Research on Random Intrusion Depth Detection of Internet of Things Based on 3D Convolutional Neural Network

Xingfei Ma[1(✉)] and Wuguang Wang[2]

[1] Wuxi Vocational Institute of Commerce, Wuxi 214153, China
maxingfei6@yeah.net
[2] WuXi City College of Vocational Technology, Wuxi 214153, China

Abstract. There are many problems in the industrial Internet of Things, such as low feature extraction rate, low detection efficiency and poor adaptability. To solve this problem, a random intrusion depth detection method based on three-dimensional convolution neural network is proposed. According to NIDS, an intrusion detection model of the Internet of Things is built, through which distributed network data packets are collected, and the principal component analysis algorithm is used to preprocess them to reduce data dimensions. Combined with deep learning theory and technology, select data features to form feature matrix. With this as the input, the random intrusion detection in the Internet of Things is completed by using 3D convolution neural network (3DCNN) combined with long and short memory (LSTM) method. The experimental results show that the F1 value of the detection method is above 0.9, indicating that the detection accuracy of the method is high.

Keywords: Three Dimensional Convolution Neural Network · Internet of Things · Random Intrusion · Depth Detection Method

1 Introduction

With the deepening of IOT application, the complex network environment and endless attacks make it face many challenges, such as hacker intrusion, security vulnerability attacks, worms and so on. IOT can be divided into three layers: perceptual layer, network layer and application layer. The perceptual layer mainly focuses on data security, such as preventing malicious node attack, sampling and data forgery. Network layer security is represented by preventing Dos attacks and ensuring routing security [1].

Application layer security can satisfy user privacy and access control. At present, most of the security mechanisms for the Internet of Things tend to be passive, and

2023 Philosophy and Social Science Research Funds for Colleges and Universities in Jiangsu Province【Research on Innovation Talent Cultivation Path of Industry-Education Integration in Higher Vocational Colleges under the Perspective of Talent Chain, Industry Chain, Innovation Chain and Value Chain】(Project No. 2023SJYB0979).

B. Wang et al. (Eds.): ICMTEL 2023, LNICST 532, pp. 262–276, 2024.
https://doi.org/10.1007/978-3-031-50571-3_19

Intrusion Detection (ID) can monitor the transmitted data of the network in real time and take measures to monitor, analyze and warn the intrusion so as to improve the network's ability to cope with external threats. Traditional intrusion detection research is not perfect, there are still the following problems: the Internet of Things environment is complex, the network traffic data collected is high-dimensional. With the change of the operating environment and structure of the Internet of Things, it is necessary to constantly update the model to detect the new unknown attacks. Based on the above background, a random intrusion depth detection method of Internet of Things based on three-dimensional convolution neural network is proposed. Build an intrusion detection model of the Internet of Things through NIDS to collect distributed network data packets. In order to reduce the data dimension, the principal component analysis algorithm is used to preprocess it. Using 3DCNN and LSTM method, the accuracy of intrusion detection is enhanced, so as to realize random intrusion detection of the Internet of Things.

2 Design of Intrusion Detection Model for Internet of Things

Network intrusion detection is the process of discovering the behavior of unauthorized user using or attempting to use computer system and the behavior of legal user abusing its privilege. Network intrusion detection can be divided into misuse intrusion detection and anomaly intrusion detection. Misuse intrusion detection, also known as knowledge-based or feature-based intrusion detection, extracts the pattern features of various attacks and forms a rule base. However, misuse intrusion detection can only identify known attacks, so there is a certain rate of under-reporting, and its rule base can be continuously expanded by self-learning to detect new attacks and reduce under-reporting rate [2]. Anomalous intrusion detection, also known as statistical or behavior-based Intrusion detection, is used to identify abnormal behaviors that differ greatly from normal activities in the host computer or network. Anomaly detection first collects historical data on normal operational behaviors within a certain period of time, in order to establish a normal behavior profile on behalf of the user, host computer or network connection. Then it collects event data and uses neural network, genetic algorithm or traditional statistical analysis to determine whether the current behaviors are abnormal behaviors by comparing with normal behavior patterns; According to the distribution of data collection, analysis and response, it can be divided into centralized network intrusion detection and distributed network intrusion detection.

Centralized intrusion detection adopts a single host to analyze its audit data or network traffic, and look for possible intrusion behavior; because of adopting the method of centralized processing, the host that realizes the intrusion detection function will become the bottleneck of the system; on the one hand, the performance of the system is affected by undertaking too much work; on the other hand, the host is often the primary target of attack, and once it is broken, the security of the system cannot be guaranteed.

