In this way, the classification accuracy is calculated iteratively until all feature variable sequences are searched. Finally, the ranking of all variables and the classification accuracy of random forest with different variables are obtained.

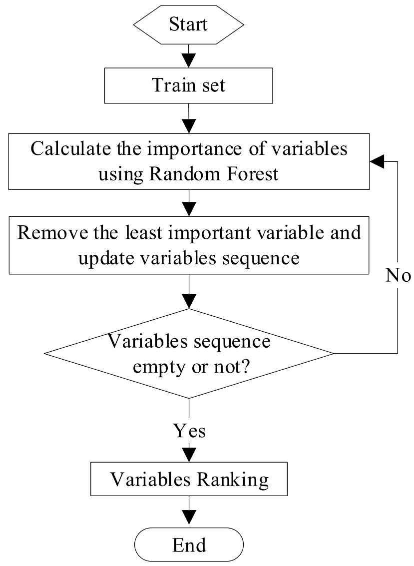

FIGURE 3. Flowchart of RF-RFE algorithm.

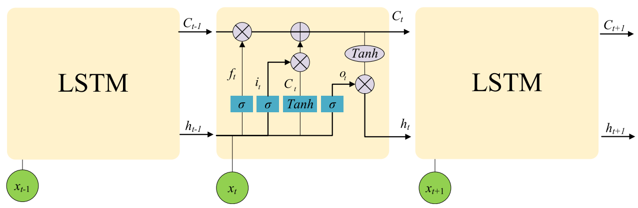

FIGURE 4. Structure of LSTM unit.

D. LSTM NETWORK

LSTM network is a kind of recurrent neural network with LSTM cells as building blocks for its hidden layers and has been proven robust for capturing long-term dependencies [43]. As a deep learning algorithm, the LSTM network can have the time-varying inputs and targets. Due to the excellent ability of solving long-term dependency problem, the LSTM network often has satisfactory performance in processing time series. As the core of LSTM network, the LSTM unit consists of a memory cell that stores updates information by three multiplicative gates (an input gate, a forget gate and an output gate), which helps overcome gradient explosion and vanish problem [39], [43]. LSTM unit structure is shown in Figure 4. Each LSTM unit updates its cell state according to the activation of each gate. The inputs provided to the LSTM are fed into operations that control the input gate i_t, output gate o_t, and forget gates f_t, which are managed in cell memory. $\tilde{C}_t$ is the initial cell state, C_t is the updated cell state and h_t is the hidden value updated at every time step t.

The mathematical expression of LSTM can be written as follows. The input gate receives the output h_{t-1} from the previous time step and the input x_t, and then outputs two values:

$$i_t = \sigma\left(W_i\left[h_{t-1}, x_t\right] + b_i\right) \quad (3)$$

$$\tilde{C}_t = tanh\left(W_C\left[h_{t-1}, x_t\right] + b_C\right) \quad (4)$$

The forget gate receives the same input and its output is:

$$f_t = \sigma\left(W_f\left[h_{t-1}, x_t\right] + b_f\right) \quad (5)$$

Then, the cell state is updated as Equation (6):

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (6)$$

The output of the output gate is:

$$o_t = \sigma\left(Wo\left[h_{t-1}, x_t\right] + b_o\right) \quad (7)$$

Finally, the output of a LSTM unit is:

$$h_t = o_t * tanh\left(C_t\right) \quad (8)$$

In the above equation, weight matrix W and bias vector b are learnable parameters in each gate, respectively; $\sigma(x)$ is the sigmoid function; and $*$ represents dot production. The input gate determines how much input information should be kept, the forget gate determines how much previous information should be forgotten, and the output gate determines how much information should be output to the next state.

For multi-layer LSTM, we just feed the output of the lower layer as the input of the higher layer and stack them repeatedly. For our classification task, we only take the outputs of all hidden units in the last layer, and then link them to fully connected layer to implement binary classification.

E. BAYESIAN OPTIMIZATION ALGORITHM (BOA)

BOA is one of the most famous estimation of distribution algorithms, which combines Bayesian network with evolutionary algorithm to solve nearly decomposable problems. In BOA, global statistical information is extracted from the best solution of current search, and Bayesian network is used to model it. Bayesian optimization mainly consists of two phases. In the first phase, a model is updated in each iteration that characterizes the "best guess" at the objective function vs. design parameter and quantifies the uncertainty in the guess. This phase of Bayesian optimization is called the learning phase. In the second phase, an acquisition function is selected that guides the optimization by determining the next point to be evaluated. Choosing the next altitude to maximize acquisition function is called the optimization phase. Compared with the traditional search algorithm, BOA has faster search speed and fewer iterations, so it has advantages in optimizing the hyper-parameters of machine learning algorithm [44]–[46]. More details about BOA for hyper-parameters of machine learning can be found in [44].

In this study, the BOA is employed to optimize the time step and the number of hidden layer nodes of LSTM, in order to achieve a better performance for traffic incident detection.

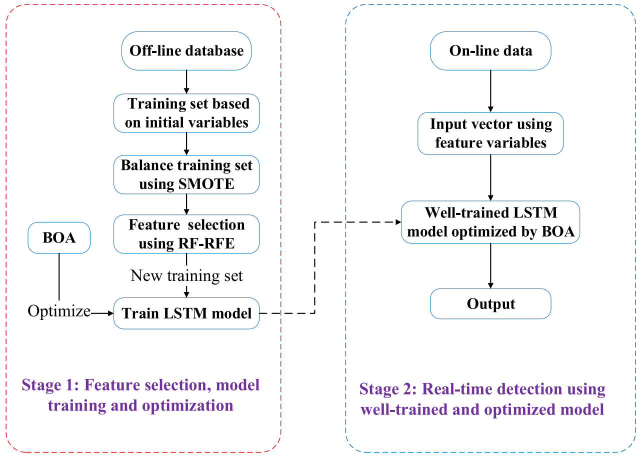

FIGURE 5. The flowchart of RF-RFE-BOA-LSTM based AID method.

F. THE PROPOSED METHOD (RF-RFE-BOA-LSTM)

The flowchart of the proposed method is illustrated in figure 5. As can be seen from Figure 5, the application of the proposed AID method includes two stages, and the main steps of the proposed method are as follows.

