


A Novel CNN-LSTM Fusion-Based Intrusion Detection Method for Industrial Internet

Jinhai Song, Henan University of Science and Technology, China

Zhiyong Zhang, Henan University of Science and Technology, China*

 <https://orcid.org/0000-0003-3061-7768>

Kejing Zhao, Henan University of Science and Technology, China

Qinhai Xue, Henan University of Science and Technology, China

Brij B. Gupta, Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan, & Lebanese American University, Beirut, Lebanon, & Center for Interdisciplinary Research at University of Petroleum and Energy Studies (UPES), Dehradun, India, & UCRD, Chandigarh University, Chandigarh, India

ABSTRACT

Industrial internet security incidents occur frequently, and it is very important to accurately and effectively detect industrial internet attacks. In this paper, a novel CNN-LSTM fusion model-based method is proposed to detect malicious behavior under industrial internet security. Firstly, the data distribution is analyzed with the help of kernel density estimation, and the Pearson correlation coefficient is used to select the strong correlation feature as the model input. The one-dimensional convolutional neural network and the long short-term memory network respectively extract the spatial sequence features of the data and then use the softmax function to complete the classification task. In order to verify the effectiveness of the model, it is evaluated on the NSL-KDD dataset and the GAS dataset, and experiments show that the model has a significant performance improvement over a single model. In the detection of industrial network traffic data, the accuracy rate of 97.09% and the recall rate of 90.84% are achieved.

KEYWORDS

Industrial Intrusion Detection, Kernel Density Estimation, Long Short-Term Memory Network, One-Dimensional Convolution, Pearson Correlation Coefficient

INTRODUCTION

The rapid development and improvement of the consumer Internet has made people begin to explore and practice the “industrial Internet.” From a macroperspective, the industrial Internet connects industrial control systems (ICSs) and the Internet with the aim of making production more intelligent. On the microlevel, the industrial Internet abandons the traditional closed and trusted environment in

DOI: 10.4018/IJISP.325232

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

industrial control, integrates the exchange characteristics of the Internet, and connects the equipment, workshops, factories, employees, and customers in the industrial system using the Internet as a hub to connect the network (Alanazi et al., 2022). It promotes the intelligentization of the industry and realizes the interindustry intercommunication and the sharing of resources. However, as the industrial Internet continues to grow, it also comes with more security concerns. Over the past few years, the number of cyber attacks on the industrial manufacturing industry also increased significantly, compared with the past (Gauthama Raman & Mathur, 2022). A cyber attack on an industrial system may cause data leakage, system damage, production interruption of industrial enterprises, and even the closure and bankruptcy of enterprises, causing harm to the national social economy (Anthi et al., 2021).

According to the security status of the industrial Internet, the traditional intrusion detection system cannot effectively deal with most intrusions. Traditional intrusion detection mainly uses pattern matching and different protocol analysis techniques for detection (Gupta et al., 2009; Mishra et al., 2011). By establishing normal behavior patterns or modeling known attacks as the detection benchmark, this method is too dependent on the integrity of the modeling and must be accompanied by high rate of false positives. The industrial Internet, which links the conventional ICS to the Internet, confronts not only the security concerns of the old ICS, but also the inherent security challenges of the Internet (Chhetri et al., 2018). The cross-border integration of information technology and operation technology blurs the border between the security of industrial manufacturing and the security of the external Internet (Kou et al., 2022). In addition, and traditional detection methods are no longer suitable for detecting the current industrial Internet. The rise in popularity of deep learning has had far-reaching effects on fields such as voice and picture recognition, as well as introducing novel concepts to fields such as intrusion detection (Abu-Khzam et al., 2022; Malik et al., 2022). Applying deep learning to the field of intrusion detection can not only improve the detection rate, but also further simplify the problem of intrusion detection (Sayour et al., 2022).

