IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 3, 2019, accepted November 23, 2019, date of publication December 6, 2019,
date of current version December 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2958198

A Novel Intrusion Detection Method in Train-Ground Communication System

BING GAO AND BING BU
State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China

Corresponding author: Bing Gao (17111041@bjtu.edu.cn)

This work was supported in part by the Graduate Innovation Fund of Beijing Jiaotong University under Grant I18JB00110, in part by the Innovation Fund Project of Beijing Traffic Control Technology under Grant 9907006507, in part by the National Natural Science Foundation of China under Grant 61973026, Grant 61603031, and Grant I19L00090, and in part by the Beijing Laboratory of Urban Rail Transit.

ABSTRACT At present, the train-ground communication system based on the wireless communication protocol is a very important component of communication-based train control (CBTC) systems in intelligent transportation. Its information security is worthy of attention. In order to guarantee the security of the train-ground communication system, this paper proposes an improved AdaBoost multi-classification intrusion detection method based on the n-gram model. First, the n-gram model is used to model the state transitions of the IEEE 802.11 protocol. Then, a typical normal behavior set and typical abnormal behavior sets are obtained by learning and they can portray typical behaviors of their respective classes. Furthermore, a similarity measure algorithm is proposed to construct AdaBoost weak classifiers, which improves the classification effect of AdaBoost algorithm. At last, an AdaBoost multi-classification algorithm is presented to detect and identify the attacks. Experiments prove that the algorithm can effectively detect and distinguish attack types in the train-ground communication system.

INDEX TERMS Intrusion detection, train-ground communication, Denial of Service, similarity measure, AdaBoost, multi-classification.

I. INTRODUCTION

With the development of city scale and economic level, there are increasingly higher demands for punctuality, energy-efficiency, comfort and security of public transportation. As an advanced passenger transportation method, the urban rail transit satisfies the above needs. Therefore, it becomes popular in many large and medium-sized cities. As a kernel part of the urban rail transit, the communication-based train control (CBTC) system adopts a train-ground communication subsystem to transmit control commands and status information of trains based on the IEEE 802.11 protocol [1]–[3]. Because the train-ground communication system is very important for train operation, it may face various hostile attacks, such as Denial of Service (DoS), session hijacking and MAC address spoofing attacks. Once the train-ground communication system is attacked, the network availability will get restricted and the train operation control will be affected. In serious cases, the network may be paralyzed and public security incidents may be caused. So it is important to research the information security of wireless train-ground communication system in the urban rail transit [4].

To ensure the safety of train operation, the existing urban rail transit systems adopt various kinds of failure elimination mechanisms. However, information security is not given enough emphasis. The current information security defense methods mainly come from the general information technology (IT) and cannot meet the needs of the train-ground communication system, so the intrusion detection method needs to be specifically designed for the train-ground communication system.

Urban rail transit systems are mainly deployed in underground tunnels. The large amounts of reflections, scattering, and barriers can severely affect the performance of wireless communication. Zhang [5] describes the influence of the large-scale fading for urban rail transit systems. Guan *et al.* [6] present the propagation characteristics of near shadowing, path loss, shadow fading, fast fading, level crossing rate, and average fade duration in 2.4 GHz based on a real environment in Madrid subway. On account of the CBTC special characteristics, such as high mobility speed, fixed moving direction, and strict requirement for accurate train-location

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

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see http://creativecommons.org/licenses/by/4.0/

information, the train-ground communication system also needs to be given special consideration. Considering the path loss, fast fading and shadowing with high mobility, Lin *et al.* [7] present a novel finite state Markov channel model. The study of [8] indicates that wireless channels of the train-ground communication system are time-varying where the signal to noise ratio (SNR) is changing rapidly. In summary, the special communication environment makes the train-ground communication different from traditional mobile communication in terms of transmission delay, handover delay and data packet dropout, which leads to the increase of packet retransmission.

Intrusion detection is a prevalent information security defense method for the network. It includes two types: anomaly detection and misuse detection [9].

The anomaly detection method firstly defines a normal behavior model for system. For a newly received data, this method determines whether the system is attacked by comparing the normal behavior model with the data. Ioannou *et al.* [10] present an anomaly detection system which uses a binary logistic regression (BLR) statistical model to identify the nature of the sensor activity as malicious or benign activity. Sun *et al.* [11] propose an improved V-detector algorithm based on a three-level intrusion detection model composed by the base station, the detection nodes and the ordinary nodes to detect the cyber attacks. Usha and Kavitha [12] introduce a support vector machine (SVM) intrusion detection method based on the normalized gain in media access control (MAC) layer. Shams and Rizaner [13] propose an intrusion detection method based on the SVM which trains the normal model and uses it to judge attack behaviors. Faisal *et al.* [14] use the received signal strength (RSS) in the physical layer to detect the intrusion of wireless network. Alipour *et al.* [9] propose a wireless intrusion detection method based on behavior analysis of the IEEE 802.11 protocol, which defines a threshold value deviating from the normal model. Based on this threshold value, the attacks can be detected.

The advantages of anomaly detection method are that the attack model does not need to be constructed and the database of attack types does not to need to be artificially updated. Furthermore, the method can detect all the unknown attacks. But the disadvantage of the anomaly detection method is that it depends on the normal behavior model deeply. A small deviation from the normal behavior model in the learning phase may result in a big detection deviation in the testing phase.

In misuse detection method, it requires modeling attack behaviors. Accordingly, the system detects these attacks based on their models. Cao *et al.* [15] present a mathematical model for misbehaving nodes based on the resource sharing percentage. This method can detect the attacks based on the traffic flows. In [16], seven machine learning algorithms are used to classify the abnormal behaviors. Yao *et al.* [17] propose a novel Sybil attack detection method based on received signal strength indicator in vehicular Ad Hoc networks. OConnor and Reeves [18] use signatures of the attacks for discovering reconnaissance, DoS, and information theft attacks on bluetooth enabled devices. Onat and Miri [19] introduce an intrusion detection scheme based on packet arrival rate anomalies and receive power anomalies.

Obviously, the merit of the misuse detection method is that it can clearly obtain the attack types. On the other hand, the defect of this method is evident. It can not detect unknown attacks, and it needs to update the attack feature database artificially. Because the attack types are changeable, the slight changes of the attacks may lead to the attack detection failure.

Up to now, in the urban rail transit system, the correlation research of information security is still inadequate. The existing works mostly concern the security of device. Teo *et al.* [20] propose a simulator, named OpenRails, for railway cyber-security analysis. Then a generic API framework is developed on OpenRails for the information security study. However, this work is still in an early stage. Wu *et al.* [21] propose a challenge-response authentication process to mitigate the vulnerabilities on the standard balise air-gap interface.

In general, the information security issue of the train-ground communication system is not paid enough attention in the CBTC system. To address this problem, an effective intrusion detection method is proposed. The main contributions of the proposed method are summarized as follows. Firstly, an improved AdaBoost binary classification algorithm is proposed. In this method, a similarity measure algorithm is given to construct the weak classifiers. The similarity measure can balance the influence between the n-gram character strings and their frequency. The similarity measure algorithm improves the classification performance of AdaBoost binary classification algorithm. Secondly, based on the improved AdaBoost binary classification algorithm, an AdaBoost multiple classification algorithm is proposed to detect and identify attacks. Finally, this algorithm obtains better detection results through integrating the wireless and wired detection results.

This paper is organized as follows. In Section II, the CBTC system and its attack types are described. In Section III, an improved AdaBoost multiple classification intrusion detection method is presented based on the n-gram model and train states. In Section IV, experimental details and results verification are given. Finally, the study is concisely concluded in Section V.

II. A BRIEF INTRODUCTION TO CBTC SYSTEM AND ITS ATTACK TYPES

A. A BRIEF INTRODUCTION TO CBTC SYSTEM

A typical CBTC system is depicted in Fig. 1. It is composed of a central control subsystem, a wayside control subsystem, a vehicle control subsystem and a train-ground communication subsystem. Automatic train supervision (ATS) of the central control subsystem lies at the control central. ATS located in the station, zone controller (ZC), computer interlocking (CI) and data storage unit (DSU) consist of the wayside control subsystem. The vehicle control subsystem

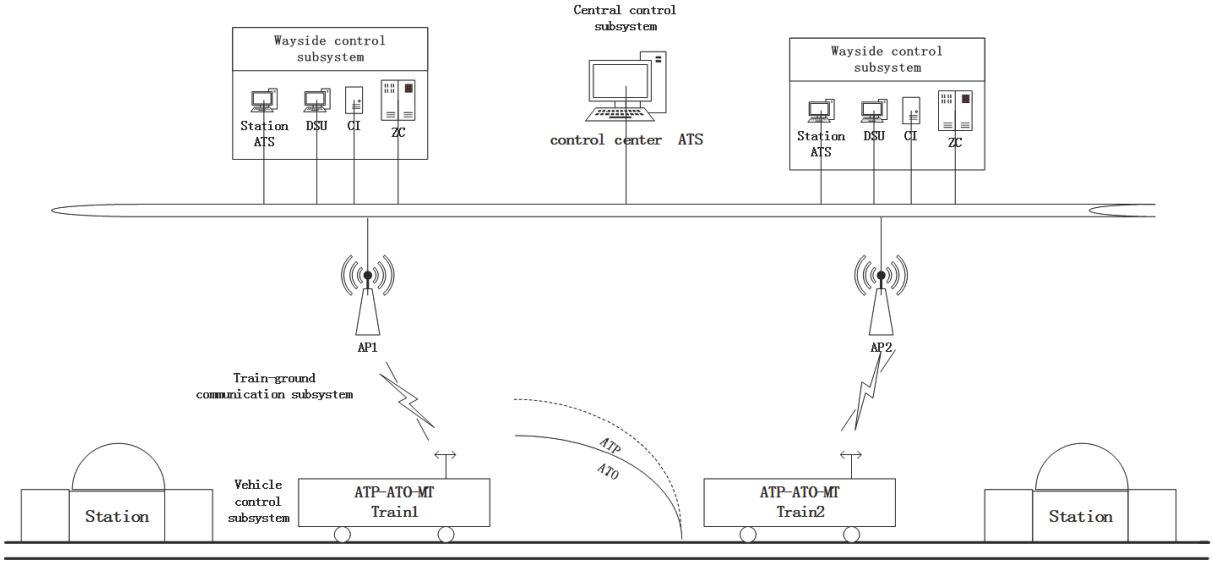

FIGURE 1. A typical architecture of the CBTC system.

includes automatic train operation (ATO) and automatic train protection (ATP). The train-ground communication subsystem includes access points which connect wayside equipments and vehicle terminals.

CBTC is a crucial system to keep train operation safe and realize the automatic train operation in urban rail transit. By the wired and wireless communication, various subsystems realize the bidirectional communication between ground equipments and trains and constitute a closed loop control system. CBTC is also an integrated train operation control system based on safety devices. Specifically, it is used to command the train operation, adjust train operation deviations and drive train automatically.

ATS located at the control center generates time table and sends operation information to all trains. CI sets the safe route for trains by controlling signalling equipments. Simultaneously, ZC generates the movement authority (MA) of the train, which is send to the vehicle control equipments to ensure train safe operation. For the MA generation, ZC collects data, such as train position, velocity and the driving direction information.

Based on the received MA, ATP computes the train protection curve and supervises the deviations between the real speed and the emergency brake trigger speed. When the real speed is greater than the emergency brake trigger speed, the train will brake urgently so as to guarantee a safe stop within the MA protection scope.