Step 1: SMOTE method is used to balance incident samples and non-incident samples in the dataset.

Step 2: RF-RFE algorithm is used to select feature variables for AID.

(2.1) The training set is built based on the initial variables and the input to the training set is:

$$Input = \left[\lambda_{i,j}\right] = \begin{bmatrix} \lambda_{1,1} & \lambda_{2,1} & \cdots & \lambda_{15,1} \\ \lambda_{1,2} & \lambda_{2,2} & \cdots & \lambda_{15,2} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{1,m} & \lambda_{2,m} & \cdots & \lambda_{15,m} \end{bmatrix} \quad (9)$$

The output to the training set is:

$$Output = \left[y_j\right] = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \quad (10)$$

where m is the total number of training samples and $\lambda_{i,j}$ is the i-th variable of the j-th input sample. The specific content of the i-th variable is shown in Table 1. $y_j \in \{0, 1\}$ is the label (target) corresponding to the j-th input sample. Incident label is represented by 1, and no-incident label is represented by 0.

(2.2) The Random Forest is used to evaluate the importance of the 15 initial variables, and the classification accuracy of the training set data is recorded.

(2.3) Delete the least important variable, the residual variables are used to construct a new training set; re-evaluate the importance of the residual variables, and classification accuracy of the new training set data is recorded.

(2.4) Repeat step 2.3 until all variables have been searched, that is, only one variable with the greatest importance remaining. After the above steps, the ranking of importance of 15 initial variables and the classification accuracy on the training set data corresponding to these variables can be obtained.

(2.5) When the highest classification accuracy rate is obtained, the input variables used are important variables.

Step 3: A training set with the feature variables is used to train LSTM network and its parameters are optimized by BOA.

Step 4: Input real-time collected feature variables data into the well-trained LSTM network and then determine whether there is a traffic incident based on the model output.

In summary, LSTM is the basic model of the proposed method, which detects traffic incidents by learning traffic variables data. In order to ensure better performance of LSTM, its hyperparameters are optimized by BOA. The input of LSTM is determined by the feature variables selected by RF-RFE to reduce the interference of redundant variables and improve the classification performance of LSTM. It is worth noting that the construction of a relatively comprehensive initial variables set is a prerequisite for effective selection of feature variables.

III. EXPERIMENTS

A. DATA DESCRIPTIONS AND PREPROCESSING

The experimental data is from the well-known traffic incident dataset of I-880 highway in the United States. The I-880 highway has a length of 9.2 miles and a number of lanes of 3-5, including some high occupancy lanes. There are 18 loop detection stations in the North Lane and 17 loop detection stations in the South Lane (see [43] for details). The database completely records the traffic volume, speed and occupancy collected by loops during the sampling period (sampling interval 30s), as well as the location and start and end time of the incident. Therefore, this database is widely used in the verification and evaluation of AID algorithms. The I-880 database includes 45 typical traffic incidents (4136 samples in total), from which 23 traffic incidents are randomly selected for training, and the remaining 22 traffic incidents are used for testing. Because the amount of non-incident data is too big, training set and test set are usually constructed from random non-incident samples that are not returned. In order to retain the information of the non-incident samples to a large extent, the proportion of incident samples in the training set and test set are both set to 20%. The composition of training set and test set is shown in Table 2. In this study, the training samples size is very close to the test samples size (instead of using most of the samples for training), in order to test the performance of the AID method with small samples size (especially small incident samples size).

In order to make the two types of samples relatively balanced, SMOTE is used to increase the traffic incident samples in the training set. The parameters of SMOTE are set as follows: the number of adjacent sample points is 5, and the oversampling magnification is 300%. The number of both types of samples in the balanced training set is 8144. In order to eliminate the influence of different dimensions, improve the training speed and classification accuracy, the data is normalized to the interval [0, 1]. The normalization formula is as follows.

TABLE 2. Training set and test set.

	Total number of samples	Number of incident samples	Number of incidents
training set	10180	2036	23
test set	10500	2100	22

$$y_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \tag{11}$$

where x_i is the value of original data, y_i is the value of normalized data, $x_{\max}$ is the maximum value of the original data, and $x_{\min}$ is the minimum value of the original data.

B. FEATURE VARIABLES SELECTION

Feature variables are selected from the initial variables set using the RF-RFE algorithm. The parameters that need to be determined include the number of random feature variables m_{try} and the number of decision trees. According to the suggestion in [37], the number of m_{try} is the square root of the total number of feature variables, so here $m_{try} = 4$. The number of decision trees is set to 1000. The initial variables obtained are in descending order of importance: {15, 12, 9, 13, 10, 6, 7, 14, 11, 3, 8, 4, 2, 5, 1}. Each time the variable ranked in the last is deleted, and the classification accuracy is recalculated. The curve of the classification accuracy with the number of variables is shown in Figure 6. The initial variables set contains 15 variables. At first, with the gradual deletion of variables, the classification accuracy is generally on the rise. Because deletion of some relatively unimportant variables can reduce the influence of redundant information on the algorithm. When the number of variables is 6, the highest classification accuracy is obtained, and then as the number of variables decreases, the classification accuracy gradually decreases, because the more important variables are removed. Therefore, the first six variables with the most importance are selected as feature variables (the difference between the measured values of upstream and downstream occupancy, the difference between the measured value and the predicted value of the downstream occupancy, the difference between the measured value and the predicted value of the upstream occupancy, the difference between the measured values of upstream and downstream traffic flow, the difference between the measured value and the predicted value of the downstream traffic volume, the measured value of downstream occupancy).

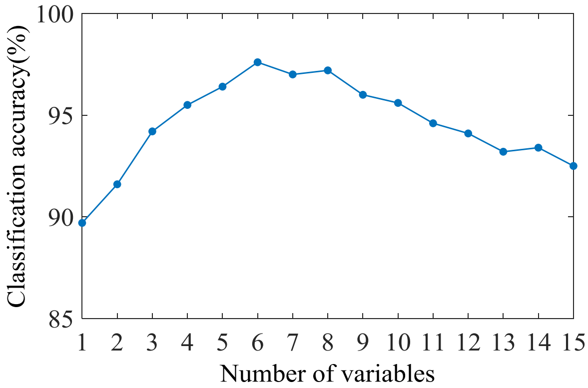

FIGURE 6. The relationship between classification accuracy and the number of variables.