Distributed network intrusion detection adopts multiple agents distributed in each part of the network to carry out intrusion detection respectively, and can deal with possible intrusion behaviors cooperatively; this method distributes the detection agents in the interested or important position on the network, independently and autonomously runs, collects intrusion information, and independently or cooperatively detects network intrusion behaviors; this intrusion detection method realizes the function and security

decentralization, solves the problem of single point failure, or restricts it within a certain scope, and does not seriously affect the security performance of the system.

2.1 Network Packet Collection

Generally speaking, intrusion detection systems can be divided into host type and network type. Host intrusion detection systems (HIDS) often use system logs, application logs, etc. as data sources, and of course, other means (such as monitoring system calls) can be used to collect information from the host for analysis. The data source of the Network Intrusion Detection System (NIDS) is the data packet on the network, and the network card of the network intrusion detection engine can be set in mixed mode to monitor and judge all the data packets within the network segment.

When installing NIDS, the key is to choose the location of the data acquisition section, because it determines the visibility of the "event", and the data acquisition section has many possibilities: (1) If the network segment is connected by a bus hub, it can be simply connected to one port of the hub; and (2) For a switched Ethernet switch, the problem becomes complicated. Because the switch does not use a shared media approach, the traditional use of a sniffer to listen on the entire subnet is no longer feasible, and NIDS can be deployed on the switch's listening port to obtain all data flow through the switch. NIDS consists of the following elements: agents, forwarders, monitors, filters, and user interfaces, the structure of which is shown in Fig. 1.

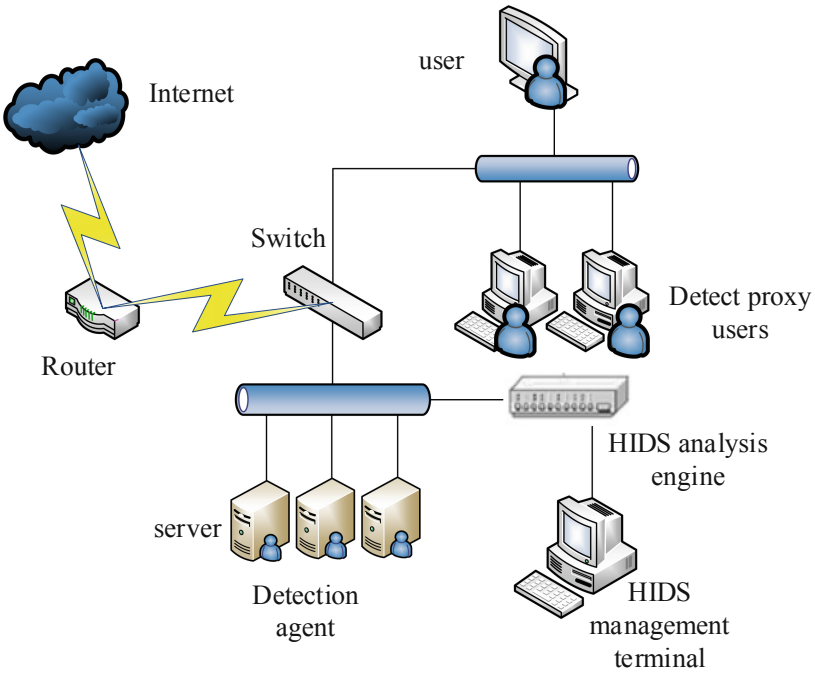

Fig. 1. Composition of NIDS

An agent is an independent entity responsible for monitoring some aspect of a host that reports the information it collects to the forwarder. For example, a proxy can monitor many connections to a host and collect suspicious information. The agent reports Telnet findings to the specified forwarder, but it does not have the authority to generate an alert directly. Usually the forwarder or monitor generates an alarm based on the information sent by the agent. Reports sent by forwarders through different agents provide a complete picture of the host's status, and reports sent by monitors through different forwarders provide a complete picture of the monitored network [3].

Filters can be seen as a hierarchy of data selection and abstraction for proxies. When the agent collects host information, it encounters two problems: (1) there may be multiple agents in a system using the same data source, such as using the same Unix log file in, and each agent processing the data source means repeated effort; and (2) there may be one agent that will be used in different Uni versions, but different Unix versions may have different locations and file formats that are not the same in Unix, which means writing several agents to accommodate different versions. The filter is proposed to solve these two problems. The proxy puts a condition on the filter, and the filter returns records that match the condition. Filters will also address issues related to system versions, enabling agents that implement uniform functionality to work with different operating systems.