ATO realizes the automatic driving, automatic speed adjustment and the automatic door controlling based on ATP protection.

The train-ground communication subsystem includes up-links and down-links between the vehicle control subsystem and the wayside control subsystem. In the operation process, CBTC transmits the data periodically to control the train operation. At the beginning of communication cycle, the vehicle control subsystem obtains the train velocity and position information by the sensors installed on the train and send them to the ZC. ZC computes and sends the MA to the vehicle control subsystem. Finally, ATO and ATP calculate the train control commands based on MA.

B. CBTC SYSTEM ATTACK TYPES

As a typical industry control system, the attacks on the CBTC system are different from traditional IT network system attacks. The reasons can be described as follows. Firstly, private protocols are massively applied in industry control systems. With the development of protocol reverse parsing technology and systematic opening, the private protocols gradually become unprotected protocols. Secondly, in order to keep steady operation, industry control system usually runs for months or years without interruption, which makes equipment obsolete, difficult to update or reboot. Thirdly, the control logic is fixed and the physical equipment is affected by network information attacks in the industry control system. In CBTC system, the information security attacks can be divided into two types: Denial of Service (DoS) attack and data spoofing attack [22].

The DoS attacks can prevent normal information exchange by disrupting the integrity of data among communicating objects or exploiting vulnerabilities of devices in the system. The DoS attacks can be divided into two types, called semantic attack and violent attack. The semantic attack utilizes system vulnerability or defect to attack target hosts, which makes target hosts reject services. The violent attack sends a large number of requests to consume network and host resources. These requests exceed the processing capacity of host or network. The semantic attack does not need a lot of data packets and only needs few data packets to realize the attack.

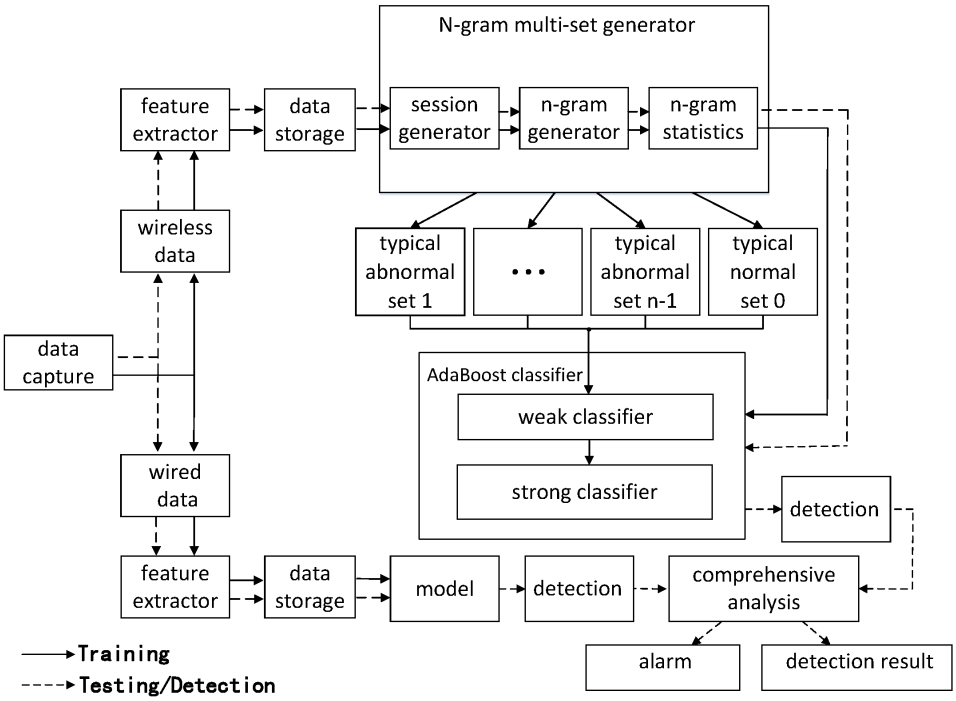

FIGURE 2. An intrusion detection system diagram based on the n-gram model.

If the DoS attacks can be performed in CBTC system, the data packet will be delayed or discarded, and as a result, train cannot periodically receive the control command information. In serious cases, the train stops frequently. Data spoofing attack mainly modifies the contents of payloads in the data pockets to affect the execution of physical device. Generally, the False Data Injection (FDI) attack can be regarded as data spoofing attack. The attacker falsifies the obtained data to mislead the receiver. As a result, the receiver makes a wrong judgment. In addition, man-in-the-middle attack is an important attack in CBTC system, which can be included in the data spoofing attack. In wired network, man-in-the-middle attack is realized by the Address Resolution Protocol (ARP). In wireless network, as a middleman, the attacker not only plays a ''false sender'' role in communicating with the original receiver, but also plays a ''false receiver'' role in communicating with the original sender, which makes the communication data can be intercepted and falsified by this middleman. In CBTC system, the communication data is used to transmit the important control commands. If the data is falsified by an attacker, which can affect train operation security.

III. AN IMPROVED ADABOOST MULTIPLE CLASSIFICATION ALGORITHM BASED ON THE N-GRAM MODEL

A. AN INTRUSION DETECTION SYSTEM BASED ON THE N-GRAM MODEL

This paper proposes an intrusion detection method based on the n-gram model. The n-gram model, as a critical concept in natural language processing, is used to evaluate the comparability between the two character strings. Based on the Markov assumption, the n-gram model can describe that the appearing probability of any character string only relates to the finite one or several character strings in front of itself. In general, when modelling, the number of finite character strings cannot be too big. Otherwise, the string combination space becomes sparse. Jurafsky and Martin [23] indicate that the generative texts from different corpus have no repeatability proven by experiments. In our paper, the n-gram model presented in [9] is introduced into the proposed method, shown in Fig. 2. In the method, no public data set is available, so the specific data set of the train-ground communication system needs to be collected.

In Fig. 2, the data capture module, the feature extractor module and the data storage module are used to generate the database of the wired and wireless train-ground communication system, and the database includes the normal data and the abnormal data. Specifically, the data capture module firstly captures the IEEE 802.11 wireless data of the train-ground communication system in MAC layer. Further, the feature extractor module extracts kinds of features from the captured frames, including protocol type, type, subtype, source address, destination address, retry flag, train position and train velocity. The protocol type is used to guarantee that the protocol is the IEEE 802.11 protocol. Source address and destination address indicate the communication endpoints. Based on the communication endpoints, sessions can be established. The contents of sessions are composed of type and subtype which can be found in the MAC frame header and reflect the communication flow among the endpoints. The train position and train velocity is very

important safety-critical information, which can reflect the attack impact on train states [24]. In addition, for DoS attack, the retransmission times can be regard as an attack feature, so those retransmission frames are retained. For management and data frame, the retry flag indicates whether this frame is retransmission frame or not. The retry flag of retransmission frame is 1. Unlike management and data frame, control frame do not have to wait in the sequence to retransmit, so there is no retransmission frame and the retry flag is 0. Finally, the data storage module is used to store the generated data from the feature extractor module and generate a database which includes the normal data and the abnormal data.

In Fig. 2, three modules constitute the n-gram multi-set generator module. They are session generator module, n-gram generator module, and n-gram statistics module. By using the n-gram multi-set generator module, a data set is mapped to a n-gram model and indicated as: $V \stackrel{f}{\Rightarrow} V^f$, where V denotes a data set and V^f means the n-gram model expression of the data set. Specially, the session generator module classifies the frames to obtain a series of sessions according to the source and destination MAC addresses, which means that all the frames with the same srcMac and dstMac addresses are categorized in the same session. A communication session S_l can be represented as a sequence of different exchanged frames in MAC layer:

$$S_l = [\tau_1, \ldots, \tau_k] \tag{1}$$

where τ_i represents a type-subtype string, $i = 1, 2, \ldots, k$. Therefore, a communication session can indicate the state exchanges according to time series. Further, a session fragment, namely sub-session, is considered in a time interval $\Delta T_i = [t_i, t_{i+1}]$:

$$S_{l,\Delta T_i} = [\tau_{j_1}, \ldots, \tau_{j_k}] \tag{2}$$

Any n consecutive messages are regarded as a n-gram pattern in a sub-session, called a n-gram. The n-gram generator module is used to build the n-gram model based on above sub-sessions. For example, the sliding window sized n is used to partition the sub-session $S_{l,\Delta T_i}$ and a multi-set of n-gram is obtained:

$$Q_{l,\Delta T_i} = \Big\{[\tau_{j_1}, \ldots, \tau_{j_n}], [\tau_{j_2}, \ldots, \tau_{j_{n+1}}], \ldots, [\tau_{j_{k-n+1}}, \ldots, \tau_{j_k}]\Big\} \tag{3}$$

where every element of $Q_{l,\Delta T_i}$ is a n-gram and it can partially reflect the order of the frames and the state exchanges in the 802.11 protocol. It should be noted that a multi-set is a set where the element can repeat [25]. Consequently, $Q_{l,\Delta T_i}$ can be defined as a set of two-dimensional vector (ng, c) where "ng" represents a n-gram and "c" represents times of ng. The n-gram statistics module is used to collect all the (ng, c) and generate a n-gram model which means that the n-gram model is a set of all the (ng, c).

The typical abnormal set module and the typical normal set module are used to collect typical abnormal and normal n-gram patterns. They can be obtained by learning. They are saved to constitute the needed sets which are regarded as the classification reference for abnormal and normal behaviors.

The AdaBoost classifier module is composed of a weak classifier module and a strong classifier module. Given weak classifier weight coefficient matrices, weak classifiers are structured based on the proposed similarity measure algorithm on the typical normal set and the typical abnormal set. Then, these weak classifiers can be combined to be a strong classifier.

In this intrusion detection system diagram, a comprehensive detection conclusion can be obtained by incorporating the wired intrusion detection result into the wireless intrusion detection result. Meanwhile, it can send out alarm information to the system. Specifically, the train position and velocity features are used to detect attacks. In every period, the MA is calculated. If the MA dose not change during the continuous Q periods, an attack is considered to happen. The threshold Q can be obtained by training.

In general, the intrusion detection method can be divided into anomaly detection and misuse detection [9]. In this paper, we combine the two methods. The mixed intrusion detection method means that the method combines the anomaly detection and misuse detection. Therefore, the mixed intrusion detection method can identify the normal behaviors and attack behaviors simultaneously.

B. SIMILARITY MEASURE

In this subsection, a concept named similarity is introduced and several algorithms about the similarity measure are given. The concept "similarity" is used to construct weak classifiers for a training set in the following subsection. Unless otherwise explicitly stated, TS denotes a training set. For TS, a normal typical feature set is expressed as follows:

$$\text{NTFS} = \{(ng_1^N, c_1^N), \ldots, (ng_r^N, c_r^N)\} \tag{4}$$

where the superscript "N" denotes normal behaviors, ng_i^N is a typical normal n-gram character string and c_i^N is the upper bound of its frequency in normal behaviors, $i = 1, \ldots, r$. For a new received ng_i^N, if its frequency is greater than c_i^N, it belongs to an abnormal behavior.

Furthermore, an abnormal typical feature set of a certain type of attack is expressed as follows:

$$\text{ATFS} = \{(ng_1^A, c_1^A), \ldots, (ng_s^A, c_s^A)\} \tag{5}$$

where the superscript "A" denotes abnormal behaviors. ng_i^A is a typical abnormal n-gram character string and c_i^A is the threshold of its frequency, $i = 1, \ldots, s$. For a new received ng_i^A, if its frequency is higher than the threshold, it is viewed as an abnormal behavior.