FIGURE 7. A simple architecture of LSTM network.

The feature variables are used to construct the training set. The input matrix of the training set is:

$$\begin{bmatrix} \lambda_{15,1} & \lambda_{12,1} & \lambda_{9,1} & \lambda_{13,1} & \lambda_{10,1} & \lambda_{6,1} \\ \lambda_{15,2} & \lambda_{12,2} & \lambda_{9,2} & \lambda_{13,2} & \lambda_{10,2} & \lambda_{6,2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \lambda_{15,16288} & \lambda_{12,16288} & \lambda_{9,16288} & \lambda_{13,16288} & \lambda_{10,16288} & \lambda_{6,16288} \end{bmatrix} \tag{12}$$

where, each row vector represents a training sample, a total of 16288; each column vector represents a set of feature variables. The output (label) corresponding to the first 8144 input samples is 1, indicating incidents; and the output (label) corresponding to the 8144 input samples is 0, indicating non-incident.

C. TRAINING AND OPTIMIZATION OF LSTM

First, architecture of the LSTM network needs to be determined. Since the LSTM network has not been used for traffic incident detection, there are no references in the literature regarding an optimal internal architecture. As shown in Figure 7, a simple architecture of LSTM network is adopted, including an input layer, a LSTM layer, including an input layer, an LSTM layer, a fully connected layer, a softmax layer, and an output layer. The input layer contains 6 nodes corresponding to 6 feature variables for traffic incident detection. The LSTM layer is also called a hidden layer and contains some LSTM units. The network is followed by a fully connected layer with 2 nodes corresponding to the number of classes. To predict class labels, the network ends a softmax layer and a classification output layer. The output of the fully connected layer is classified as either 0 or 1 by "softmax". In the softmax layer, the classification probability is calculated, and the classification output layer classifies test data into two classes: the incident and non-incident. A binary

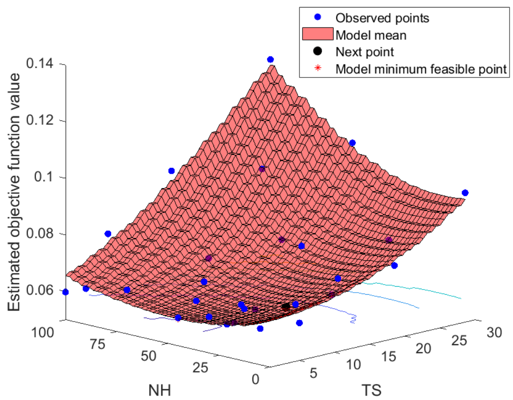

FIGURE 8. The objective function model.

cross-entropy loss function is selected as the cost function denoted by:

$$loss = -(ylog(y^{*}) + (1 - y)\, log(1 - y^{*})) \tag{13}$$

where y is the true label and y^{*} is the output from the full connected layer. In fact, the binary cross-entropy loss function is used to determine the closeness between the actual output and the expected output.

The weights and biases of the LSTM network are initialized randomly and Adaptive Moment Estimation (Adam) algorithm is used to optimize the internal parameters of LSTM network during training. Adam is a kind of stochastic gradient descent method with the following parameters. Initial learning rate = 0.001, beta1 = 0.9, beta2 = 0.999, decay = $1\mathrm{E}^{-5}$, mini batch size = 4, gradient threshold = 1 and max epochs = 100. There are two key external parameters directly influencing the output of the LSTM network. One is the time step and the other is the number of hidden layer nodes. The time step determines the amount of data (including a part of historical data) used for prediction/classification at the next time, which helps the LSTM network to learn long-term dependency information to obtain superior performance. The time step TS and the number of hidden layer nodes NH are tuned using BOA. The value range of the two parameters is set as follows: $TS \in \{1, 2, 3, \cdots, 30\}$ and $NH \in \{20, 40, 60, 80, 100\}$. The objective function of BOA is the classification loss of validation data. For training LSTM network, 5-fold cross validation is performed. Figure 8 shows the objective function model. Figure 9 shows the relationship between function evaluations and the minimum objective. The optimized parameters are calculated as $TS = 6$ and $NH = 20$, and the observed minimum of the objective function is 0.1409.

D. THE COMPARISONS AND ANALYSIS

In order to illustrate the effectiveness of balancing data sets by SMTOE and selecting feature variables by RF-RFE, the method using SMOTE and LSTM (SMOTE-LSTM), and the method using RF-RFE and LSTM (RF-RFE-LSTM), and the single LSTM method (LSTM) are used for comparison.

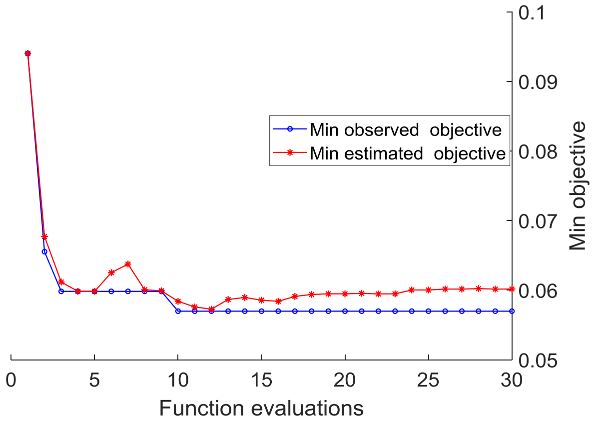

FIGURE 9. The relationship between function evaluations and the minimum objective.

TABLE 3. The hyperparameters of comparison methods.

Methods	Hyperparameters
KNN-SVM	The number of nearest neighbor points is 9, kernel parameter of Radial Basis Function (RBF) is 63.15, and penalty parameter is 20.49.
FDL	The deep learning model has four stacked Auto-Encoders whose number of neurons are 11, 12, 10 and 14 respectively.
WT-LR	Probability threshold is 0.5.
GAN-SVM	The kernel parameter of Radial Basis Function (RBF) is 30.63, and penalty parameter is 8.52.
TSA-SVM	The kernel parameter of Radial Basis Function (RBF) is 26.79, and penalty parameter is 13.24.