Most of the traditional intrusion detection methods based on machine learning algorithms only use a single algorithm for classification and recognition, without performing feature processing on the data, which is relatively sluggish, the detection rate is not high, and it cannot accurately respond to the detection of intrusion behavior (Zhang et al., 2021). In order to achieve accurate and rapid detection of industrial Internet intrusion, it is necessary to respond to security risks and threats from both ICS and ordinary Internet. In this study, the authors selected two datasets, namely NSL-KDD (Tavallae et al., 2009) and Gas pipeline datasets (Morris et al., 2015), which contain a large number of different attack types. At the same time, this approach is convenient, accurate, and fast to detect attack behaviors, remove irrelevant features in different datasets, and better improve the classification performance. The authors designed an intrusion detection classification model based on neural network (CNN)-long short-term memory (LSTM). The main contributions are as follows:

1. Analyzing the original dataset, using one-hot encoding, standardization, and normalization for preprocessing, using the kernel density estimation (KDE) map to compare the data distribution, and using the Pearson correlation coefficient for feature selection to improve the accuracy of model recognition.
2. Designing a model for classification evaluation and comparative analysis of intrusion detection attacks. The test results showed that the model has better performance than the reference model.
3. Verifying the effectiveness and generalization of the evaluation method for intrusion detection classification on the intrusion detection dataset and the industrial control dataset.

RELATED WORK

The industrial Internet network traffic data are complex and changeable, and the feature redundancy is high, which makes security breaches difficult to detect. With the addition of deep learning, detecting these security issues has become easier and more efficient (Hammad et al., 2021; Helwan et al., 2021).

Al-Abassi et al. (2020) proposed an intrusion detection method based on deep neural network and decision tree. Firstly, Al-Abassi et al. Extracted a new balanced representation from the unbalanced original data through multiple sparse autoencoders, and then passed to the deep neural network and decision tree for classification and detection. Validated on the real SCADA (i.e., supervisory control and data acquisition) dataset, the results showed that the combined architecture can handle complex data samples with mixed features and has high detection accuracy.

Li et al. (2019) proposed a detection model using multi-CNN fusion model, using prior information or clustering algorithms to cluster features, considering the correlation between features. The 1-dimensional data are converted into 2-dimensional, and the blank is filled with zeros; multiple CNN models are trained with the clustered data, there is no output layer, and then the multiple models are fused. Li et al. verified the model on the NSL-KDD dataset and achieved good experimental results.

Ren et al. (2022) calculated the Pearson correlation coefficient, and compared the correlation of different features according to the coefficient to filter features. By setting different correlation thresholds and comparing the impact on detection results under different threshold strengths, they found the optimal thresholds and classified and detected them under different learning models. They used the UNSW-NB15 and CSE-CIC-IDS2018 datasets for experimental verification, and concluded that the feature selection by the Pearson correlation coefficient can significantly improve the model accuracy.

Kravchik and Shabtai (2021) used one-dimensional CNN, shallow variational auto-encoders (VAEs), and principal components analysis to detect ICS data and apply them to the time domain and frequency domain of the data, showing the effectiveness in the frequency domain and excellent performance on the SWaT, BATADAL, and WADI datasets.

In most of the above studies, various learning models did not make full use of binary and multiclassification experimental verification. In addition, there is no sufficient verification, the experimental dataset is single, and there is no generalization consideration. In this study, by calculating the Pearson correlation coefficient, selecting appropriate features, and performing classification experiments by combining different machine learning and deep learning models, the authors carried out a detailed analysis of the industrial Internet intrusion detection

PROPOSED APPROACH

In this section, the authors introduce the data preprocessing method, present the design of the attack detection experiment, and describe the evaluation metrics.

Data Preprocessing

Original data often have problems such as redundancy, loss, and sample imbalance, so corresponding processing is required, that is, data preprocessing. In this study, the authors selected the NSL-KDD dataset and the Gas pipeline dataset (Morris et al., 2015). The NSL-KDD is optimized on the basis of KDD-99, contains more comprehensive network attacks than KDD-99, and is more suitable for testing model performance. The Gas pipeline is more representative of industrial control datasets. Tables 1 and 2 show the sample labels. Among them, the Gas pipeline dataset (Morris et al., 2015) is the second iteration of the dataset for natural gas pipeline systems. Compared with the first iteration of the Gas pipeline dataset (Morris et al., 2014), the authors excluded the obvious patterns contained in the first iteration. The pattern of the model will lead to a significant correlation between specific parameters and the results predicted by the algorithm, making it unsuitable for IDS research. In this study, the authors used the second iteration of the Gas pipeline dataset (Morris et al., 2015). All the data the researchers used in this work are available on the ICS Cyber Attack Datasets's Web site (<https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets?pli=1>) and the Canadian Institute for Cybersecurity's Web site (<https://www.unb.ca/cic/datasets/nsl.html>).