Considering any $(ng_1, c_1) \in$ NTFS or ATFS and $(ng_2, c_2) \in$ TS, similarity is defined as follows:

$$S = \begin{cases} p * \left(\dfrac{L_1}{L_2}\right)^2 + q * F^2, & \text{if } L_1 > 0 \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

Algorithm 1 Similarity Measure With NTFS

Input: (1) any $(ng, c) \in$ TS; (2) NTFS; (3) weight coefficient pair (p, q).

Output: similarity S^N between (ng, c) and NTFS.

1: **for all** $(ng_i^N, c_i^N) \in$ NTFS **do**
2: $ng_1 \leftarrow ng_i^N, c_1 \leftarrow c_i^N, ng_2 \leftarrow ng, c_2 \leftarrow c$.
3: Calculate F according to Eq. (7).
4: Calculate the similarity according to Eq. (6), denoted by S_i^N.
5: **end for**
6: $S^N = \max S_i^N$.
7: **return** S^N.

Here L_1 is the number of the same type-subtypes between the ng_1 and the ng_2 by bitwise comparison. L_2 is the number of type-subtypes in the ng_2. And the item "$\frac{L_1}{L_2}$" is the ratio which represents the type-subtype similarity between ng_2 and ng_1. F is a decision about whether (ng_2, c_2) is similar with (ng_1, c_1) from the view of frequency. Specifically, for $(ng_1, c_1) \in$ NTFS, its definition is expressed as follows:

$$F = \begin{cases} 1, & \text{if } c_2 < c_1, \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

For $(ng_1, c_1) \in$ ATFS, its definition is expressed as follows:

$$F = \begin{cases} 1, & \text{if } c_2 > c_1, \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

(p, q) is a pair of weight coefficients which is used to adjust or balance the contributions from the respective items and $p, q \in (0, 1)$. From Eq. (6), It is noted that if $L_1 = 0$, (i.e., there is no same type-subtype between the ng_1 and the ng_2), the similarity is equal to 0.

Now, based on Eq. (6), a similarity measure algorithm between any $(ng, c) \in$ TS and NTFS is given in Algorithm 1.

Furthermore, according to Eq. (6), a similarity measure algorithm between any $(ng, c) \in$ TS and ATFS is given in Algorithm 2.

C. WEAK CLASSIFICATION

In this subsection, a similarity-based weak classification method is given for TS which includes normal data and attack data.

Suppose that the number of data category is n for TS. First, n typical feature sets (TFS) are extracted from TS and these sets are denoted by $\text{TFS}_1, \ldots, \text{TFS}_n$ where TFS_i can portray typical behaviors of the class "i", $i = 1, \ldots, n$.

Furthermore, the weak classifier construction method is described in Algorithm 3 where every weak classifier gives a rough classification result.

In the step 16 of Algorithm 3, there may exist k_1 and k_2 so that $S_{j,k_1} = S_{j,k_2} = \max\limits_{1 \le k \le n} S_{j,k}$. According to the step 17 of Algorithm 3, (ng_i, c_i) can be classified as class k_1 or class k_2. Considering that Algorithm 3 is only a weak classification algorithm, such classification results are acceptable. Besides, l pairs of weak classification weight coefficients (p_i, q_i) in Algorithm 3 can be written as the following matrix

$$W = \begin{pmatrix} p_1 & q_1 \\ p_2 & q_2 \\ \vdots & \vdots \\ p_l & q_l \end{pmatrix}$$

Algorithm 2 Similarity Measure With ATFS

Input: (1) any $(ng, c) \in$ TS; (2) ATFS; (3) weight coefficient pair (p, q).

Output: similarity S^A between (ng, c) and ATFS.

1: **for all** $(ng_i^A, c_i^A) \in$ ATFS **do**
2: $ng_1 \leftarrow ng_i^A, c_1 \leftarrow c_i^A, ng_2 \leftarrow ng, c_2 \leftarrow c$.
3: Calculate F according to Eq. (8).
4: Calculate the similarity according to Eq. (6), denoted by S_i^A.
5: **end for**
6: $S^A = \max S_i^A$.
7: **return** S^A.

Algorithm 3 Weak Classifier Construction Method Based on Similarity

Input: (1) TS with n-class data and its typical feature sets $\text{TFS}_1, \ldots, \text{TFS}_n$; (2) l pairs of weak classification weight coefficients (p_i, q_i) where every (p_i, q_i) can generate a weak classifier and $p_i, q_i \in (0, 1)$, $i = 1, 2, \ldots, l$.

Output: weak classifiers WC_i, $i = 1, 2, \ldots, l$.

1: **for** $i = 1, \ldots, l$ **do**
2: $p \leftarrow p_i, q \leftarrow q_i$.
3: **for all** $(ng_j, c_j) \in$ TS **do**
4: $ng \leftarrow ng_j, c \leftarrow c_j$.
5: **for** $k = 1, \ldots, n$ **do**
6: **if** TFS_k is a normal typical feature set **then**
7: NTFS $\leftarrow \text{TFS}_k$.
8: Execute Algorithm 1.
9: $S_{j,k} \leftarrow S^N$.
10: **else**
11: ATFS $\leftarrow \text{TFS}_k$.
12: Execute Algorithm 2.
13: $S_{j,k} \leftarrow S^A$.
14: **end if**
15: **end for**
16: Solve

$$k* = \arg \max_{1 \le k \le n} S_{j,k}.$$

17: $\text{WC}_i : (ng_j, c_j) \rightarrow k*$, which means that the i-th weak classifier WC_i classifies (ng_j, c_j) as the $k*$-th class behavior.
18: **end for**
19: **end for**
20: **return** WC_i, $i = 1, 2, \ldots, l$.

and it is called a weak classification weight coefficient matrix.

D. ADABOOST ALGORITHM BASED ON MULTIPLE CLASSIFICATION

For a complicated system, it is a better choice to obtain a comprehensive analysis based on the multiple experts' decisions, rather than the decision of any one of them. Based on this idea, an AdaBoost (an abbreviation for Adaptive Boosting) algorithm generates a strong classifier by combining several weak classifiers. Specifically, the AdaBoost algorithm enhances the weights of these samples which are mistakenly classified by the previous iteration and reduces the weights of those samples which are accurately classified by the previous iteration. As the weight increased, those data that are mistakenly classified can receive more attention from the weak classifier in the next iteration. Simultaneously, the AdaBoost algorithm uses the weighted majority voting method to combine these weak classifiers. More specifically, the weight of the weak classifier with a small classification error rate is increased to make it play a larger role in the voting. Otherwise, the weight of the weak classifier with a large classification error rate is reduced to make it play a smaller role in the voting. The above-mentioned idea conceives the well-known AdaBoost binary-classification algorithm. It is used to solve the two-class classification problem and cannot directly address the multi-class classification problem. To solve the multi-class classification problem, the most straightforward idea is to transform a multi-classification problem into a two-classification problem and then get multi-category results based on the voting mechanism. A feasible idea is described as follows.

For a training data set with n-type ($n > 2$) labels, two types of labels are selected from them and then the corresponding data is extracted to obtain a group of training data subsets. In this way, $n*(n-1)/2$ groups of training data subsets can be derived.

Each group of training data subsets only has two types of labels and consequently this problem is a two-class problem. The above-mentioned two-class classification AdaBoost algorithm can be used to generate a strong classifier. As there are $n*(n-1)/2$ groups of training data subsets, a total of $n*(n-1)/2$ two-class AdaBoost strong classifiers can be obtained.

For actual testing, a new received data (ng, c) is sent to the above-mentioned $n*(n-1)/2$ two-class AdaBoost strong classifiers and a total of $n*(n-1)/2$ two-class classification results are derived. By using the voting mechanism, a final classification result is given.

Based on the above statements, an AdaBoost multi-classification construction algorithm is described in Algorithm 4.

Algorithm 4 AdaBoost Multi-Classification Construction Algorithm

Input: (1) labeled training data set $\mathrm{T} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, where $x_i = (ng_i, c_i) \in \mathrm{TS}$, y_i is the class label of x_i and $y_i \in Y = \{1, \ldots, n\}$, $i = 1, 2, \ldots, N$, with n denoting the num of categories and $n > 2$; (2) all the typical feature sets $\mathrm{TFS}_1, \ldots, \mathrm{TFS}_n$, where TFS_i can portray typical behavior of the i-th class data, $i = 1, \ldots, n$; (3) weak classification weight coefficient matrix W_i, $i = 1, 2, \cdots, \frac{n*(n-1)}{2}$.

Output: binary-classification AdaBoost strong classifiers $G_i(x)$, $i = 1, 2, \cdots, \frac{n*(n-1)}{2}$.

1: Use k to represent a binary-classifier serial number and initialize $k = 0$.
2: **for** $i = 1, \ldots, n-1$ **do**
3: **for** $j = i+1, \ldots, n$ **do**
4: $k \leftarrow k + 1$.
5: $S_k = \{(x, y) \in \mathrm{T} \mid y \in \{i, j\}\}$.
6: **for all** $(x, y) \in S_k$ **do**
7: **if** $y == i$ **then**
8: $y = 1$.
9: **else**
10: $y = -1$.
11: **end if**
12: **end for**
13: Taking S_k with TFS_i and TFS_j, and W_k as the input, execute Algorithm 3 and generate weak classifiers.
14: Taking S_k and the weak classifiers generated in the last step as the input, using AdaBoost binary-classification algorithm and derive an AdaBoost binary-classification strong classifiers $G_k(x)$.
15: **end for**
16: **end for**
17: **return** $G_i(x)$, $i = 1, 2, \cdots, \frac{n*(n-1)}{2}$.

In the step 5 of Algorithm 4, a two-class data set S_k with class labels i and j is derived. Considering the class label set $Y = \{-1, +1\}$ in the AdaBoost binary-classification algorithm, it is needed to relabel S_k with class labels $+1$ and -1 to use the AdaBoost binary-classification algorithm, which corresponds to the steps from 6 to 12 in Algorithm 4.

Furthermore, an AdaBoost multi-classification detection algorithm is proposed in Algorithm 5. Based on the wired intrusion detection, a comprehensive intrusion detection result is given, which is regarded as a reasonable conclusion. If wireless data is attacked and train states are affected, the attack can be considered as an effective attack. If wireless data is attacked and train states are not affected, the attack is viewed as an insufficient attack. It may be that safety computer carries out data verification in time, which makes train states safe.

Furthermore, because the traditional AdaBoost algorithm is sensitive to the data noise, it is not suitable for the current CBTC data set. Specifically, in the train-ground communication system, there exist complicated environments, such as underground tunnel environment, train high-speed and linear motion, and quick switching of AP. These environments lead

Algorithm 5 AdaBoost Multi-Classification Detection Algorithm

Input: (1) binary-classification AdaBoost strong classifiers $G_i(x)$, $i = 1, 2, \cdots, \frac{n*(n-1)}{2}$, which are derived based on Algorithm 4; (2) any (ng, c) from a testing set or actual data set.

Output: classification result on (ng, c).