The BOA are also used for optimization of these comparison methods.

In addition, several state-of-the-art AID methods are also introduced for comparison, including SVM and KNN ensemble learning (KNN-SVM) [34], Fuzzy Deep Learning (FDL) [28], a hybrid AID method using Wavelet Transformation and Logistic Regression (WT-LR) [33], a hybrid AID method using GAN and SVM (GAN-SVM) [35], and Tabu Search Algorithm optimized-SVM (TSA-SVM) [22]. In order to ensure the performance of comparison methods. The parameters of these comparison methods are set and optimized according to the corresponding literatures.

In order to ensure the fairness of comparison, all AID methods are compared using the same dataset. The form of input data is also determined according to the corresponding literature. If there is no specific requirement, the form of input data are determined as the same as the proposed method. The hyperparameters of comparison methods are shown in Table 3.

1) EVALUATION CRITERIA

As mentioned in Section 1, traffic incident detection can be viewed as a binary-classication. In this study, the True

TABLE 4. Confusion matrix for AID methods evaluation.

	Predicted incident	Predicted incident
Actual incident	TP	FN
Actual incident	FP	TN

Positive (TP) indicates that an incident instance is correctly identified, and the true Negative (TN) indicates that a non-incident instance is correctly classified. The False Positive (FP) indicates an error that an incident instance is marked as a non-incident instance, and the False Negative (FN) indicates an error that a non-incident instance is marked as an incident instance. The above four states (FP, TN, FP and FN) can be more intuitively represented by the confusion matrix. Confusion matrix for AID evaluation is shown in Table 4.

Some evaluation criteria including True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR), False Negative Rate (FNR), Aaccuracy, Precision and F1-score are often employed to evaluate classification performance. These criteria are calculated by the following formula:

$$TPR = \frac{TP}{TP+FN} = Sensitivity = Recall \tag{14}$$

$$TNR = \frac{TN}{TN+FP} = Specificity \tag{15}$$

$$FPR = \frac{FP}{TN+FP} = 1 - TNR \tag{16}$$

$$FNR = \frac{FN}{TP+FN} = 1 - TPR \tag{17}$$

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{18}$$

$$Precision = \frac{TP}{TP+FP} \tag{19}$$

$$F1-score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \tag{20}$$

The TPR is also called Sensitivity or Recall, which represents the the ratio of positive correctly classified samples to the total number of positive samples. Whereas TNR is also called Specificity or inverse Recall is expressed as the ratio of the correctly classified negative samples to the total number of negative samples. Thus, the TPR represents the proportion of the negative samples that were correctly classified, and the TNR is the proportion of the positive samples that were correctly classified. The FPR represents the proportion of the negative samples that were incorrectly classified. The FNR represents the proportion of positive samples that were incorrectly classified.

Precision is also called positive prediction value that represents the proportion of positive samples that were correctly classified to the total number of positive predicted samples. Predictive values (positive and negative) reflect the performance of the prediction.