Data Normalization

First, it is necessary to normalize the data type. Each intrusion detection dataset has many feature attributes, such as basic features, content features, and time features, and the data representations may be of different types. The normalization of data types is about unifying different data types into the same digital identifier, eliminating the difference in characteristic attributes caused by different data types, and facilitating subsequent calculations. Then, the researcher performs numerical normalization. It is necessary to select min-max normalization to scale the data uniformly in the range [0, 1]. The dimension of the data is to be planned in a unified manner, so that the value of each characteristic attribute is kept at the same order of magnitude. The min-max formula is as follows:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

One-Hot Encoding

The researcher converts the label to a vector of 0s and 1s. For features with many label categories, one-hot encoding is not suitable. One-hot encoding will produce a vector with as many dimensions as the amount of categories of labels. The more categories, the higher the dimension and the sparser the result, which is more prone to parallelism and multicollinearity problems. Thus, it also faces the risk that it may become a sparse matrix. Before performing one-hot encoding on the dataset, it is necessary to calculate the Pearson correlation coefficient and complete the feature screening to reduce the feature dimension (Ren et al., 2022) and, at the same time, avoid the above risks.

Visualization of Sample Distribution

The researcher analyzes the data by visualizing the sample distribution. For prediction and classification research, one of the most important factors affecting test results is the distribution of training and test sets. Inconsistent data distribution will most likely lead to model overfitting (García et al., 2012). In this research, the authors chose KDE. The kernel function creates an independent probability density curve for each sample value and then sums these smooth curves to obtain a smooth continuous probability distribution curve. By drawing the KDE map and comparing the distribution of the training set and the test set, it is smoother than the histogram and is less affected by the bin width. Also, the dataset is balanced according to the distribution map, so that the sample distributions are more similar, which facilitates the fitting of the model. The KDE function and kernel function calculation formula are as follows:

$$\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^n K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right) \quad (2)$$

$$K(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}x^2\right) \quad (3)$$

f represents the overall probability density function, h is a hyperparameter, called bandwidth or window, n represents the total number of samples, and K represents the kernel function. The authors used the Gaussian kernel function, so K represents the Gaussian kernel function.

Table 1. Samples label of the KDD dataset

Label Name	Label Value
DOS	0
Normal	1
Probing	2
R2L	3
U2R	4

Table 2. Samples label of the gas pipeline dataset

Label Name	Label Value
Normal	0
NMRI	1
CMRI	2
MSCI	3
MPCI	4
MFCI	5
DoS	6
Reconnaissance	7

The above preprocessing is to make the input data homogeneous and eliminate heterogeneous data, making the network easier to learn and converge.

According to Ren et al. (2022), when the threshold of Pearson coefficient is selected as 0.2, the comprehensive effect of classification is optimal. In order to ensure that enough features for learning are available, the authors selected features with a Pearson correlation coefficient greater than or equal to 0.1 when performing feature selection on the data. The Pearson correlation coefficient is often used to describe the strength of the correlation between variables. It does not change due to changes in scale or location between variables, and it has high computational efficiency, which is excellent when dealing with large-scale data. The measurement interval of the Pearson correlation coefficient is $[-1, 1]$, which also allows the Pearson correlation coefficient to represent rich relationships. The closer the result value is to 1, the stronger the positive correlation between the variables; the closer the result value to -1, the stronger negative correlation between the variables; the closer the result value to 0, the weaker the correlation. Therefore, the researcher should select positively correlated features that have a strong impact on classification in the data. Among them, Figure 1 shows the heatmaps of the data correlation of each variable in the KDD datasets. Taking KDD as an example, the authors retained character features such as “protocol_type,” “service,” and “flag.” Among numerical features, they screened out features with a correlation coefficient value less than 0.1. Table 3 shows the partial correlation coefficient values of the KDD dataset.