Use k to represent a binary-classifier serial number and initialize $k = 0$.
Denote the vote counting of the i-th class by c_i and initialize $c_i = 0, i = 1, \ldots, n$.
for $i = 1, \ldots, n-1$ **do**
  for $j = i+1, \ldots, n$ **do**
    $k \leftarrow k + 1$.
    $x \leftarrow (ng, c)$.
    $y \leftarrow G_k(x)$.
    if $y == 1$ **then**
      $c_i = c_i + 1$.
    else
      $c_j = c_j + 1$.
    end if
  end for
end for
Solve

$$k* = \arg \max_{1 \le k \le n} c_k.$$

return $k*$.

to random delay and packet dropout, which can be regarded as the noise of data. The noise data looks like an attack, but it is actually a random interference [26]. In other words, the random interference noise data looks like DoS attack data, but it is still normal data. Consequently, these noise data are easily wrongly classified. Because the traditional AdaBoost algorithm mainly focuses on those samples which are difficult to be classified, it is hypersensitive to noise data. Based on this fact, the attention of the noise data should be weakened in our data set and the idea comes from the communication noise processing field where the noise is often decreased and the signal is enhanced.

In the proposed AdaBoost algorithm, the concept of similarity is introduced to construct weak classifier. In this case, most of weak classifiers have a classification accuracy greater than 0.5. Few weak classifiers have a classification accuracy less than 0.5, which are discarded. Simultaneously, the sample weights are optimized based on the maximal classification accuracy. The advantages are obvious. Firstly, selecting similarity measure algorithm can improve classification accuracy of weak classifiers to make final classification accuracy of weak classifier combination approach 1. Secondly, the sample weights are optimized and updated by selecting the maximum weak classification accuracy, which weakens the weights of noise samples. Thirdly, the similarity measure method is simple for avoiding overfitting. For AdaBoost algorithm, the simpler weak classification algorithm is, the better effect it has in preventing overfitting. Finally, based on the efficient typical feature sets, the detection results can avoid trapping in local optimum.

TABLE 1. Station information of Beijing Subway Line 7.

ZC	CI	Station
ZC1	CI1	Beijing West Railway Station
		Wanzi
		Daguanying
ZC2	CI2	Guangan Men Nei
		Caishikou
		Hufang Qiao
		Zhushikou
	CI3	Qiaowan
		Ciqikou
		Guangqu Men Nei
ZC3	CI4	Guangqu Men Wai
		Shuangjing
		Jiulong Shan
ZC4	CI5	Dajiao Ting
		Baizi Wan
		Huagong
		NanlouZi Zhuang
ZC5	CI6	Happy Valley Scenic Area
		Fatou
		Shuanghe
	CI7	Jiaohua Chang
ZC6	CI8	Jiaohua Chang Cheliang Duan

IV. SIMULATION RESULTS AND DISCUSSIONS

In this section, the experiment is introduced and detection results are analyzed. Firstly, the train-ground communication data set was collected from the Beijing Subway Line 7 simulation platform, including normal and attack data. Then, the proposed intrusion detection method was carried out and its performance was evaluated.

A. DATA SET COLLECTION

In this paper, a Beijing Subway Line 7 simulation platform is used to simulate the train operation. The subway map is described in Table 1. In this simulation platform, the data set of train-ground communication system is collected.

In the field of information security, there are two types of intrusion detection data sets, namely wired network data set and wireless network data set. For the wired network data set, the KDDCUP 99 data set [27] is widely used. It contains almost 5 million records and has 41 features. Every record is viewed as a sample and it has been labeled as normal or abnormal. Further, these abnormal samples include 39 kinds of attacks which can be divided into 4 categories. For the wireless network data set, the AWID data set [28] developed in 2015 is a widely used data set and it is used to detect WLAN-based attacks. In the AWID, a wireless frame is regarded as a sample. Every sample contains 155 features, covering all the fields of MAC layer and Radiotap information. The AWID includes 16 kinds of attacks which can be divided into 3 categories: injection attack, flood attack and impersonation attack.

TABLE 2. Attack types.

No	Attack types	Description	Attribution
1	Beacon Flood	Forge a large number of beacon frames to make the terminals unable to access the correct AP (access point)	DoS
2	Probe Request Flood	Forge a large number of probe request frames to consume legitimate AP resources	DoS
3	Probe Response Flood	Forge a large number of probe response frames to affect station access networks	DoS
4	Authentication Flood	Forge a large number of authentication frames to break the connection between the legitimate user and the AP	DoS
5	Association Flood	Forge a large number of association frames to break the connection between the legitimate user and the AP	DoS
6	Deauthentication Flood	Forge a large number of deauthentication frames to break the connection between the legitimate user and the AP	DoS
7	Deassociation Flood	Forge a large number of deassociation frames to break the connection between the legitimate user and the AP	DoS
8	RTS Flood	Forge RTS frames with a large Duration value to block the wireless channels	DoS
9	CTS FLood	The fake legal node sends a large number of CTS frames to cause other nodes to stop sending data.	DoS
10	Man-in-the-middle Attack	Intercept and falsify the normal communication data between the legitimate user and AP.	Data spoofing attack
11	UDP Flood	Attack train-ground communication system AP through the wired network	DoS
12	ICMP Flood	Make the network channel be blocked by flow among backbone network switch, ZC and AP	DoS
13	TCP/SYN Flood	Attack train-ground communication system AP through the wired network	DoS

As a typical industrial control system, CBTC system uses the train-ground communication subsystem based on the IEEE 802.11 protocol to transmit train control commands periodically. Existing public intrusion detection data sets are not suitable for the train-ground communication system. The reasons can be described as follows. Firstly, the public intrusion detection data set can only be used to detect those attacks which are based on the general protocol. Traditional IT networks mostly use public network protocols, but many protocols for CBTC system are private, such as RSSP-I protocol and SFP protocol. Furthermore, in CBTC system, many intrusions can only be perceived in these private protocols, and they cannot be detected in the general protocols. Secondly, as a classical cyber-physical system, in CBTC system, some attacks are difficult to reflect in the network data. These attack features can only be displayed in physical characteristics, such as train position and train velocity. Finally, there are many problems for the existing data sets, such as the outdated KDDCUP 99 and the AWID with missing sample values, which affects the use of these data sets.

For CBTC system, when constructing a new data set, the following rules should be considered. Firstly, attack types should conform to the characteristics of CBTC system. Specifically, when selecting an attack type, it is necessary to consider whether the attack can affect the train operation, because the CBTC system adopts a large number of functional security mechanisms, such as two-out of three or double two out of two security computer and redundant network design. Although the existing data sets include most of attack types, the effective attack types are limited. Secondly, the features selected in the sample should reflect the impact of attack on train states. The selected sample features should simultaneously consider the physical and cyber features, such as communication protocols, traffic, system state parameters, and control input and output parameters. Thirdly, the sample values of the data set cannot be missing too much, otherwise the accuracy of the training model will be affected.

In the process of data set construction, the selection of attack types and sample features is a very important task. According to the above-mentioned rules, the selected attack types and features are described in table 2 and table 3 respectively. In table 2, the attack types include wireless attacks and wired attacks, because the train-ground communication system not only includes wireless network, but also includes wired network. For the attacks between the AP and the VOBC, in addition to the wireless attack itself leading to the communication interruption, attacking the AP through the wired network also causes communication interruption between the AP and the VOBC. Specifically, UDP flood, TCP/SYN flood and ICMP flood are sent to consume the resources of AP by the wired network, so that the AP cannot respond to the normal communication request. Therefore, UDP flood, TCP/SYN flood and ICMP flood attacks are added into the attack types between AP and VOBC in this paper. These attacks are DoS attacks. In addition, man-in-the-middle attack has been added to the attack types.

In table 3, the selected sample features not only include MAC frame header field and traffic statistics, but also include physical features. The reasons are described as follows. In CBTC system, the background traffic of train-ground communication is stable, and the network topology seldom changes. If a traffic-based DoS attack occurs, traffic will change significantly. So it is meaningful to extract traffic features. In addition, as a typical cyber-physical system, when suffering from attacks, the results of CBTC attacks can be reflected in physical devices. So it is also necessary to add physical features in table 3. In this paper, 14 features are selected as sample features of our data set and they are from various aspects and are more comprehensive for CBTC system.

TABLE 3. Sample features.

No	Feature	Expression	Description	Type	Source
1	protocol type	PT	protocol type	discrete	network
2	duration	D	channel occupancy time of wireless frame	continuous	MAC frame
3	type	T	frame type	discrete	frame control
4	subtype	ST	frame subtype	discrete	frame control
5	retry flag	R	frame retransmission flag	discrete	frame control
6	source address	SA	transmission source	discrete	MAC frame
7	destination address	DA	the last receiving address	discrete	MAC frame
8	receiver address	RA	represent the wireless workstation which is responsible for processing the frames	discrete	MAC frame
9	transmission address	TR	represent the wireless interface which transmits frames to a wireless medium	discrete	MAC frame
10	the percentage of manage frame	MN	the percentage of manage frames in a time interval	continuous	traffic feature
11	the percentage of control frame	CN	the percentage of control frames in a time interval	continuous	traffic feature
12	the percentage of data frame	DN	the percentage of data frames in a time interval	continuous	traffic feature
13	position	P	train positon	continuous	physical feature
14	velocity	V	train velocity	continuous	physical feature

It is worth noting that there is a data imbalance problem between the attack data and the normal data in this data set. To address this problem, oversampling method is used to balance the data. The principle of the oversampling method is to generate the synthetic minority class samples to balance the distribution between the samples of the majority and minority classes [29]. SMOTE [30] is a famous oversampling method. The algorithm improves the sample random replication algorithm and avoids overfitting. Its main idea is to generate new minority class samples by interpolating between several adjacent minority class samples [31]. It is also worth noting that this data set is primarily applicable to train-ground communication system. The selected attacks are mainly DoS attacks, and the attack target mainly concentrates on the communication network between VOBC and AP.

B. ATTACK SCENARIO DESCRIPTION

The CBTC information flow is described in Fig. 3. VOBC periodically transmits train information, such as direction of motion, position and velocity, to ZC by train-ground communication network. Based on the received information, in conjunction with the route setting and track occupancy information, ZC computers MA and sends it to VOBC. VOBC sends the platform screen door information to CI by the train-ground communication network. After receiving the platform screen door information, CI controls the movement of screen door and replies the state of the screen door to VOBC. VOBC sends train control and scheduling information to ATS by the train-ground communication network, such as the number of train group, train number, moving direction, velocity, position, train operation mode and screen door state. Meanwhile, ATS sends scheduling information to VOBC, such as train classes-priority, train waiting for a receiving track and the surplus time of train stop. According to the information flow, the bidirectional data between ZC and VOBC was collected.

In the experiment, the normal train operation scenario means that the train operated based on the normal information flow mentioned in Fig. 3. The attack scenario means the attacks were implemented. The attack information flow is also described in Fig. 3, highlighted in red. In the experiment, three trains operated on the line and AP switching was considered. The sampling frequency of AP switching was 500 milliseconds. The IEEE 802.11 protocol was used as communication protocol between the train and the AP. The trains started from the Baizi Wan station and stopped at Shuangjing station. When the train mode was upgraded from restricted manual mode (RM) to automatic train operation mode (AM), the attacks were performed. One attack tool is mdk3, which is used for the wireless network. The other is hping3, which is used for the wired network. In train-ground communication network, deauthentication flood, disassociation flood, authentication flood, association flood, beacon flood, RTS flood, CTS flood, UDP flood, TCP/SYN flood and ICMP flood were implemented. Particularly, ICMP flood, TCP/SYN flood and UDP flood are wired attacks. These attacks affect AP, which causes train-ground communication breakdown. The breakdown can cause the train control information transmission failure. As a result, train degradation occurs. Attack frequency can be divided into random and persistent frequency. The attack frequency includes 5 times per second, 10 times per second, 20 times per second, 30 times per second, 50 times per second, 100 times per second, 200 times per second, 300 times per second, 500 times per second, 700 times per second, 1000 times per second, 1500 times per second, 1800 times per second, 2000 times per second and 3000 times per second.