The transponder is responsible for collecting the information provided by each agent on the host and communicating with the outside world. Each host under distributed monitoring has a forwarding NIDS that controls the work of the agent, processes the data from the agent, and responds to requests from the monitor.

The monitor controls the work of the transponder and processes the data it sends. The biggest difference between a monitor and a forwarder is that the monitor controls the forwarders on different hosts and the forwarder controls the agents on a single host. The monitor communicates with the user interface, thus providing a managed entry point to the entire distribution. The NIDS user interface provides users with visual graphics to manage and communicate user instructions to the monitor.

Network packet collection unit is the basic component of NIDS. Generally, by intercepting all the traffic of the whole network, it simply filters out the unconcerned data according to the information of source host, destination host and service protocol port, and then sends the interested data to the higher application for analysis. On the one hand, the network packet collection module should be able to collect all the packets on the network, especially to detect the fragmented packets (which may contain attacks), on the other hand, the efficiency of data interception module to intercept the packets is also very important, which directly affects the speed of the whole NIDS and the adaptability of NIDS to the modern high-speed network [4].

In general network environment, we can use the API interface provided by the system directly to collect data packets. This method is very simple, easy to implement, but its function is limited and inefficient. At present, many software use libpcap library to collect network packets, and it can filter the data to reduce the data that need analysis.

2.2 Packet Preprocessing

In general network environment, we can use the API interface provided by the system directly to collect data packets. This method is very simple, easy to implement, but

its function is limited and inefficient. At present, many software use libpcap library to collect network packets, and it can filter the data to reduce the data that need analysis (Fig. 2).

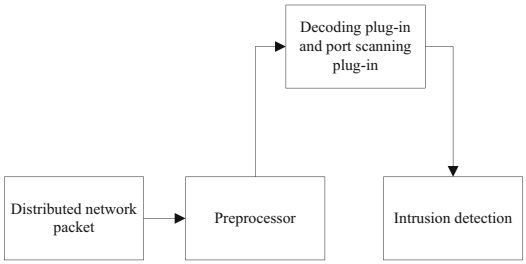

Fig. 2. Structure Diagram of Data Packet Preprocessing

(1) Data dimension reduction

The basic idea of principal component analysis is to construct a series of linear combinations of the original variables to form several comprehensive geometric indexes to remove the relevance of the data, so as to keep the variance information of the original high-dimensional data to the greatest extent. Principal Component Analysis (PCA) algorithm is a linear dimensionality reduction method. Combining MIS and PCA, this paper puts forward a new algorithm based on MIS and PCA, called MIS-PCA algorithm. Firstly, the algorithm transforms the data set into matrix, calculates the mutual information value (MI) of attributes, other mutual information value (LMessI), maximum mutual information value (MaxMI), then calculates the absolute mutual information reliability and relative mutual information reliability according to the correlation of MI, LessMi and MaxMI, and obtains the mutual information comprehensive reliability according to the absolute mutual information reliability and relative mutual information reliability. Finally, the matrix is screened by using the mutual information comprehensive reliability, and the dimension is reduced by principal component analysis [5].

The principle of MIS-PCA based on mutual information comprehensive credibility is the same as that of PCA, but the difference is only that the PCA algorithm is applied to do a feature screening before dimension reduction; the threshold H is set to filter out the feature attributes of raw data information that is almost useless. The specific steps of the MIS-PCA algorithm are as follows:

Input: data matrix A_{nm}, m represents the number of samples, n represents the number of attributes; Mutual information comprehensive confidence threshold k, contribution rate S.

Step 1: calculate MI (mutual information value), MIA (absolute mutual information credibility), MIR (relative mutual information credibility) and MIS (mutual information comprehensive credibility) of each attribute, compare MIS with comprehensive mutual information credibility threshold k, and filter features to obtain a new matrix k.

Step 2: use PCA algorithm to reduce the dimension of matrix B. The operations are as follows:

1) Centralize the sample matrix and get matrix C_{nm};
2) Covariance matrix D based on C_{nm};
3) Eigenvalue e_i and eigenvector E_i of covariance matrix.
4) Selected transform base

The maximum M of the eigenvalues is selected, and the corresponding M eigenvectors are used as column vectors to form the eigenvector matrix E_{nM}.

5) Calculate the reduced dimension data matrix $F = EE^T$.
6) Use the PCA algorithm to reduce the dimension of the matrix B.