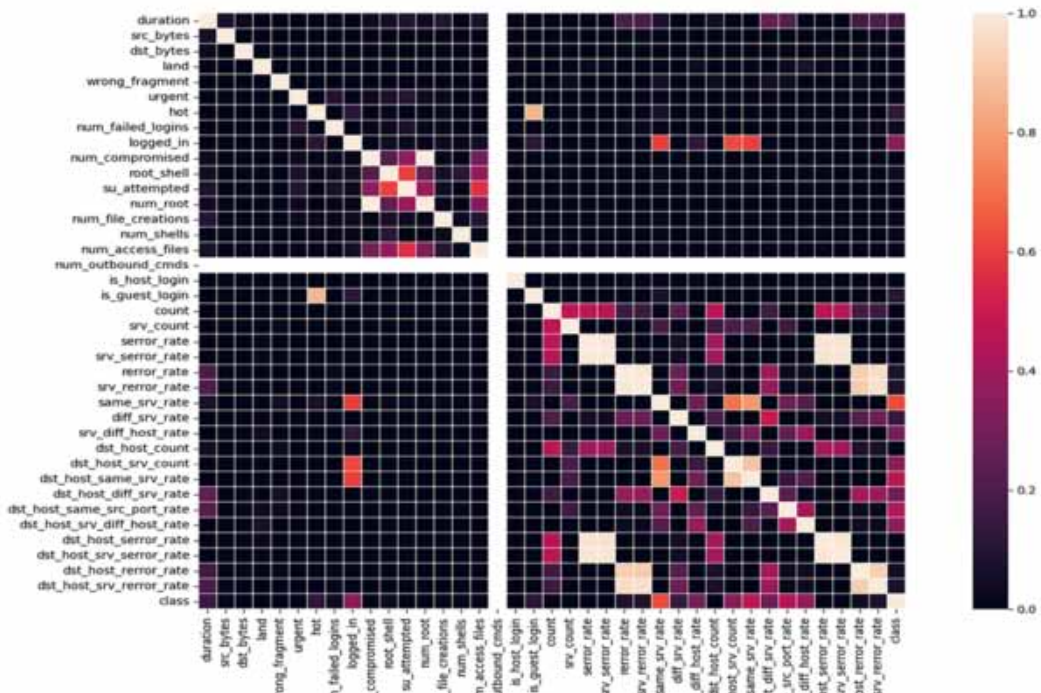
After the feature selection of the Pearson correlation coefficient, the authors used the KDE of the kernel function to visualize the data distribution; the kernel function is a Gaussian function. In order to avoid artificial design during the reconstruction process, they selected a random seed. First, they integrated the data, and then randomly scrambled it given a random seed 42. Subsequently, they divided according to the original training-test ratio 8/2 and then visualized the KDE distribution. The data distribution after division is approximately balanced and the training and test data do not

Table 3. Partial correlation coefficient values for the KDD dataset

Feature Name	Correlation Coefficient Value
duration	0.113273
src_bytes	0.011125
dst_bytes	0.006922
land	-0.01149
wrong_fragment	-0.08755
urgent	0.029384
hot	0.103386
num_failed_logins	0.153461
logged_in	0.283995
...	...

Figure 1. Heatmap of the KDD

Note. The color of the blocks represents the strength of correlation. The darker the color, the weaker the correlation, and the lighter the color, the stronger the correlation.



overlap. Figure 2 illustrates the sample distribution diagram. Among them, 0 to 4 represent different peaks, namely sample categories Dos, normal, probe, R2L, U2R. The higher the peak, the denser the data. Tables 4 and 5 show the KDD datasets before and after data preprocessing.

Figure 2. Estimation diagram of unbalanced (on the left) and approximate balanced (on the right) KDD kernel density
Note. The peak value represents density