C. EXPERIMENTAL DETAILS AND RESULTS VERIFICATION

1) IMPLEMENTATION DETAILS

In this experiment, normal data and attack data are selected from the data set to validate the proposed intrusion detection method. The attack data includes authentication and association attack data, beacon attack data, CTS attack data, disassociation and deauthentication attack data and RTS attack data. Their quantity information is listed in table 4.

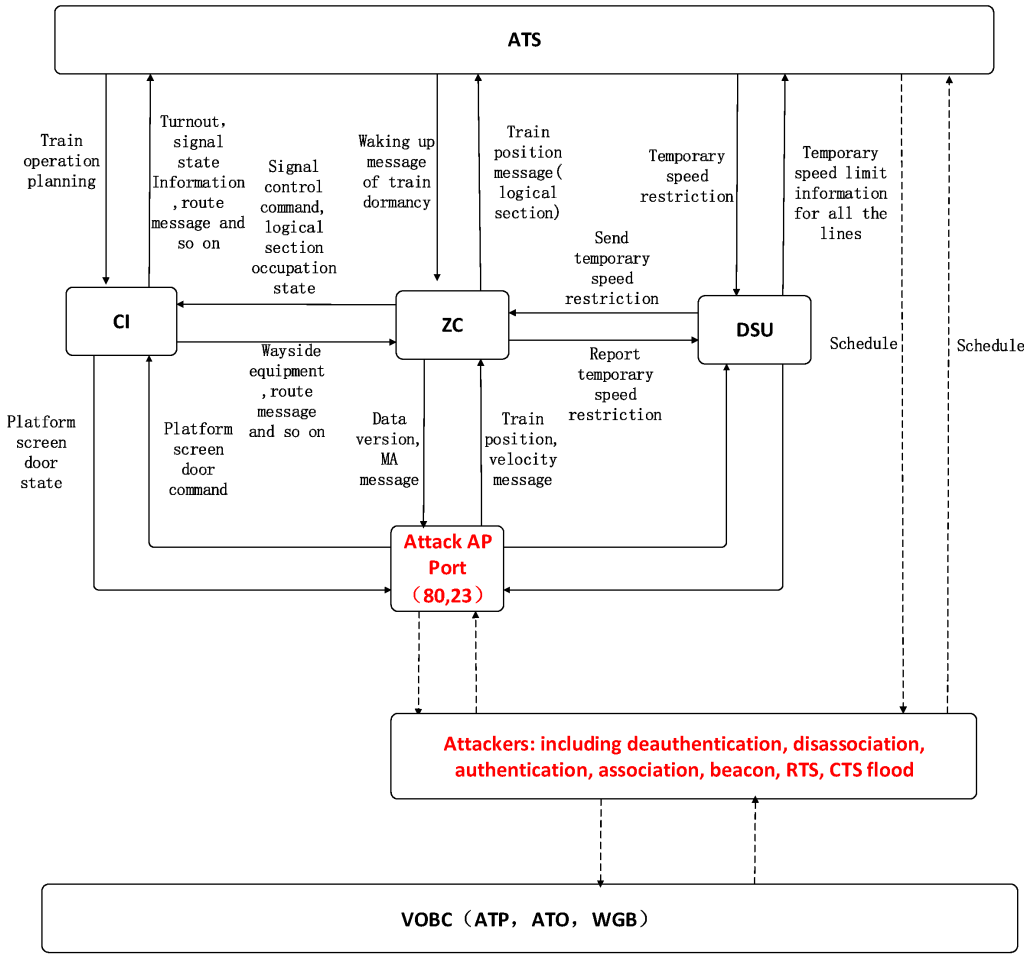

FIGURE 3. The information flow of CBTC system under attack.

The detection period (or the observation time window) is 10 seconds. Furthermore, the size of sliding time window is 4 in the n-gram model, which has been proven to be a better choice in [9]. By learning the collected data, typical feature sets are obtained for normal behavior, authentication and association attack behavior, beacon attack behavior, CTS attack behavior, disassociation and deauthentication attack behavior and RTS attack behavior.

Although the exponential growth of the number of n-grams may exist in some object data [32], this phenomenon does not exist in the train-ground communication system. For the IEEE 802.11 protocol, in the wireless train-ground data set, the management frames and control frames are extracted by the source addresses and destination addresses, which forms many sessions. Then, every session is divided into sub-sessions by sliding time window sized 4. The sliding time window sized 4 has a good description effect for the IEEE 802.11 protocol [9]. Because the amount of types and subtypes is 30 in the management frames and control frames and the size of sliding time window is 4, there may be $30^4 = 810,000$ kinds of n-grams in mathematics. However, because the communication frames are only closely related to scanning, authentication, and association, many n-grams do not appear and the number of n-grams is small actually. Specifically, in the train-ground communication system, there are about 800 n-grams. This fact has been verified in [9].

Further, the CBTC system uses the wireless communication protocol to transmit control commands. The control and business logic is fixed and the control commands are periodic,

TABLE 4. The quantity information statistics of frames.

Category	Frame Number
Normal behaviors	290844
Abnormal behaviors in the authentication and association attack	447195 (where 335397 for training and 111798 for testing)
Abnormal behaviors in the beacon attack	554511 (where 415884 for training and 138627 for testing)
Abnormal behaviors in the CTS attack	2116047 (where 1587036 for training and 529011 for testing)
Abnormal behaviors in the deauthentication and disassociation attack	255522 (where 191640 for training and 63882 for testing)
Abnormal behaviors in the RTS attack	842301 (where 631725 for training and 210575 for testing)

which makes the number of n-grams reduce. In addition, compared with the general wireless communication protocol used in the office building, the number of trains is limited within the wireless coverage of an AP. In a word, because of the characteristics of the train-ground communication system, the number of n-grams is not big.

In the learning process of typical feature sets, based on the expert experience, a part of the typical features are given and viewed as the first round of candidate feature set. Further, a new feature is added to the first round of candidate feature set and they constitute the second round of candidate feature set. In this paper, a wrapped feature selection method is used, which uses the classifier performance as the evaluation criterion of the selected features. Specifically, the similarity measure of the weak classifier is viewed as an evaluation index. With the increase of the selected features, when the performance of the classifier is no longer significantly improved, the feature selection iteration stops. Furthermore, to ensure the diversity of weak classifiers, four-fifths of features are randomly selected from every typical feature set in the process of constructing each weak classifier.

Because the typical data set has 6 types, according to Algorithm 4, 15 weak classification weight coefficient matrices are given

$$W_i = \begin{pmatrix} 0.8 & 0.05 \\ 0.8 & 0.5 \\ 0.9 & 0.05 \\ 0.9 & 0.5 \end{pmatrix}, \quad i = 1, 2, \cdots, 15$$

2) RESULTS VERIFICATION

In this experiment, four statistical indicators are adopted to evaluate the detection performance: the detection rate, the false positive rate, the false negative rate and the attack identification error rate (AIER). Specifically, in a detection period, S denotes the number of elements and M denotes the number of the data types. The detection rate is defined as follows:

$$\gamma = \frac{r}{S} \tag{9}$$

where r represents the number of those elements which are correctly detected in the data set. The false positive rate is defined as follows:

$$\epsilon^{+} = \frac{p}{N} \tag{10}$$

where N represents the number of those elements which are normal behaviors in the data set; p represents the number of those elements which are normal behaviors and are wrongly identified as attack behaviors in the data set. The false negative rate is defined as follows:

$$\epsilon^{-} = \frac{n}{A} \tag{11}$$

where A represents the number of those elements which are attack behaviors in the data set; n represents the number of those elements which are attack behaviors and are wrongly identified as normal behaviors in the data set. The AIER is defined as follows:

$$\delta = \frac{\mu}{A} \tag{12}$$

where the definition of A can be found in Eq. (11) and μ represents the number of those elements which are attack behaviors but are not correctly identified in the data set.

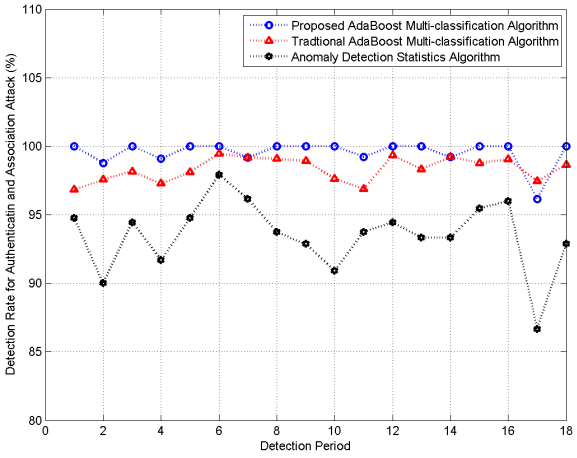

FIGURE 4. Detection rate for authentication and association attack.

In this experiment, the proposed improved AdaBoost multi-classification algorithm is verified. The anomaly detection statistics algorithm of reference [9] and the traditional AdaBoost multi-classification algorithm are used as comparative experiments. The simulation results can be found in Fig. 4-Fig. 23.

Fig. 4-Fig. 7 show the detection results of the authentication and the association attacks. Fig. 4 reveals that the detection rate of the proposed AdaBoost multi-classification algorithm is better than the traditional AdaBoost multi-classification algorithm and the anomaly detection statistics algorithm in all the detection periods except the 17th detection period. The detection rate of the proposed AdaBoost multi-classification algorithm is greater than 98.78%, except in the 17th detection period. In the 17th detection period, the detection rate is 96.15%. The detection rate of the traditional AdaBoost multi-classification algorithm is greater than 96.83%. The detection rate of the anomaly detection statistics algorithm is greater than 86.67%. The detection rate of the proposed AdaBoost multi-classification algorithm is similar to that of the traditional AdaBoost multi-classification algorithm and is significantly different from that of the anomaly detection statistics algorithm. The reason is that the anomaly detection statistics algorithm only detects the normal behaviors and abnormal behaviors, and cannot classify the abnormal behaviors. Fig. 5 shows that the false positive rate of the proposed AdaBoost multi-classification algorithm is lower than 3.448% and the false positive rate of the traditional AdaBoost multi-classification algorithm is lower than 3.774% in the authentication and the association attacks.

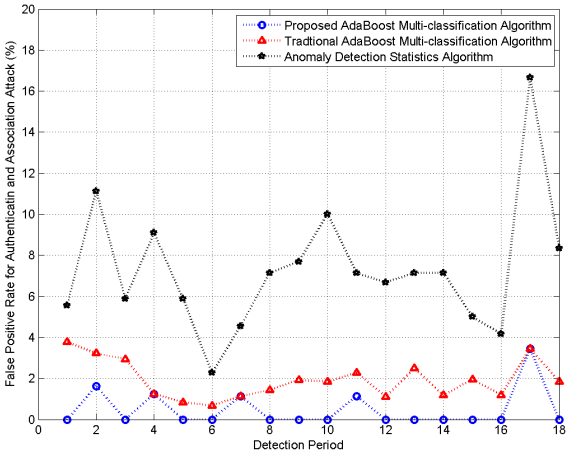

FIGURE 5. False positive rate for authentication and association attack.