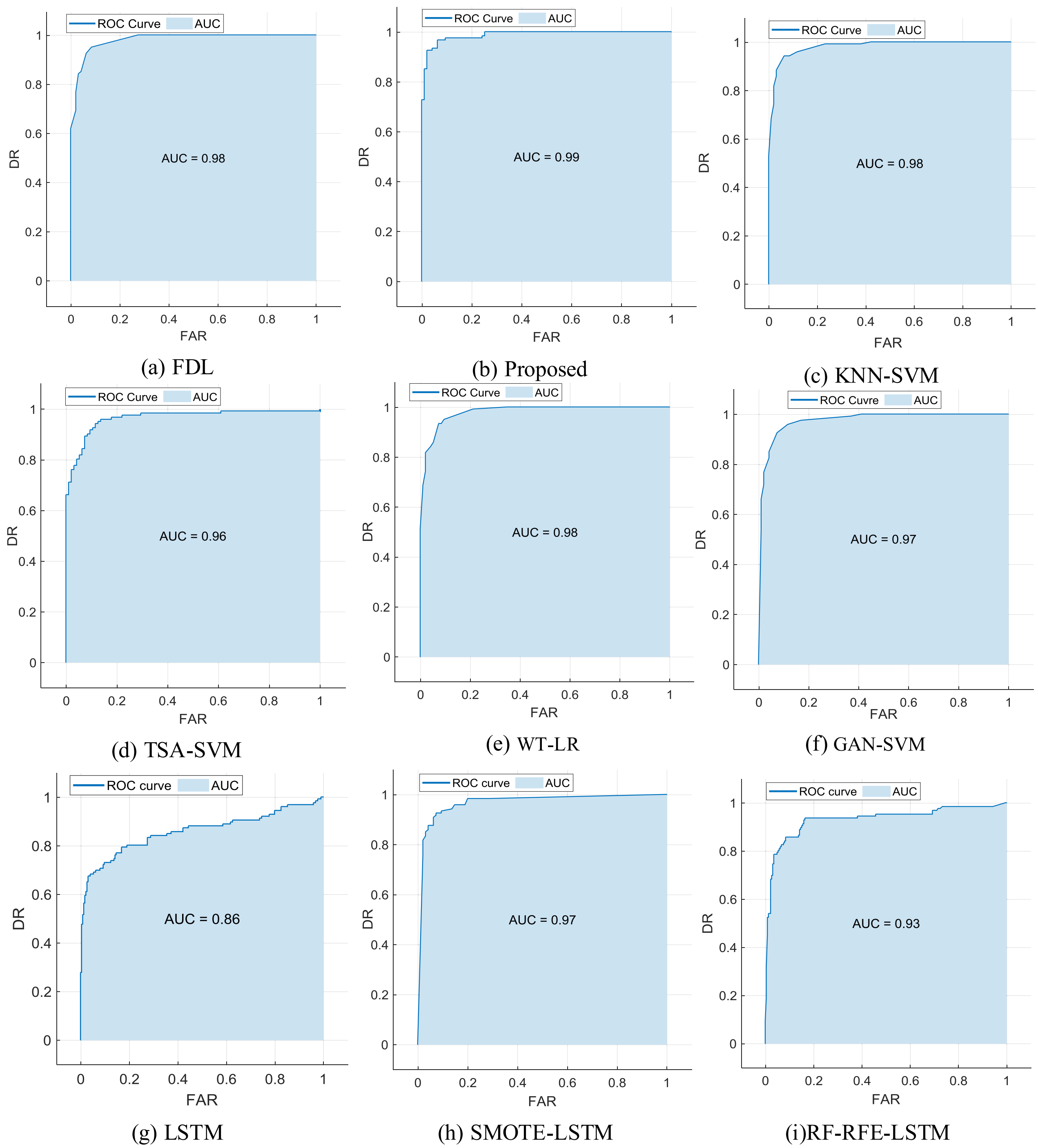

FIGURE 10. ROC curves of different methods, (a) FDL, (b) SMOTE-RF-RFE-LSTM, (c) KNN-SVM, (d) TSA-SVM, (e) WT-LR, (f) GAN-SVM, (g) LSTM, (h) SMOTE-LSTM, (i) RF-RFE-LSTM.

Accuracy is defined as a ratio between the correctly classified samples to the total number of samples. It is worth noting that Accuracy is sensitive to the imbalanced data.

F1-score represents the harmonic mean of Precision and Recall. The value of F1-score is ranged from zero to one, and the large values of F1-score indicate superior classification performance.

As important evaluation criteria for AID methods, TRP and FPR are usually called the Detection Rate (DR) and the False Alarm Rate (FAR) respectively. In addition, Mean Time to Detect (MTTD) is another important criterion to evaluate the performance of AID methods. MTTD is defined as the average of time elapsed between the actual start time (reported time) of the incident and time when the incident is first detected by an AID method. However, MTTD is not considered in some studies [6], [31], [34], [35], [47]. Obviously, we expect AID methods to obtain high DR (close to 100%), low FAR (close to 0%) and short MTTD (as short as possible).

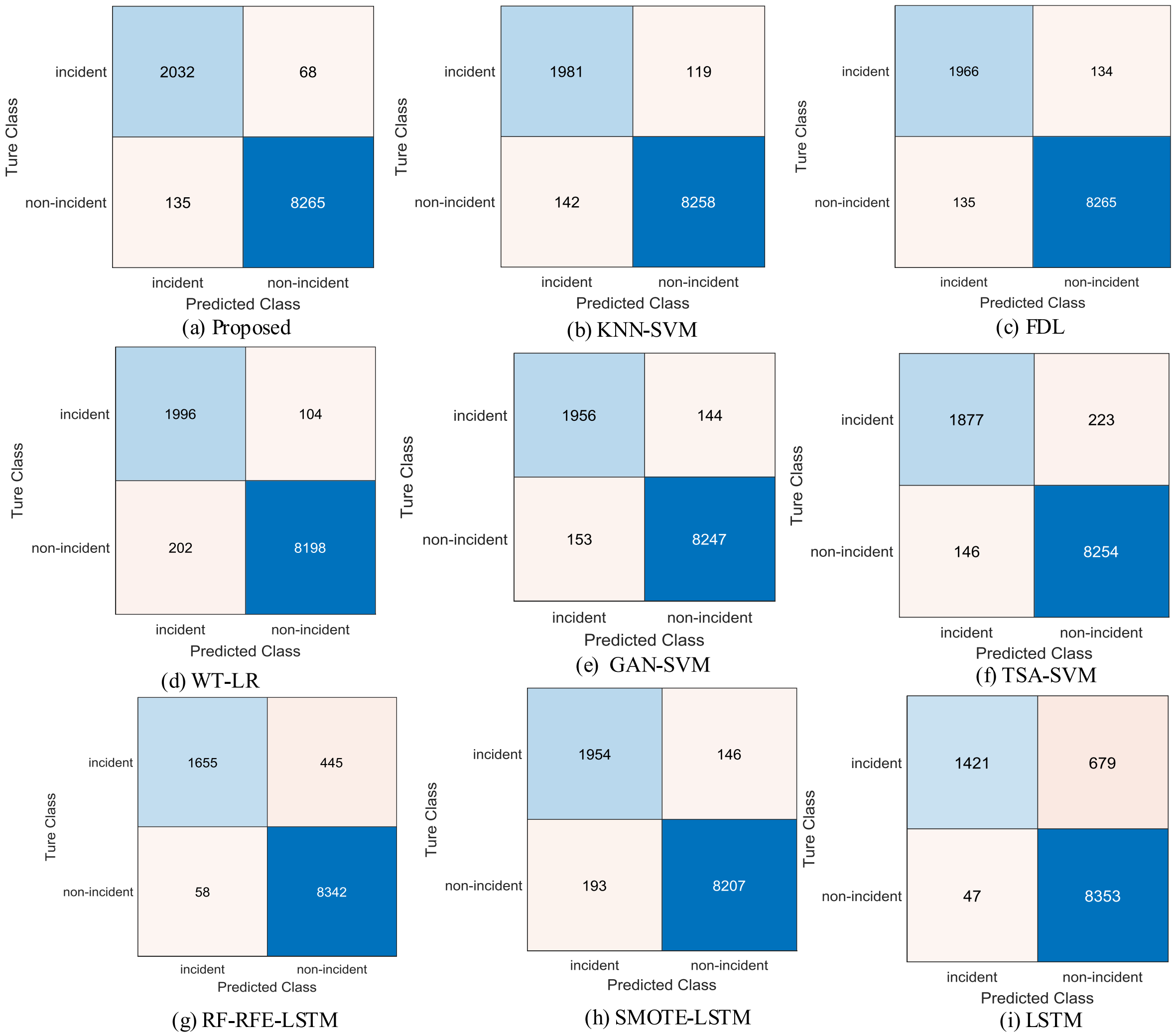

FIGURE 11. **The confusion matrix of each AID method, (a) proposed method, (b) KNN-SVM, (c) FDL, (d) WT-LR, (e) GAN-SVM, (f) TSA-SVM, (g) RF-RFE-LSTM, (h) SMOTE-LSTM, (i) LSTM.**