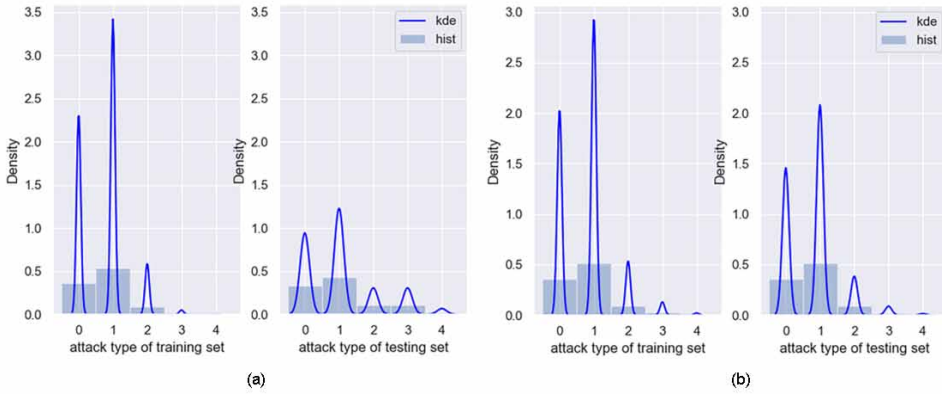


Table 4. Unprocessed KDD dataset

Type	Train	Test
Normal	65536	9711
DOS	45921	7458
Probing	11656	2421
R2L	995	2421
U2R	52	533

Table 5. Preprocessed KDD dataset

Type	Train	Test
Normal	65416	11638
DOS	45921	8149
Probing	11912	2165
R2L	2914	502
U2R	496	89

Classification Model

The authors compared various models in machine learning and deep learning, selected a model with a strong Pearson correlation coefficient for fitting training and compared the outcome. The experimental comparison model in this study mainly includes multilayer perceptron (MLP), random forest, and naive Bayes. In the machine learning model, the MLP consists of stacked multilayer, dense layers or multilayer linear layers. The expectation is output by transferring the data in the hidden layer and using a nonlinear excitation function. Random forest belongs to the parallelization method in the ensemble method. Its base learner is a decision tree, which is composed of multiple subtrees (submodels) that are independent and independent of each other, and the subtrees perform random sampling with replacement on the given dataset. The joint analysis operation of naive Bayes

mainly emphasizes its conditional independence, that is, the classification based on Bayes' theorem under conditional independence. The learning strategy of the support vector machine (SVM) is to maximize the interval, that is, to find the separation hyperplane with the largest geometric interval, so as to realize the efficient classification of data.

The CNN-LSTM Model

The Convolutional Neural Network

A CNN is a feedforward neural network which is widely used for image recognition (Al Sobhahi & Tekli, 2022; Chui et al., 2022). The main application principle of a CNN is to use its local sensitivity and direction selection characteristics to avoid complex preprocessing of data and reduce complexity (Islam et al., 2020). CNNs can be divided into three layers: Convolutional, pooling, and fully connected, for feature extraction, down-sampling, and classification, respectively (Hassan et al., 2020). The convolutional layer is the main component of a CNN; it extracts feature information by applying filters to the feature map from the previous layer (Zhang et al., 2018). The pooling layer reduces the amount of parameters by down-sampling a given dimension, usually selecting the largest value, in order to achieve spatial invariance. Finally, the fully connected layer makes a decision based on the features obtained by the above two layers.

The Long Short-Term Memory Network

The LSTM alleviates the short-term delay memory problem of the recurrent neural network (RNN), and has the ability to learn and save data information longer. The core part of the LSTM is the cell state. The cell state corresponding to time t is C_t . The cell state runs through all moments. During forward propagation, gates are used to control the increase or decrease of information in C_t . The gate in the LSTM is implemented by a neural network layer with activation function sigmoid, and the output value of the gate is between 0-1. Then, it is necessary to multiply the value vector of the gate and the target data bit by bit to achieve the effect of controlling the data flow. The mathematical formulas of the LSTM are as follows:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (4)$$

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

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (6)$$

$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t \quad (7)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (8)$$

$$h_t = o_t * \tanh(c_t) \quad (9)$$

i_t, f_t , and o_t represent the input gate, forget gate, and output gate, respectively. W represents the weight matrix, b represents the bias vector, c_t represents the current storage unit state, and h_t represents the current hidden state. Each gate uses the sigmoid activation function, and the state and output use the hyperbolic tangent function as activation function. These two functions are saturation functions, which are convenient for realizing the gating effect.