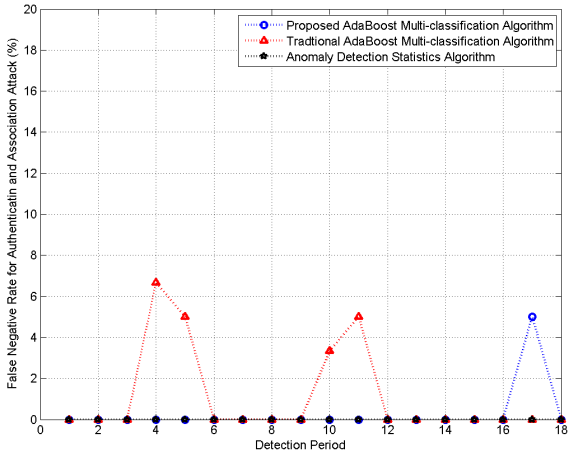

FIGURE 6. False negative rate for authentication and association attack.

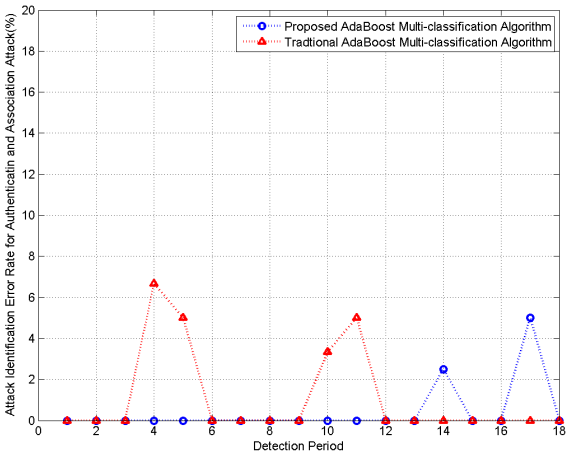

FIGURE 7. Attack identification error rate for authentication and association attack.

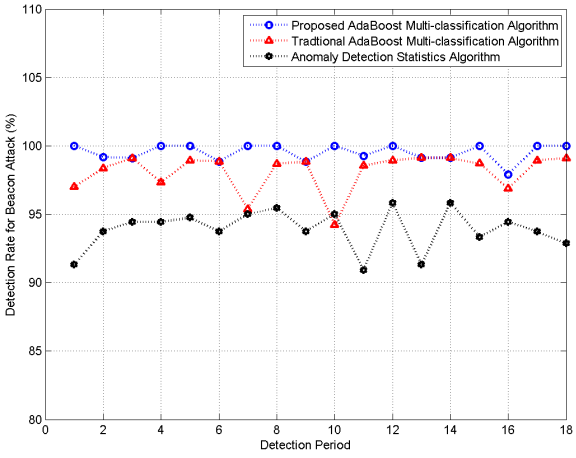

FIGURE 8. Detection rate for beacon attack.

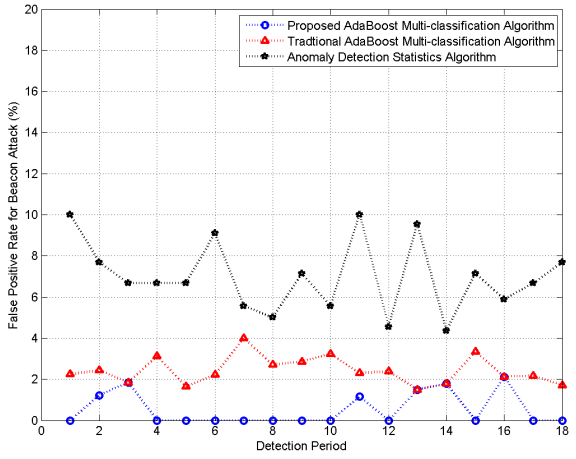

FIGURE 9. False positive rate for beacon attack.

The maximum false positive rate of the anomaly detection statistics algorithm is 16.67% and the minimum false positive rate of the anomaly detection statistics algorithm is 2.273%. The false positive rate of the anomaly detection statistics algorithm is obviously worse than that of the proposed AdaBoost multi-classification algorithm and the traditional AdaBoost multi-classification algorithm. Further, in Fig. 6, the proposed AdaBoost multi-classification algorithm has a lower false negative rate than the traditional AdaBoost multi-classification algorithm in all the detection periods except the 17th detection period. The false negative rate of the anomaly detection statistics algorithm is 0, because this algorithm only models the normal behavior and any behavior being different from the normal behavior is viewed as the abnormal behavior. Fig. 7 shows that the maximum AIER of the proposed AdaBoost multi-classification algorithm is 5% and the maximum AIER of the traditional AdaBoost multi-classification algorithm is 6.667%.

The detection results of the beacon attacks are described in Fig. 8-Fig. 11. Fig. 8 reveals that the proposed AdaBoost multi-classification algorithm has a higher detection rate than the traditional AdaBoost multi-classification algorithm and the anomaly detection statistics algorithm. The detection rate of the proposed AdaBoost multi-classification

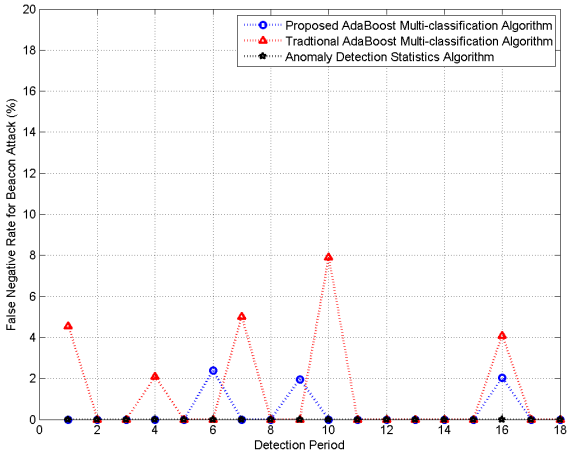

FIGURE 10. False negative rate for beacon attack.

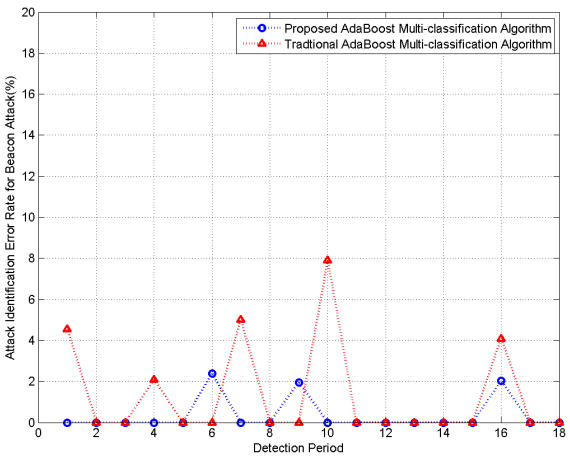

FIGURE 11. Attack identification error rate for beacon attack.

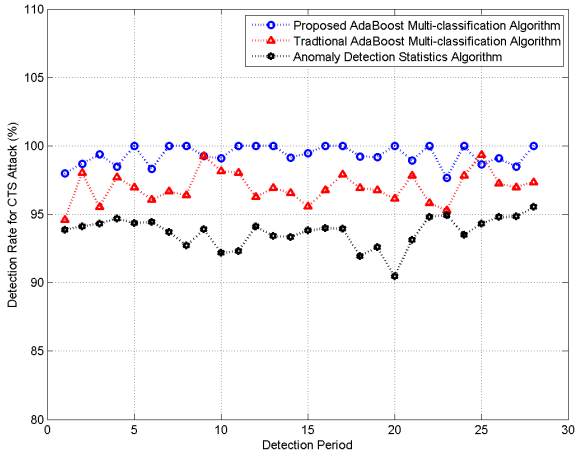

FIGURE 12. Detection rate for CTS attack.

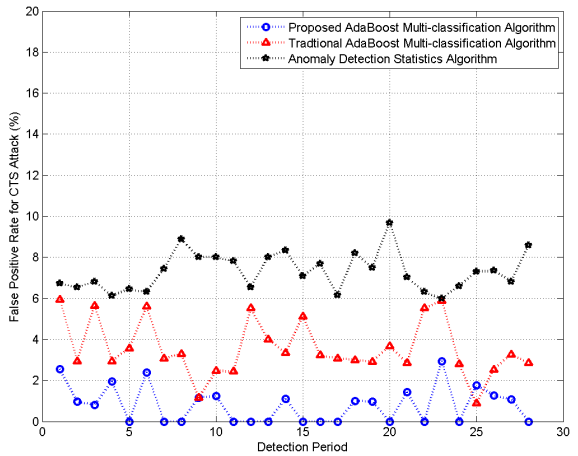

FIGURE 13. False positive rate for CTS attack.

algorithm is greater than 97.92%. The detection rate of the traditional AdaBoost multi-classification algorithm is greater than 94.20%. The detection rate of the anomaly detection statistics algorithm is greater than 90.91%. Fig. 9 shows that the false positive rate of the proposed AdaBoost multi-classification algorithm is lower than 2.128% and the false positive rate of the traditional AdaBoost multi-classification algorithm is lower than 4%. The maximum false positive rate of the anomaly detection statistics algorithm is 10% and the minimum false positive rate of the anomaly detection statistics algorithm is 4.348%. Therefore, it is concluded that the proposed algorithm has a lower false positive rate than the traditional AdaBoost multi-classification algorithm and the anomaly detection statistics algorithm. The false positive rate of the anomaly detection statistics algorithm is obviously worse than that of the proposed AdaBoost multi-classification algorithm and the traditional AdaBoost multi-classification algorithm. In Fig.10, the proposed AdaBoost multi-classification algorithm has a lower false negative rate than the traditional AdaBoost multi-classification algorithm in all the detection periods except the 6th and 9th detection periods. The false negative rate of the anomaly detection statistics algorithm is 0. Because this algorithm belongs to anomaly detection which only models the normal behavior, any behavior being different from the normal behavior can be viewed as the abnormal behavior. Fig. 11 shows that the maximum AIER of the proposed AdaBoost multi-classification algorithm is 2.381% and the maximum AIER of the traditional AdaBoost multi-classification algorithm is 7.895%.

The detection results of CTS attacks are given in Fig. 12 -Fig. 15. In Fig. 12, it is found that the proposed AdaBoost multi-classification algorithm has a higher detection accuracy than the traditional AdaBoost multi-classification algorithm and the anomaly detection statistics algorithm in all the detection periods except the 25th detection period. The minimum detection rate of the proposed AdaBoost multi-classification algorithm is 97.65% and the maximum detection rate is 100%. The minimum detection rate of the traditional AdaBoost multi-classification algorithm is 94.59% and the maximum detection rate is 99.33%. The minimum detection rate of the anomaly detection statistics algorithm is 90.48%

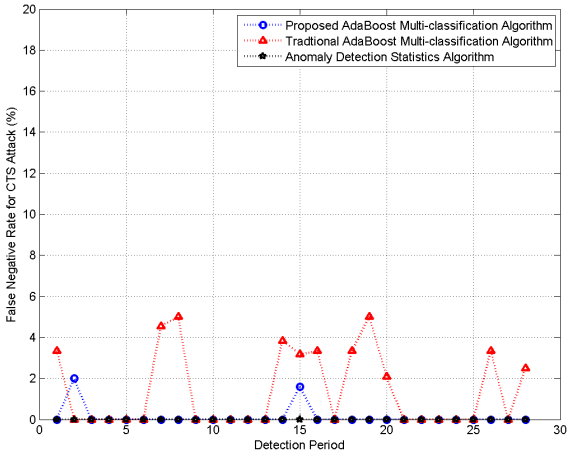

FIGURE 14. False negative rate for CTS attack.