However, a high DR is often accompanied by a high FAR; while the FAR is reduced, the DR is also reduced. Similarly, short MTTD is often accompanied by high FAR. Therefore, when evaluating an AID method, these three criteria are needed to be traded off. In some studies, DR, FAR and MTTD are used to construct Performance Index (PI) for performance evaluation. In this study, the PI proposed in [15] is used to evaluate the performance of AID methods. The calculation formula of this PI is as follows.

$$PI = \left(1.01 - \frac{DR}{100}\right) \times \left(\frac{FAR}{100} + 0.001\right) \times MTTD \quad (21)$$

In addition to these criteria mentioned above, the Receiver-Operating-Characteristic (ROC) curve is also used to evaluate performance of AID method in our study. A ROC curve is drawn by plotting the DR (TPR) against the FAR (FPR) at any threshold. According to the detection threshold: both values of DR and FAR are 0 when the threshold is high enough, whereas they are equal to 1 when the threshold is low enough. Therefore, as a numeric performance metric the area under the ROC curve (AUC) represents the detection performance, with larger AUC values indicating better detection.

2) RESULTS AND DISCUSSION

Figure 10 shows the ROC curve and AUC obtained by each AID method. It can be seen from Figure 10 that the ROC curve of proposed method is closer to the upper left corner of the figure than that of other AID methods, the AUC of proposed method is 0.99, which is larger than that of other comparison methods. The results illustrate that the proposed method can better balance DR and FAR, that is, for the same FAR, the proposed method has a higher DR; for the same DR, the proposed method has a lower FAR. Therefore, compared with other methods, the proposed method performs better in terms of AUC. More specifically, the AUC of SMOTE-LSTM, RF-RFE-LSTM and LSTM are 0.97, 0.93 and 0.86, respectively, which indicates that the effect of balancing the data set using SMOTE is obvious, and selection

TABLE 5. Evaluation criteria values of different methods (the best result is highlighted in bold).

Methods	DR	FAR	Accuracy	Precision	F1-Score	MTTD (min)	PI
Proposed	**96.76%**	1.61%	**98.07%**	93.77%	**95.24%**	2.31	**0.00268**
KNN-SVM	94.33%	1.69%	97.51%	93.31%	93.82%	2.42	0.00283
FDL	93.62%	1.61%	97.44%	93.57%	93.60%	2.68	0.00311
WT-LR	95.05%	2.40%	97.09%	90.81%	92.88%	**2.20**	0.00273
GAN-SVM	93.14%	1.82%	97.17%	92.75%	92.94%	2.47	0.00292
TSA-SVM	89.38%	1.74%	96.49%	92.78%	91.05%	3.05	0.00358
RF-RFE-LSTM	78.81%	0.69%	95.21%	96.61%	86.81%	4.74	0.00508
SMOTE-LSTM	93.05%	2.30%	96.77%	91.01%	92.02%	2.63	0.00324
LSTM	67.67%	**0.56%**	93.09%	**96.80%**	79.65%	5.86	0.00621

of feature variables using RF-RFE can improve the classification performance of LSTM to a certain extent.

The confusion matrix of each AID method is shown in Figure 11. The TP, TN, FP, and FN obtained by each AID method on the test set are clearly presented by the confusion matrix. It can be seen from Figure 12, the proposed method has the largest sum of TP and TN (or the smallest sum of FP and FN). Especially, TP of the proposed algorithm is the largest, which indicates that it can identify the most incident samples. Moreover, the FP of the proposed method is relatively small, although it is larger than that of LSTM and RF-RFE-LSTM. LSTM and RF-RFE-LSTM have not balanced the dataset, thus they ignored the minority samples (incident samples) to a certain extent, which leads to a large number of incident samples not detected. Naturally, very few non-incident samples are misclassified as incident samples. Therefore, it is necessary to balance the dataset for traffic incident detection. In general, the proposed method has superior performance.

Table 5 illustrates evaluation criteria values (DR, FAR, Accuracy, Precision, F1-Score, MTTD, PI) of different AID methods. It can be seen from Table 5 that the four criteria of DR, Accuracy, F1-Score, and PI of the proposed method are all the best. The MTTD of proposed method is only slightly larger than the MTTD of WT-LR and is smaller than the MTTD of the remaining comparison methods. Although FAR and Precision of LSTM and RF-RFE-LSTM are better than that of the proposed method, other criteria of these two methods are obviously much worse. They get lower FAR and higher Precision at the cost of extremely low DR, that is, many incident samples are missed. F1-Score is a comprehensive criterion that balances DR and Precision. The F1-Scores of LSTM and RF-RFE-LSTM are 79.65% and 86.81% respectively, which are significantly lower than that of other methods. The F1-Score of the proposed method is as high as 95.24%. Especially in terms of PI, the PI value of the proposed method is the smallest one, which demonstrates that the overall performance of the proposed method is the best.

By comparing the criteria values of SMOTE-LSTM and LSTM, it is found that the performance of SMOTE-LSTM is significantly better. Specifically, its DR increased by 11.14%, F1 increased by 7.16%, MTTD decreased by 1.12 min, Accuracy increased by 2.12%, PI decreased by 0.00113, while FAR increased by only 0.13%, and precision decreased by only 0.18%. The above results show that SMOTE has a significant effect on dealing with imbalanced traffic incident data.

By comparing the criteria values of RF-RFE-LSTM and LSTM, it is found that the of RF-RFE-LSTM has better performance. Specifically, its DR increased by 25.38%, F1 increased by 12.37%, MTTD decreased by 3.23 min, Accuracy increased by 3.69%, PI decreased by 0.00297, while FAR increased by only 1.74%, and precision decreased by only 5.79%. The above results show that the feature variables selection using RF-RFE has achieved a certain effect, which not only helps to improve DR and reduce MTTD, but also keeps FAR and Precision almost unchanged.

In addition, the excellent performance of the proposed method also illustrates the effectiveness of SMOTE in dealing with imbalanced data and small sample data, and RF-RFE can capture the feature variables used for AID accurately.