In this algorithm, the contribution rate is the ratio of the sum of selected eigenvalues to the sum of all eigenvalues, which can generally be expressed by formula (16). In addition, in the algorithm design, B is not used as a parameter input, but actually M is selected according to the contribution rate of eigenvalues.

$$S = \frac{\sum_{i=1}^{M} e_i}{\sum_{i=1}^{n} e_i} \tag{1}$$

where, e_i represents the characteristic value.

The principal component analysis (PCA) method has the characteristics of easy calculation and strong interpretation. The PCA algorithm measures information by the variance of the data: the larger the variance, the more key information it contains, otherwise, the less information it contains. Therefore, PCA is a process that transforms the coordinate projection of high dimensional data into a new coordinate system to represent the data in the direction of maximum variance.

(2) Data standardization

In data analysis, it is very common for data to have units, such as GDP in units of 100 million or millions, then there will be problems with the size of numbers due to unit problems; this situation may have an impact on the analysis and therefore needs to be processed, provided that the relative meaning of the numbers is not lost, that is, the larger the number before represents the higher GDP, the processed data cannot lose this characteristic [6]. To calculate the distance, the digits 1 and 2 can be subtracted directly to the distance value 1; another set of digits 10000 and 20000 can be subtracted directly to the distance value 10000. If the larger the number is, the farther the distance is, then the apparent 10000 is greater than 1, but this is only due to data units, not actual expectations.

In such cases, before data analysis, it is sometimes necessary to standardize the data. Data standardization is to convert the original data into dimensionless indicator evaluation values through a certain mathematical transformation method, that is, all indicator values are at the same quantitative level, so that comprehensive analysis and comparison can be carried out. Standardization is one of the most common dimensional processing methods. The calculation formula is:

$$a_i' = \frac{a_i - b}{c} \tag{2}$$

In the formula, a_i' represents the processed data, a_i represents the pre-processed data, and b and c represent the average and standard deviation of the data.

This processing makes the data show a characteristic that the average value of the data must be 0 and the standard deviation must be 1. The compressed size of the data is processed, and the data has special characteristics (the average standard deviation of 0 is 1).

This kind of processing is used in many research algorithms, for example, it is necessary to standardize before clustering analysis, or it is used by default in factor analysis. In clustering, for example, the underlying algorithm is based on distance to measure clustering, so the default SPSSAU selects for normalization [7]. In addition, there are some special research methods, such as sociological mediation, or moderation studies, may also standardize the data.

2.3 Intrusion Identification Based on 3D CNN

In 2D convolution neural network, convolution operation is applied to 2D feature map, and the convolution process can only learn features from spatial dimension. When used in video analysis problems, the motion information of multiple consecutive frames is captured. Therefore, 3D convolution and 3D pooling are performed during the convolution and pooling phases of CNN to simultaneously compute features from both spatial and temporal dimensions.

3D convolution is achieved by stacking several consecutive frames together in a 3D kernel convolution. Through this structure, the feature mapping in the convolution layer is connected to a plurality of consecutive frames in the previous layer to capture the target information. One 3D convolution core can only extract one type of feature from the frame cube, because kernel weights are copied across the entire cube. The general design principle of CNN is to increase the number of feature maps by generating multiple types of features from the same set of lower level feature maps. Based on 3D convolution operation, we can design different 3DCNN networks for different analysis tasks.

2.3.1 Construction of Data Characteristic Matrix

Feature extraction is an indispensable step in intrusion detection, which is directly related to the efficiency and accuracy of detection, and thus affects the performance of detection model. Characteristics, also known as attributes, describe the characteristics of a thing in one way or another. Usually we describe the characteristics of a transaction. In 3D CNN algorithm, the more information the feature contains, the more advantageous it is to distinguish different things. High-dimensional feature space will lead to data disaster, and the sample size will increase exponentially as the dimension increases. Increase model processing time. Moreover, some features in these high-dimensional data may contain very little or irrelevant information, which has no effect on the classification of machine learning algorithms. Intrusion detection algorithm is a classification problem, which is used to distinguish abnormal data. In the process of designing an intrusion detection algorithm, two problems are usually considered, one is the recognition performance of the model, the other is whether the input data contains valid features. Therefore, the

selection of classification algorithm, data size will lead to the efficiency of the latter model. Therefore, the feature extraction of the original data set is a crucial step.