The CNN-LSTM Model

The model the authors propose in this paper includes CNN and LSTM. The main model of the CNN is to perform convolution pooling operation on the input data, and then enter the fully connected layer to complete the classification. This architecture makes it highly translation invariant, and the RNN has

an excellent effect on the prediction of sequence data, so, in this study, the authors chose the variation of the RNN- LSTM to capture longer distance dependencies of features. The structure level is:

conv1D maxpooling×2→flatten→fullConnection→LSTM→softmax→output

First, the authors used the local feature extraction of the CNN to learn more comprehensively the data features, and then they used the sensitivity of the LSTM to the order of data features to deal with its influence. The very purpose of using the fusion learning method was to improve the performance of the model through the complementarity of different learning methods (Alghamdi & Bellaiche, 2023). Figure 3 presents the flowchart of the proposed model and Figure 4 represents the algorithm model.

Evaluation Index

The authors used the confusion matrix and its derived indicators to evaluate the performance. A confusion matrix is a summary of the prediction results for a classification problem. By summarizing the number of actual categories and classification categories, it is easy to understand the defects of the model prediction and the categories in which errors occurred. Table 6 gives the definition of the confusion matrix and Table 7 presents the evaluation of the classification models.

EXPERIMENTS

Experimental Environment

In this study, the experimental test environment was Windows 10 PC, Intel(R) Core (TM) i5-7300HQ CPU @ 2.50GHz, 8.00GB RAM. The authors implemented the algorithms using Sklearn, keras, and tensorflow libraries in the Python language.

Model Parameters

The previous section provided the overall architecture of the model. The authors chose leaky_relu as the activation function in the network. In order to avoid model overfitting, they added a dropout layer. The dropout layer would randomly drop neurons, get rid of the dependencies between neurons, and improve the generalization performance. The authors opted for the leaky_relu function for the activation function because this function not only avoids gradient disappearance and prevents gradient saturation, but it also does not cause the problem of gradient disappearance, does not cause the permanent death of neurons due to the negative value of the weight parameter, and does not involve complex exponential calculations and operations. The expression of this method high efficiency and fast convergence speed. The expression is as follows,, and a is in the $(0,1)$ interval:

$$f(x) = \max(ax, x) \tag{10}$$

As the bottom layer of CNN-LSTM, the LSTM mainly stores the timing information of the main features of intrusion data extracted by the CNN. The activation function also uses leaky_relu.