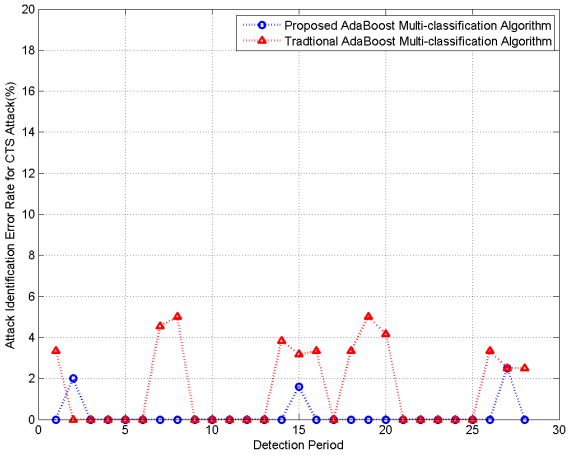

FIGURE 15. Attack identification error rate for CTS attack.

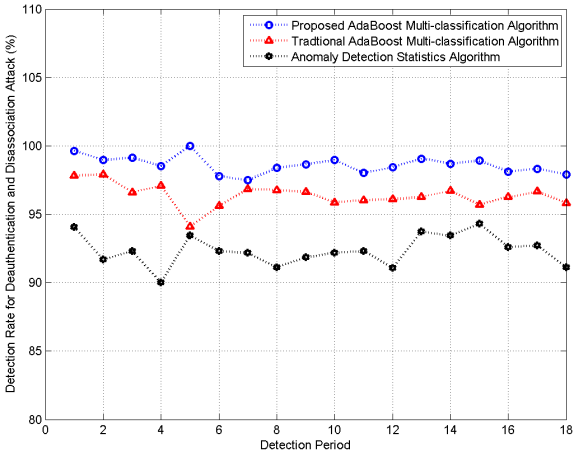

FIGURE 16. Detection rate for deauthentication and disassociation attack.

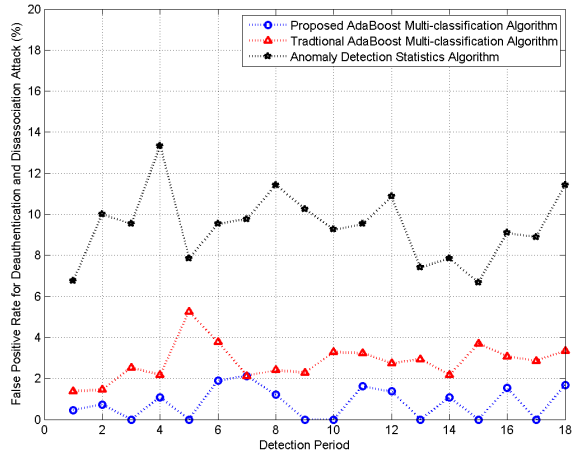

FIGURE 17. False positive rate for deauthentication and disassociation attack.

and the maximum detection rate of that is 95.52%. Fig. 13 shows that the proposed AdaBoost multi-classification algorithm has a lower false positive rate than the traditional AdaBoost multi-classification algorithm and the anomaly detection statistics algorithm in all the detection periods except the 25th detection period. Specifically, the maximum false positive rates of the anomaly detection statistics algorithm, the traditional AdaBoost multi-classification algorithm and the proposed AdaBoost multi-classification algorithm are 9.677%, 5.932% and 2.941% respectively. The minimum false positive rates of the anomaly detection statistics algorithm, the traditional AdaBoost multi-classification algorithm and the proposed AdaBoost multi-classification algorithm are 5.983%, 0.8772% and 0% respectively. In Fig. 14, the results reveal that the proposed AdaBoost multi-classification algorithm has a lower false negative rate than the traditional AdaBoost multi-classification algorithm in all the detection periods except the second detection period and the false negative rate of the anomaly detection statistics algorithm is 0. Fig. 15 shows that the maximum AIER of the proposed AdaBoost multi-classification algorithm is 2.5% and the maximum AIER of the traditional AdaBoost multi-classification algorithm is 5%.

The detection results of the disassociation and deauthentication attacks are shown in Fig. 16-Fig. 19. Fig. 16 reveals that the proposed AdaBoost multi-classification algorithm has a higher detection accuracy than the other two algorithms. Specifically, the minimum detection rate is 97.48% in the proposed AdaBoost multi-classification algorithm, it is 94.12% in the traditional AdaBoost multi-classification algorithm and it is 90% in the anomaly detection statistics algorithm. According to Fig. 17, it is found that the proposed AdaBoost multi-classification algorithm has a lower false positive rate than the other two algorithms. Specifically, the maximum false positive rate of the anomaly detection statistics algorithm is 13.33% and the minimum false positive rate of the anomaly detection statistics algorithm is 6.667%. The maximum false positive rate of the proposed AdaBoost multi-classification algorithm is 2.128% and the minimum false positive rate of the proposed AdaBoost multi-classification algorithm is 0%. The maximum false positive rate of the traditional AdaBoost multi-classification

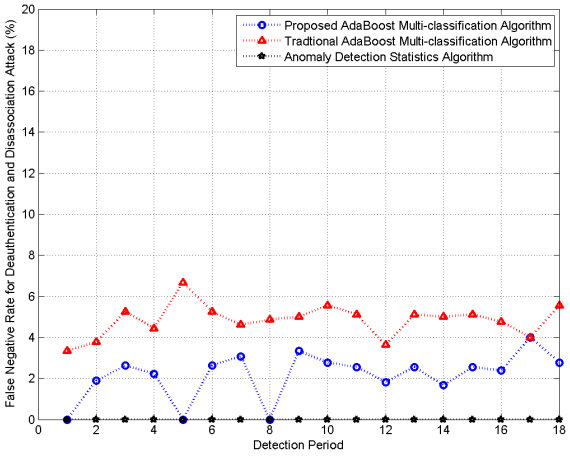

FIGURE 18. False negative rate for deauthentication and disassociation attack.

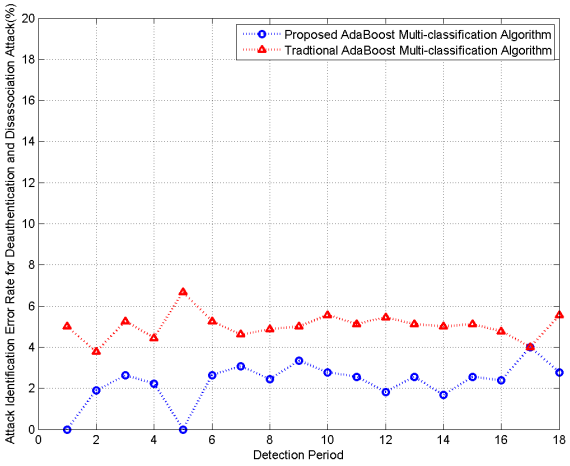

FIGURE 19. Attack identification error rate for deauthentication and disassociation attack.

algorithm is 5.263% and the minimum false positive rate of the traditional AdaBoost multi-classification algorithm is 1.376%. In Fig. 18, the proposed AdaBoost multi-classification algorithm has a lower false negative rate than the traditional AdaBoost multi-classification algorithm. The maximum false negative rate of the proposed AdaBoost multi-classification algorithm is 4% and the minimum false negative rate of the proposed AdaBoost multi-classification algorithm is 0%. The maximum false negative rate of the traditional AdaBoost multi-classification algorithm is 6.667% and the minimum false negative rates of the traditional AdaBoost multi-classification algorithm is 3.333%. The false negative rate of the anomaly detection statistics algorithm is 0. Because this algorithm only models the normal behavior and any behavior being different from the normal behavior is viewed as the abnormal behavior. Fig. 19 shows that the maximum AIER of the proposed AdaBoost multi-classification algorithm is 4% and the maximum AIER of the traditional AdaBoost multi-classification algorithm is 6.667%.

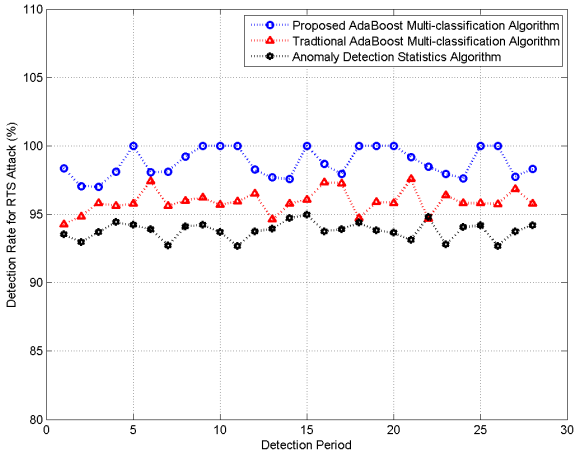

FIGURE 20. Detection rate for RTS attack.

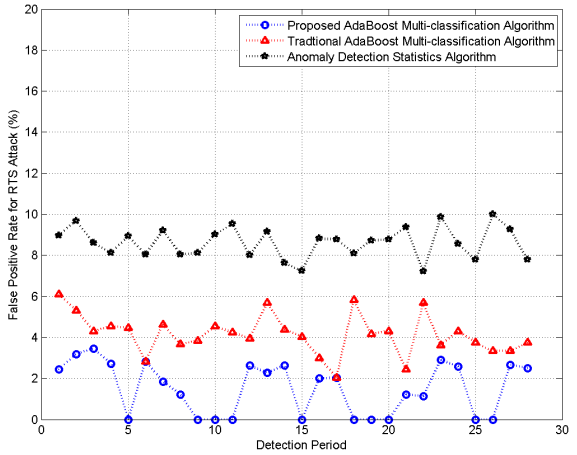

FIGURE 21. False positive rate for RTS attack.

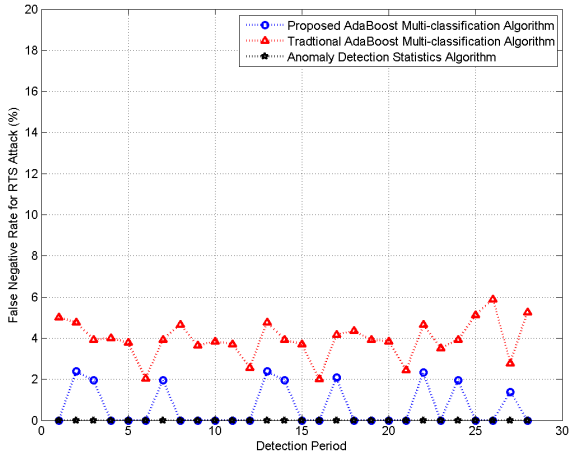

FIGURE 22. False negative rate for RTS attack.