The degree to which Information Gain (IG) algorithms measure feature h_i based on information entropy and exist to reduce information uncertainty in classification systems. For classification system, the data set of samples is $H = \{h_1, h_2, ..., h_i, ..., h_M\}$, the category set of samples is $G = \{g_1, g_2, ..., g_M\}$, among which $h_i = \{h_i(1), h_i(2), ..., h_i(j), ..., h_i(q)\}^T$, $h_i(j)$ is the first feature of i, $h_i(j) = \{d_1, d_2, ..., d_l, ..., h_R\}$, d_l is the l value of j, that is, there are M samples in the data set, q features in each sample, and R values in each feature.

The value set of sample category is $g_i = \left\{\hat{g}_1, \hat{g}_2, ..., \hat{g}_p\right\}$, that is, sample category has p.

In information theory, information entropy represents the uncertainty of information system caused by the existence of random variables. Assuming that the probability of occurrence of each category $\hat{g}_i$ is $f\left(\hat{g}_i\right)$, the entropy of the classification system is:

$$V(G) = -\sum_{i=1}^{p} f\left(\hat{g}_i\right) \ln f\left(\hat{g}_i\right) \tag{3}$$

Under the condition that the value of feature $h_i(j)$ is d_l, the definition of system conditional entropy is:

$$V(G|d_l) = f(d_l) \sum_{i=1}^{p} f\left(\hat{g}_i|d_l\right) \ln f\left(\hat{g}_i|d_l\right) \tag{4}$$

When the value of feature $h_i(j)$ is d_l, the information gain of the feature is defined as the difference between the entropy $V(G)$ of the classification system and the conditional entropy $V(G|d_l)$ of feature $h_i(j)$ under a given category d_l, and the calculation formula is

$$U(d_l) = V(G) - V(G|d_l) \tag{5}$$

The information gain value $U(d_l)$ indicates the degree to which the uncertainty of data set classification is reduced by the value d_l of feature $h_i(j)$. The greater the U value, the greater the reduction of uncertainty, the greater the importance of the classifier, and the stronger the classification ability of the feature. On the contrary, the smaller $U(d_l)$ is, the smaller the uncertainty of data set classification is reduced by the value d_l of feature $h_i(j)$, and the smaller the importance of the feature to the classifier is, the weaker the classification ability of the feature is.

In this chapter, the features in the training set are arranged in descending order according to the $U(d_l)$ value, and a small number of features with the highest order can be selected to achieve feature extraction. Eliminate the features that rank behind and have no or little influence on classification, reduce the operation amount of the classifier, and improve the operation efficiency.

Based on the features extracted by information gain calculation, the feature matrix is constructed. First, the MI based ranking strategy is adopted to expand the samples at each time into a dynamic feature matrix. The real-time features in the matrix can reflect the

relevant features at the current time, and the delay features reflect the dynamic characteristics of the system operation. In this experiment, the data sets pass the nonparametric verification of SPSS software, and the minimum confidence levels are greater than 0.05, which conforms to the Gaussian distribution. Therefore, this paper uses PCA feature extraction algorithm to obtain the principal component space Y, and introduces the principal component features of the first z moments of Y to form the following augmented matrix.

$$Y = \begin{bmatrix} y_1^{t-z} & y_1^{t-z+1} & \cdots & y_1^t \\ y_2^{t-z} & y_2^{t-z+1} & \cdots & y_2^t \\ \cdots & & & \\ y_r^{t-z} & y_r^{t-z+1} & \cdots & y_r^t \end{bmatrix} \tag{6}$$

where, Y is the augmented matrix; y_i^t represents the value of the i th feature at time t; r is the number of selected features.

Calculate the MI between each feature and all other time features to measure the serial correlation between features, i.e. $J(i, j)$.

$$J(i, j) = y_i^t \cdot Y \tag{7}$$

Then the characteristic matrix J is constructed.

$$J = \begin{pmatrix} o_{11} & \cdots & o_{1r} \\ \vdots & \ddots & \vdots \\ o_{t1} & \cdots & o_{tr} \end{pmatrix} \tag{8}$$

The value standard of matrix J is as follows:

MI greater than 1, value 5;
MI is between 0.8 and 1, and the value is 4;
MI is between 0.6 and 0.8, and the value is 3;
MI between 0.4 and 0.6, value 2;
MI is between 0.2 and 0.4, value 1;
If MI is less than 0.2, the value is 0;

2.3.2 Intrusion Identification

The constructed data feature matrix is regarded as a feature map composed of numbers, and the 3D convolution neural network algorithm (3D CNN), which is commonly used in the depth learning algorithm, is used for intrusion recognition. 3D CNN is one of the representative algorithms in the field of depth learning. It is a feedforward neural network including convolution computation, and is widely used in image processing, natural language processing and many other fields. In 3D CNN, the parameter sharing of its convolution kernel and the sparsity of the connection between layers can enable the network to extract feature information with less computation, which has a stable effect and no additional feature engineering requirements for data. The error function value is

obtained by calculating the difference between the real value and the predicted value, so as to adjust the network parameters reversely until the model reaches the optimum.