IV. CONCLUSION

In this study, a hybrid method named RF-RFE-BOA-LSTM is proposed integrating the RF-RFE and the BOA-optimized LSTM network for traffic incident detection. Firstly, a relatively comprehensive set of initial variables is constructed using basic traffic variables and their combinations. Secondly, feature variables are selected using the RF-RFE algorithm. Then, the feature variables are used to construct a training set for the LSTM network, and the hyper-parameters of the LSTM network are optimized by BOA. Notably, the SMOTE is employed to solve the problem of imbalance between incident sample size and non-incident sample size. Finally, we conduct experiments to test performance of the proposed method based on the well-known I-880 data set, and several state-of-the-art AID methods are introduced for comparison. The confusion matrix and ROC curve are used to illustrate the experimental results more intuitively. The experimental results illustrate that the proposed AID method outperforms all comparison AID methods in terms of AUC, DR, Accuracy, F1-Score and PI. In terms of MTTD, the proposed AID method achieved the second-best results. Based on the comparison and analysis of the experimental results, the conclusions can be drawn that the proposed method is a better method for traffic incident detection, because of its excellent performance.

In the future, more data sets (particularly collected from urban roads) should be used to test the performance of proposed AID method for drawing a more general conclusion. To further improve performance, if available, more spatiotemporal data could be used for construction of initial variables set.

REFERENCES

[1] S. Tang and H. Gao, "Traffic-incident detection-algorithm based on nonparametric regression," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 38–42, Mar. 2005.

[2] H. Dia and K. Thomas, "Development and evaluation of arterial incident detection models using fusion of simulated probe vehicle and loop detector data," *Inf. Fusion*, vol. 12, no. 1, pp. 20–27, Jan. 2011.

[3] Y.-S. Jeong, M. Castro-Neto, M. K. Jeong, and L. D. Han, "A wavelet-based freeway incident detection algorithm with adapting threshold parameters," *Transp. Res. C, Emerg. Technol.*, vol. 19, no. 1, pp. 1–19, Feb. 2011.

[4] Y. Asakura, T. Kusakabe, L. X. Nguyen, and T. Ushiki, "Incident detection methods using probe vehicles with on-board GPS equipment," *Transp. Res. C, Emerg. Technol.*, vol. 81, pp. 330–341, Aug. 2017.

[5] E. D'Andrea and F. Marcelloni, "Detection of traffic congestion and incidents from GPS trace analysis," *Expert Syst. Appl.*, vol. 73, pp. 43–56, May 2017.

[6] X. Liu, H. Cai, R. Zhong, W. Sun, and J. Chen, "Learning traffic as images for incident detection using convolutional neural networks," *IEEE Access*, vol. 8, pp. 7916–7924, Jan. 2020.

[7] C. Long Mak and H. S. L. Fan, "Development of dual-station automated expressway incident detection algorithms," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 480–490, Sep. 2007.

[8] M. S. Shehata, J. Cai, W. M. Badawy, T. W. Burr, M. S. Pervez, R. J. Johannesson, and A. Radmanesh, "Video-based automatic incident detection for smart roads: The outdoor environmental challenges regarding false alarms," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 2, pp. 349–360, Jun. 2008.

[9] J. Ren, B. Li, Y. Liu, Y. Chen, L. Xin, and J. Shi, "Detecting and positioning of traffic incidents via video-based analysis of traffic states in a road segment," *IET Intell. Transp. Syst.*, vol. 10, no. 6, pp. 428–437, Aug. 2016.

[10] H. Payne and S. Tignor, "Freeway incident-detection algorithms based on decision trees with states," *Transp. Res. Rec.*, vol. 682, pp. 30–37, Jan. 1978.

[11] C. L. Dudek, C. J. Messer, and N. B. Nuckles, "Incident detection on urban freeways," *Transp. Res. Rec.*, vol. 495, no. 495, pp. 12–24, 1974.

[12] B. N. Persaud and F. L. Hall, "Catastrophe theory and patterns in 30-second freeway traffic data—Implications for incident detection," *Transp. Res. A, Gen.*, vol. 23, no. 2, pp. 103–113, Mar. 1989.

[13] Y. J. Stephanedes and A. P. Chassiakos, "Application of filtering techniques for incident detection," *J. Transp. Eng.*, vol. 119, no. 1, pp. 13–26, Jan. 1993.

[14] A. R. Cook and D. E. Cleveland, "Detection of freeway capacity reducing incidents by traffic-stream measurements," *Transp. Res. Rec.*, vol. 495, pp. 1–11, Jan. 1974.

[15] P. Chakraborty, C. Hegde, and A. Sharma, "Data-driven parallelizable traffic incident detection using spatio-temporally denoised robust thresholds," *Transp. Res. C, Emerg. Technol.*, vol. 105, pp. 81–99, Aug. 2019.

[16] A. Karim and H. Adeli, "Comparison of fuzzy-wavelet radial basis function neural network freeway incident detection model with California algorithm," *J. Transp. Eng.*, vol. 128, no. 1, pp. 21–30, Jan. 2002.

[17] D. Srinivasan, X. Jin, and R. L. Cheu, "Evaluation of adaptive neural network models for freeway incident detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 1, pp. 1–11, Mar. 2004.

[18] A. B. Parsa, H. Taghipour, S. Derrible, and A. Mohammadian, "Real-time accident detection: Coping with imbalanced data," *Accident Anal. Prevention*, vol. 129, pp. 202–210, Aug. 2019.

[19] F. Yuan and R. L. Cheu, "Incident detection using support vector machines," *Transp. Res. C, Emerg. Technol.*, vol. 11, nos. 3–4, pp. 309–328, Jun. 2003.

[20] S. Chen, W. Wang, and H. van Zuylen, "Construct support vector machine ensemble to detect traffic incident," *Expert Syst. Appl.*, vol. 36, no. 8, pp. 10976–10986, Oct. 2009.

[21] J. Xiao and Y. Liu, "Traffic incident detection using multiple-kernel support vector machine," *Transp. Res. Record: J. Transp. Res. Board*, vol. 2324, no. 1, pp. 44–52, Jan. 2012.

[22] B. Yao, P. Hu, M. Zhang, and M. Jin, "A support vector machine with the tabu search algorithm for freeway incident detection," *Int. J. Appl. Math. Comput. Sci.*, vol. 24, no. 2, pp. 397–404, Jun. 2014.