Fig. 20 -Fig. 23 show the detection results of RTS attacks. In Fig. 20, it is revealed that the minimum detection rate is 97.01% and the maximum detection rate is 100% in the proposed AdaBoost multi-classification algorithm. The minimum detection rate of the traditional AdaBoost

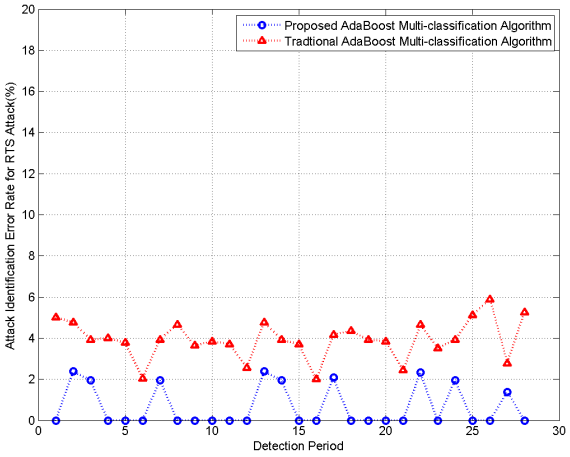

FIGURE 23. Attack identification error rate for RTS attack.

multi-classification algorithm is 94.26% and the maximum detection rate is 97.56%. The minimum detection rate of the anomaly detection statistics algorithm is 92.66% and the maximum detection rate is 94.95%. In Fig. 21, the maximum false positive rate of the anomaly detection statistics algorithm is 10% and the minimum false positive rate of the anomaly detection statistics algorithm is 7.207%. The maximum false positive rate of the proposed AdaBoost multi-classification algorithm is 3.448% and the minimum false positive rate of the proposed AdaBoost multi-classification algorithm is 0%. The maximum false positive rate of the traditional AdaBoost multi-classification algorithm is 6.098% and the minimum false positive rate of the traditional AdaBoost multi-classification algorithm is 2.041%. Furthermore, in Fig. 22, compared with the traditional AdaBoost multi-classification algorithm, the proposed AdaBoost multi-classification algorithm has a lower false negative rate. The maximum false negative rate of the proposed AdaBoost multi-classification algorithm is 2.381% and the minimum false negative rate of the proposed AdaBoost multi-classification algorithm is 0%. The maximum false negative rate of the traditional AdaBoost multi-classification algorithm is 5.882% and the minimum false negative rates of the traditional AdaBoost multi-classification algorithm is 2%. The false negative rate of the anomaly detection statistics algorithm is 0. Fig. 23 shows that the maximum AIER of proposed AdaBoost multi-classification algorithm is 2.381% and the maximum AIER of the traditional AdaBoost multi-classification algorithm is 5.882%.

In summary, experiment results show that the proposed method has a better detection performance for the above-mentioned DoS attacks, compared with the traditional AdaBoost multi-classification algorithm and the anomaly detection statistics algorithm.

V. CONCLUSION

This paper researches the intrusion detection issue of the train-ground communication system in urban rail transit. In the MAC layer, an intrusion detection method based on the n-gram model is proposed by analyzing the IEEE 802.11 protocol. Firstly, the n-gram model of the data is established. Secondly, based on the n-gram model, a weak classifier construction method is given by the proposed similarity measure algorithm. Finally, an AdaBoost intrusion detection method is presented. Simultaneously, the wired intrusion detection is carried out by detecting the train states such as the train position and velocity. An comprehensive detection conclusion can be obtained by integrating the wireless and wired detection results. The proposed intrusion detection method can not only detect the attacks with a higher detection rate, but also can identify the attack types.

REFERENCES

[1] *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, standard 802.11-1997, Nov. 1997, pp. 1–445.
[2] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment6: Medium Access Control (MAC) Security Enhancements*, IEEE Standard 802.11, 2004. [Online]. Available: https://ci.nii.ac.jp/naid/10031092440/en/
[3] *IEEE 802.11: Wireless LANs*. Accessed: Nov. 21, 2011. [Online]. Available: http://standards.ieee.org/about/get/802/802.11.html
[4] X. Wang, L. Liu, L. Zhu, and T. Tang, "Joint security and QoS provisioning in train-centric CBTC systems under sybil attacks," *IEEE Access*, vol. 7, pp. 91169–91182, 2019.
[5] Y. P. Zhang, "Novel model for propagation loss prediction in tunnels," *IEEE Trans. Veh. Technol.*, vol. 52, no. 5, pp. 1308–1314, Sep. 2003.
[6] K. Guan, Z. Zhong, J. I. Alonso, and C. Briso-Rodriguez, "Measurement of distributed antenna systems at 2.4 GHz in a realistic subway tunnel environment," *IEEE Trans. Veh. Technol.*, vol. 61, no. 2, pp. 834–837, Feb. 2012.
[7] S. Lin, Z. Zhong, L. Cai, and Y. Luo, "Finite state Markov modelling for high speed railway wireless communication channel," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2012, pp. 5421–5426.
[8] H. Wang, F. R. Yu, L. Zhu, T. Tang, and B. Ning, "Finite-state Markov modeling for wireless channels in tunnel communication-based train control systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 3, pp. 1083–1090, Jun. 2014.
[9] H. Alipour, Y. B. Al-Nashif, P. Satam, and S. Hariri, "Wireless anomaly detection based on IEEE 802.11 behavior analysis," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 10, pp. 2158–2170, Oct. 2015.
[10] C. Ioannou, V. Vassiliou, and C. Sergiou, "An intrusion detection system for wireless sensor networks," in *Proc. 24th Int. Conf. Telecommun. (ICT)*, May 2017, pp. 1–5.
[11] Z. Sun, Y. Xu, G. Liang, and Z. Zhou, "An intrusion detection model for wireless sensor networks with an improved V-detector algorithm," *IEEE Sensors J.*, vol. 18, no. 5, pp. 1971–1984, Dec. 2018.
[12] M. Usha and P. Kavitha, "Anomaly based intrusion detection for 802.11 networks with optimal features using SVM classifier," *Wireless Netw.*, vol. 23, no. 8, pp. 2431–2446, 2017.
[13] E. Shams and A. Rizaner, "A novel support vector machine based intrusion detection system for mobile ad hoc networks," *Wireless Netw.*, vol. 24, no. 5, pp. 1821–1829, Jul. 2018.
[14] M. Faisal, S. Abbas, and H. U. Rahman, "Identity attack detection system for 802.11-based ad hoc networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2018, no. 1, p. 128, 2018.
[15] X. Cao, L. Liu, W. Shen, A. Laha, J. Tang, and Y. P. P. Cheng, "Real-time misbehavior detection and mitigation in cyber-physical systems over WLANs," *IEEE Trans. Ind. Informat.*, vol. 13, no. 1, pp. 186–197, Feb. 2017.
[16] R. Abdulhammed, M. Faezipour, A. Abuzneid, and A. Alessa, "Enhancing wireless intrusion detection using machine learning classification with reduced attribute sets," in *Proc. 14th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jun. 2018, pp. 524–529.
[17] Y. Yao, B. Xiao, G. Wu, X. Liu, Z. Yu, K. Zhang, and X. Zhou, "Multi-channel based Sybil attack detection in vehicular ad hoc networks using RSSI," *IEEE Trans. Mobile Comput.*, vol. 18, no. 2, pp. 362–375, Feb. 2019.

[18] T. OConnor and D. Reeves, "Bluetooth network-based misuse detection," in *Proc. Annu. Comput. Secur. Appl. Conf. (ACSAC)*, Dec. 2008, pp. 377–391.
[19] I. Onat and A. Miri, "An intrusion detection system for wireless sensor networks," in *Proc. IEEE Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, vol. 3, Aug. 2005, pp. 253–259.
[20] Z. Teo, B. A. N. Tran, S. Lakshminarayana, W. G. Temple, B. Chen, R. Tan, and D. K. Y. Yau, "Securerails: Towards an open simulation platform for analyzing cyber-physical attacks in railways," in *Proc. IEEE Region Conf. (TENCON)*, Nov. 2016, pp. 95–98.
[21] Y. Wu, J. Weng, Z. Tang, X. Li, and R. H. Deng, "Vulnerabilities, attacks, and countermeasures in balise-based train control systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 4, pp. 814–823, Apr. 2017.
[22] A. A. Cardenas, S. Amin, and S. Sastry, "Secure control: Towards survivable cyber-physical systems," in *Proc. 28th Int. Conf. Distrib. Comput. Syst. Workshops*, Jun. 2008, pp. 495–500.
[23] D. Jurafsky and J. Martin, "Language modeling with N grams," in *Proc. Speech Lang. Process.*, 2017.
[24] X. Wang, L. Liu, L. Zhu, and T. Tang, "Train-centric CBTC meets age of information in train-to-train communications," *IEEE Trans. Intell. Transp. Syst.*, to be published.
[25] A. Syropoulos, "Mathematics of multisets," in *Workshop Membrane Computing*. Berlin, Germany: Springer, 2000, pp. 347–358.
[26] X. Wang, L. Liu, T. Tang, and W. Sun, "Enhancing communication-based train control systems through train-to-train communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 4, pp. 1544–1561, Apr. 2019.
[27] (2007). *KDDCup1999*. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/KDDCUP99
[28] C. Kolias, G. Kambourakis, A. Stavrou, and S. Gritzalis, "Intrusion detection in 802.11 networks: Empirical evaluation of threats and a public dataset," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 184–208, 1st Quart., 2015.
[29] S. Barua, M. M. Islam, X. Yao, and K. Murase, "MWMOTE–majority weighted minority oversampling technique for imbalanced data set learning," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 2, pp. 405–425, Feb. 2013.
[30] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, 2002.
[31] E. Ramentol, Y. Caballero, R. Bello, and F. Herrera, "SMOTE-RSB*: A hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using SMOTE and rough sets theory," *Knowl. Inf. Syst.*, vol. 33, no. 2, pp. 245–265, 2012.
[32] C. Wressnegger, G. Schwenk, D. Arp, and K. Rieck, "A close look on *n*-grams in intrusion detection: Anomaly detection vs. classification," in *Proc. ACM Workshop Artif. Intell. Secur.*, 2013, pp. 67–76.

BING GAO received the B.S. degree in electronic and information engineering from Xi'an Technological University, Shaanxi, China, in 2009, and the M.S. degree in software engineering from Beijing Normal University, Beijing, China, in 2014. She is currently pursuing the Ph.D. degree in traffic information engineering and control with Beijing Jiaotong University, Beijing. Her research interests include train communication security, train operation control, train-ground, and train-to-train communication technology.

BING BU received the Ph.D. degree from Beijing Jiaotong University, Beijing, China, in 2001.

From 2001 to 2002, he participated in the research and development of time-division synchronous code division multiple access in Siemens Ltd. From 2002 to 2007, as a high delegate of Samsung, he took part in the standardization works of EUCH and Long Term Evolution. From 2007 to 2011, he was an Associate Professor with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University. During that time, as a Key Member, he participated in the research and development of the first home-made communications-based train control (CBTC) system of China. He was in charge of the design of the data communication system in CBTC, which has been successfully used in several business operating subway lines in China, such as Beijing Yizhuang Line, Beijing Changping Line, Chongqing No.3 Line. From October 2011 to October 2012, he was a Visiting Scholar with Carleton University, Ottawa, ON, Canada. He is currently a Professor and the Ph.D. Supervisor with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University. His research interests include train communication security, the theories and techniques for the improvement of train-to-ground wireless communications, and methods for the optimization of train control in CBTC with unreliable wireless networks.