In a typical 3D CNN, the first several layers are convolutions, and the last ones close to the output layer are fully connected one-dimensional networks (i.e., traditional BP neural networks). In 3D CNN network, nonlinear mapping of different features can be realized through convolution layer. The work of feature extraction is also mainly completed by the convolution layer, which outputs the feature vector after convolution through multiple convolution calculations. In the process of convolution, after the current receptive field and convolution nucleus have a convolution operation, they slide to the next window to continue convolution, and the sliding amplitude is set by the convolution step. The specific definition formula of convolution operation is as follows.

$$\beta_i^j = \varsigma\left(\sum_{i\in\chi_j} w_{ij}\alpha_i * \gamma_{ij} + \delta_j\right) \tag{9}$$

where, β_i^j represents the convolution output of layer j; α_i represents the characteristic matrix of layer i; γ_{ij}, w_{ij} represents the convolution kernel and weight of layer i and layer j; δ_j represents the offset term of layer j; χ_j represents the subset of input characteristic graph used to calculate β_i^j; ς stands for the activation function.

The complexity of the whole network becomes higher due to more features after the convolution layer operation. Pooling can divide the image into many non overlapping regions, and then calculate the nodes in different regions. Therefore, after the pooling layer, the feature dimensions are reduced while the main feature information is retained, and the complexity of the network structure is reduced.

$$\lambda_l = \xi\left(w_{jl}\beta_i^j\right) + \delta_l \tag{10}$$

where, λ_l represents the output of pooling layer l; ξ represents pooling function; δ_l, w_{jl} represents the offset and weight of pooling layer l; ξ stands for the activation function.

In the fully connected layer, each neuron is connected to all outputs of the previous layer. Usually at the end of CNN, the final classification function can be realized. After the input data has undergone convolution pooling and other operations, the output feature vectors pass through the full connection layer, are classified through the Softmax function, and the prediction results are output.

The output calculation formula of the full connection layer is as follows:

$$\mu_x = \zeta(w_{xl}\lambda_l) + \delta_x \tag{11}$$

where, μ_x represents the output of full connection layer x; ζ represents the full connection layer activation function, and w_{xl} represents the connection weight value.

In the structural design of 3D CNN model, for the large size of convolution kernel in the classic LeNets architecture, the two continuous convolution and pooled stacking methods are prone to over fitting. In this paper, the depth separable convolution layer is used to replace the conventional convolution layer, and the nonlinear modules in the network are added to compress the network parameters and accelerate the convergence

speed of the model. The batch normalization layer is added to normalize the standard normal distribution of the middle layer, so as to reduce the impact of super parameter fluctuations in the network, smooth the optimization space in the training and accelerate the network convergence. In this paper, we choose to use PReLU function as the activation function, which can effectively reduce the risk of over fitting and the vulnerability of neural units in the training process. Finally, LSTM method is integrated to make the intrusion detection more accurate by using its feature of preserving the sequence of features.

After the structural adjustment of the convolutional neural network model, an intrusion detection model based on 3D CNN is established to process the input data. For the processed network data packets, the mainstream method is to build a network model based on one-dimensional convolution to process and classify the network data. In the data processing phase, this method converts the one-dimensional sequence data type to the three-dimensional digital matrix form to form a classifier. The specific process is as follows:

The first step is data input. The data samples are preprocessed to expand the features, and then converted into a 21 * 21 two-dimensional matrix as the input of the network.

The second step is feature extraction. Feature extraction mainly includes Depth Separable convolution, Max Pooling, batch normalization and PReLU activation function. Batch normalization and activation functions are nested in each convolution layer. By discarding the original convolution operation, two separable convolutions are used for feature extraction to reduce the model training parameters.

The third step is to classify the branch roads. The secondary branch and the primary network share the convolution layer, and the secondary branch connects a full connection behind the convolution layer. The data is divided into two categories: normal and abnormal. The branch network is set to make a second classification judgment, so as to prevent feature loss in the process of down sampling and ensure that the network can learn distinguishing features between various abnormal samples and normal samples.