[23] J. Xiao, X. Gao, Q.-J. Kong, and Y. Liu, "More robust and better: A multiple kernel support vector machine ensemble approach for traffic incident detection," *J. Adv. Transp.*, vol. 48, no. 7, pp. 858–875, Nov. 2014.

[24] S. Chen and W. Wang, "Decision tree learning for freeway automatic incident detection," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 4101–4105, Mar. 2009.

[25] Q. Liu, J. Lu, S. Chen, and K. Zhao, "Multiple Naïve Bayes classifiers ensemble for traffic incident detection," *Math. Problems Eng.*, vol. 2014, Apr. 2014, Art. no. 383671.

[26] D. Singh and C. K. Mohan, "Deep spatio-temporal representation for detection of road accidents using stacked autoencoder," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 879–887, Mar. 2019.

[27] P. Chakraborty, Y. O. Adu-Gyamfi, S. Poddar, V. Ahsani, A. Sharma, and S. Sarkar, "Traffic congestion detection from camera images using deep convolution neural networks," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2672, no. 45, pp. 222–231, Jun. 2018.

[28] C. El Hatri and J. Boumhidi, "Fuzzy deep learning based urban traffic incident detection," *Cognit. Syst. Res.*, vol. 50, pp. 206–213, Aug. 2018.

[29] Y. Fang, Q. Yang, L. Zheng, X. Zhou, and B. Peng, "A deep cycle limit learning machine method for urban expressway traffic incident detection," *Math. Problems Eng.*, vol. 2020, Jul. 2020, Art. no. 5965089.

[30] T. Huang, S. Wang, and A. Sharma, "Highway crash detection and risk estimation using deep learning," *Accident Anal. Prevention*, vol. 135, Feb. 2020, Art. no. 105392.

[31] L. Li, X. Sheng, B. Du, Y. Wang, and B. Ran, "A deep fusion model based on restricted Boltzmann machines for traffic accident duration prediction," *Eng. Appl. Artif. Intell.*, vol. 93, Aug. 2020, Art. no. 103686.

[32] J. Wang, X. Li, S. S. Liao, and Z. Hua, "A hybrid approach for automatic incident detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1176–1185, Sep. 2013.

[33] S. Agarwal, P. Kachroo, and E. Regentova, "A hybrid model using logistic regression and wavelet transformation to detect traffic incidents," *IATSS Res.*, vol. 40, no. 1, pp. 56–63, Jul. 2016.

[34] J. Xiao, "SVM and KNN ensemble learning for traffic incident detection," *Phys. A, Stat. Mech. Appl.*, vol. 517, pp. 29–35, Mar. 2019.

[35] Y. Lin, L. Li, H. Jing, B. Ran, and D. Sun, "Automated traffic incident detection with a smaller dataset based on generative adversarial networks," *Accident Anal. Prevention*, vol. 144, Sep. 2020, Art. no. 105628.

[36] Y. Liu, J.-W. Bi, and Z.-P. Fan, "Multi-class sentiment classification: The experimental comparisons of feature selection and machine learning algorithms," *Expert Syst. Appl.*, vol. 80, pp. 323–339, Sep. 2017.

[37] Y. Tian, K. Zhang, J. Li, X. Lin, and B. Yang, "LSTM-based traffic flow prediction with missing data," *Neurocomputing*, vol. 318, pp. 297–305, Nov. 2018.

[38] J. Mackenzie, J. F. Roddick, and R. Zito, "An evaluation of HTM and LSTM for short-term arterial traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1847–1857, May 2019.

[39] L. Mou, P. Zhao, H. Xie, and Y. Chen, "T-LSTM: A long short-term memory neural network enhanced by temporal information for traffic flow prediction," *IEEE Access*, vol. 7, pp. 98053–98060, Jul. 2019.

[40] G. Douzas, F. Bacao, and F. Last, "Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE," *J. Artif. Intell. Res.*, vol. 465, pp. 1–20, Oct. 2018.

[41] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[42] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 389–422, Jan. 2002.
[43] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[44] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Proc. Adv. Neural Inform. Process. Syst.*, 2012, pp. 2951–2959.
[45] A. Klein, S. Falkner, S. Bartels, P. Hennig, and F. Hutter, "Fast Bayesian hyperparameter optimization on large datasets," *Electron. J. Statist.*, vol. 11, no. 2, pp. 4945–4968, 2017.
[46] Q. Shang, D. Tan, S. Gao, and L. Feng, "A hybrid method for traffic incident duration prediction using BOA-optimized random forest combined with neighborhood components analysis," *J. Adv. Transp.*, vol. 2019, Jan. 2019, Art. no. 4202735.
[47] H. Teng and Y. Qi, "Application of wavelet technique to freeway incident detection," *Transp. Res. C, Emerg. Technol.*, vol. 11, nos. 3–4, pp. 289–308, Jun. 2003.

QIANG SHANG received the Ph.D. degree from the Transportation College, Jilin University, Changchun, China, in 2017. He is currently a Lecturer with the Shandong University of Technology, Zibo, China. He has authored more than 20 academic articles in journals. His research interests include traffic data analysis, traffic model, and intelligent transportation systems.

LINLIN FENG received the Ph.D. degree from the School of Psychology, Shandong Normal University, Jinan, China, in 2017. She is currently a Lecturer with the Shandong University of Technology, Zibo, China. She has been hosting the National Social Science Foundation Project and the Ministry of Education (MOE) in China Project of Humanities and Social Sciences. Her research interests include traffic psychology, behavior, and safety.

SONG GAO is currently the Dean and a Professor with the School of Transportation and Vehicle Engineering, Shandong University of Technology, Zibo, China. He is also a middle-aged Expert with outstanding contribution in Shandong Province and the Executive Director of the China Automotive Engineering Society. He has published more than 80 articles. He has presided over the National 863 Plan of Electric Vehicle Major Project and more than 30 other projects. His current research interests include energy system matching theory and the control technology of electric vehicle, intelligent vehicles, and intelligent transportation systems.

Dr. Gao has received the Second Prize of national level teaching achievements and six prizes of provincial level scientific and technological progress.