Fourth, convolution and pooling can effectively extract features, but network intrusion data also has its corresponding sequence relationship. By using the sequence relationship between network learning features, we can detect whether there is an intrusion.

3 Experiment Design and Result Analysis

3.1 Data Description

At present, KDD99 dataset is the earliest one used in the public network intrusion detection dataset. The NSL-KDD dataset is an improved version of KDD99, which removes a lot of redundant data. In this dataset, each record has 41 attribute characteristics and 1 category label. The category tag contains one normal data category and four types of intrusion attack data categories.

The Network Security Research Group of the Australian Network Security Center (ACCS) introduced the UN SW-NB 15 dataset. In this dataset, the training set has 175343 connection records, and the test set consists of 82337 connection records, including 9

intrusion attack categories and 1 normal category. Each record contains 42 attribute characteristics and 1 category label.

The model diagram of random intrusion depth detection method based on three-dimensional convolution neural network is shown in Fig. 3.

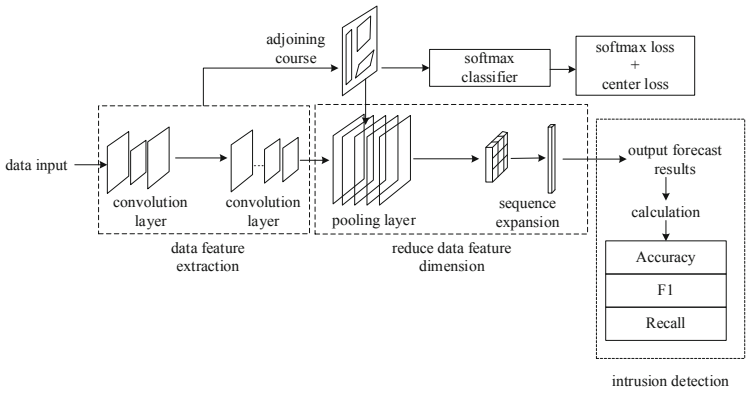

Fig. 3. Model diagram of random intrusion depth detection method based on three-dimensional convolution neural network

3.2 Feature Extraction Result Test

For the two data sets, the first 15 features are extracted using information gain. The information gain of each feature is shown in Table 1 below.

Table 1. Feature Extraction Results

Feature serial number	Information gain value	Information gain value
1	16.32	17.41
2	14.21	15.23
3	14.02	14.21
4	13.65	14.02
5	12.10	12.52
6	11.25	11.22
7	10.52	10.52
8	9.52	8.12
9	8.85	8.04
10	8.62	7.65

The feature matrix is constructed according to the extracted features. Taking one of the data samples as an example, the characteristic matrix is as follows:

$$J = \begin{bmatrix} 3 & 1 & 4 & 3 & 2 & 4 & 2 & 4 & 0 & 5 \\ 5 & 3 & 1 & 3 & 5 & 2 & 0 & 5 & 4 & 5 \\ 0 & 5 & 3 & 0 & 4 & 2 & 4 & 4 & 0 & 2 \\ 3 & 0 & 1 & 1 & 0 & 2 & 0 & 1 & 4 & 5 \\ 5 & 2 & 5 & 1 & 4 & 5 & 0 & 5 & 3 & 5 \\ 4 & 4 & 4 & 1 & 0 & 2 & 3 & 0 & 4 & 2 \\ 3 & 3 & 2 & 1 & 5 & 2 & 0 & 1 & 3 & 0 \\ 5 & 5 & 0 & 2 & 3 & 5 & 0 & 3 & 5 & 2 \\ 0 & 3 & 4 & 1 & 5 & 4 & 2 & 5 & 3 & 5 \\ 4 & 4 & 2 & 4 & 5 & 1 & 5 & 2 & 1 & 3 \\ 4 & 2 & 2 & 5 & 4 & 1 & 1 & 5 & 3 & 5 \end{bmatrix} \tag{12}$$

3.3 Test Method Performance Test

The evaluation indicators used for intrusion detection include accuracy (φ), recall (ψ) and F1. In intrusion detection, attack samples are regarded as positive samples and non attack samples as negative samples. TP represents the number of samples that were originally positive and were predicted to be positive; TN represents the number of samples originally negative and predicted to be negative; FP represents the number of samples whose positive classes are predicted to be negative; FN indicates that the number of negative samples is predicted to be positive samples.

Accuracy: it indicates the proportion of real positive samples in the total number of predicted positive samples;

Recall rate: refers to the percentage of the samples that are actually positive and correctly predicted in all the predicted positive samples. The higher the value, the higher the reliability of the positive sample predicted by the model.

$$\begin{cases} \varphi = \dfrac{TP}{TP + FP} \\ \psi = \dfrac{TP}{TP + FN} \end{cases} \tag{13}$$

F1 Measure is a balance point that considers the precision and recall at the same time, so that both can reach a relatively high value at the same time. F1 Measure can be regarded as a weighted average of the precision and recall, with the maximum value of 1 and the minimum value of 0.

$$F1 = 2\left(\frac{\phi\psi}{\phi + \psi}\right) \tag{14}$$

Under the same test data set, use the research methods, intrusion detection technology based on Bayesian network, intrusion detection method based on improved SMOTE, and

intrusion detection method based on cuttlefish optimization algorithm to conduct random intrusion detection of the Internet of Things, and get the detection results. According to the results, calculate the F1 value. The results are shown in Fig. 4 below.

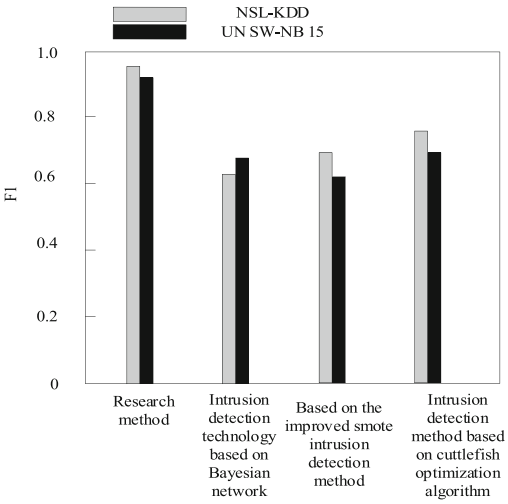

Fig. 4. Comparison Diagram of F1 Values

It can be seen from Fig. 4 that when the research method detects NSL-KDD data set and UN SW-NB 15 data set, its FI value is above 0.8. When the intrusion detection technology based on Bayesian network, the intrusion detection method based on improved SMOTE, and the intrusion detection method based on squid optimization algorithm detect NSL-KDD dataset and UN SW-NB 15 dataset, the F1 value of these three methods is less than 0.8. The comparison shows that the F1 value of the research method is significantly higher than that of the other three comparison methods, indicating that the detection accuracy of this method is higher.

4 Conclusion

As the network environment becomes more and more complex, network security accidents occur frequently, and attack means change constantly, which threaten the security of the Internet of Things. Therefore, the important role of Internet of Things intrusion detection becomes more and more obvious. Aiming at the low accuracy of intrusion detection in industrial Internet of Things, a random intrusion depth detection method based on three-dimensional convolution neural network is proposed. According to NIDS, the Internet of Things intrusion detection model is built, network data packets are collected and preprocessed. The random intrusion depth detection in the Internet of Things is completed by constructing the feature matrix and taking it as the input, using 3 DCNN and LSTM method. Experimental results show that the proposed method has high detection accuracy. Because in the actual network environment, all kinds of unpredictable

emergencies are uncontrollable. Therefore, in order to further improve the stability of the network intrusion detection model in the real environment, further research is needed.

References

1. Zhang, H., Li, Y., Lv, Z., et al.: A real-time and ubiquitous network attack detection based on deep belief network and support vector machine. IEEE/CAA J. Autom. Sin. **7**(3), 790–799 (2020)
2. Dong, R.H., Yan, H.H., Zhang, Q.Y.: An intrusion detection model for wireless sensor network based on information gain ratio and bagging algorithm. Int. J. Netw. Secur. **22**(2), 218–230 (2020)
3. Yang, W.H.: Security detection of network intrusion: application of cluster analysis method. Comput. Opt. **44**(4), 660–664 (2020)
4. Zhu, D.H., Cheng, Y.: Leak control of sensitive data in Internet of Things based on local differential privacy. Comput. Simul. **38**(02), 472–476 (2021)
5. Jo, W., Kim, S., Lee, C., et al.: Packet preprocessing in CNN-based network intrusion detection system. Electronics **9**(7), 1151 (2020)
6. Alhajjar, E., Maxwell, P., Bastian, N.: Adversarial machine learning in network intrusion detection systems. Expert Syst. Appl. **186**(2), 115782 (2021)
7. Li, J., Wu, W., Xue, D.: An intrusion detection method based on active transfer learning. Intell. Data Anal. **24**(2), 363–383 (2020)