Figure 3. Flowchart of the proposed IDS

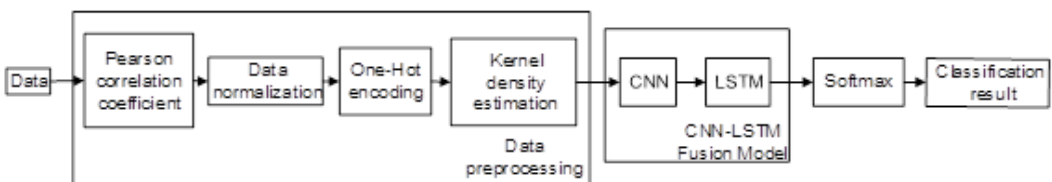


Figure 4. Algorithm model diagram: Network layer and its input and output

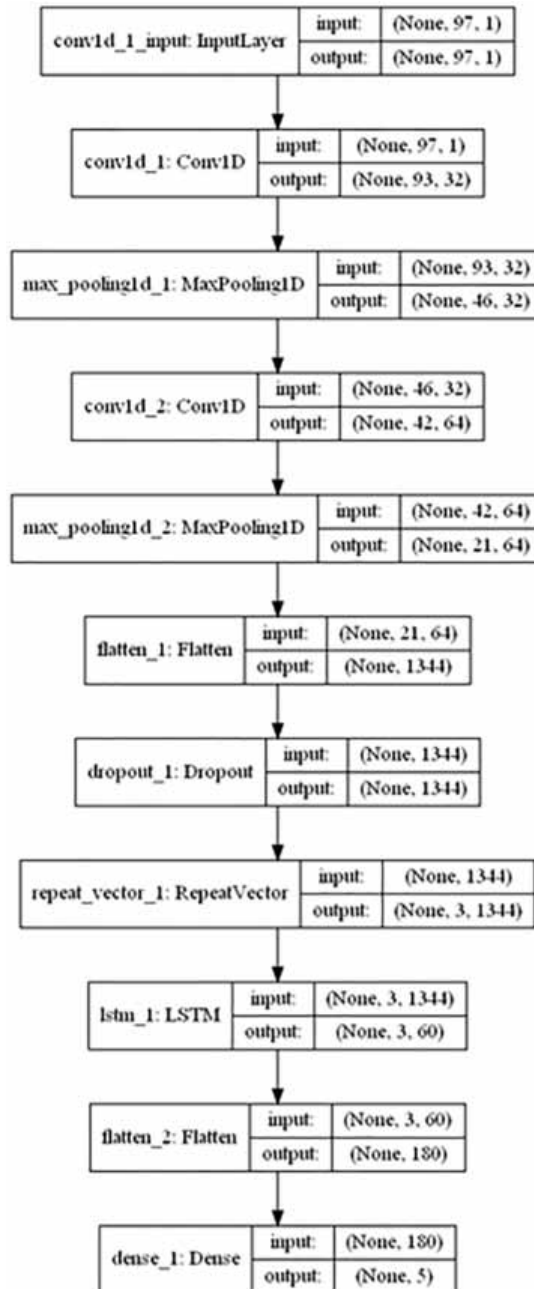


Table 6. The confusion matrix

Actual/Predicted	True Positive	True Negative
Hypothesized positive	TP	FP
Hypothesized negative	FN	TN

Note. TP = true positive; FP = false positive; FN = false negative; TN = true negative.

Table 7. Index calculation formula

Metrics	Formula
Accuracy	$ACC = (TP + TN) / (TP + TN + FP + FN)$
F1 score	$F1 = 2TP / (2TP + FP + TN)$
Precision	$Precision = TP / (TP + FP)$
Recall	$Recall = TP / (TP + FN)$

Experimental Results

Ablation Experiment

In order to prove the rationality and validity of the model, the authors performed an ablation experiment on the model before the comparative experiment. The ablation experiment is effective for almost any research goal in the field of deep learning. It explains the performance impact of model “ablation” or “melting” more intuitively by replacing or deleting different modules. Ablation experiments are in principle similar to control variables in physics. The ablation experiments in this study allowed to compare the proposed CNN-LSTM network model with a local algorithm CNN network and a two-layer LSTM network. The selected experimental dataset was NSL-KDD; the authors uniformly performed the preprocessing they described in the section *Data Preprocessing* above on the dataset to ensure that they carried out the experiment under the same conditions. Table 8 shows the experiment was a multicategory (five-category) comparative experiment and illustrates the experimental results.

After comparison, the results showed that, under the same conditions, the CNN-LSTM model has better detection performance in all indicators, and the overall performance is the best. The experimental results showed that the authors’ CNN-LSTM model has high performance in intrusion detection.

Comparative Experiment

The authors generated the comparison algorithms from the scikit-learn library and used the NSL-KDD detection dataset. They filtered out the feature items with low correlation such as “land” and “urgent” in train and test datasets, finally obtained 17-dimensional features, and performed one-hot encoding and min-max normalization on them through functions such as `get_dummies`. In view of the serious category imbalance in the dataset, the researchers visualized and reconstructed the sample distribution of the dataset through KDE. They used a Gaussian function as the kernel function.

The authors placed the processed dataset on the CNN-LSTM model for fitting training, and selected the training set of 0.3 as the validation set. Taking multiclassification as an example, Figure 7 shows the iterative training effect of CNN-LSTM and Figure 8 shows the loss curve. The authors placed the processed dataset on the selected machine learning model algorithm for fitting, training, and testing. To avoid numerical fluctuation caused by errors, the researchers selected the average of five experimental results as the final result value. Tables 9 and 10 show the binary and multiclassification accuracy of each algorithm on the NSL-KDD dataset. Among them, the two categories are normal and abnormal, and the multicategory includes five categories: Dos, Normal, Probing, R2L, and U2R.

Table 8. Multiclassification ablation experiment

Models	Accuracy	F1 Score	Precision	Recall
CNN	97.70%	86.71%	89.17%	86.72%
LSTM	97.02%	76.57%	77.68%	75.75%
CNN-LSTM	98.50%	91.51%	90.58%	93.08%

Figure 5. Training curves of the CNN-LSTM

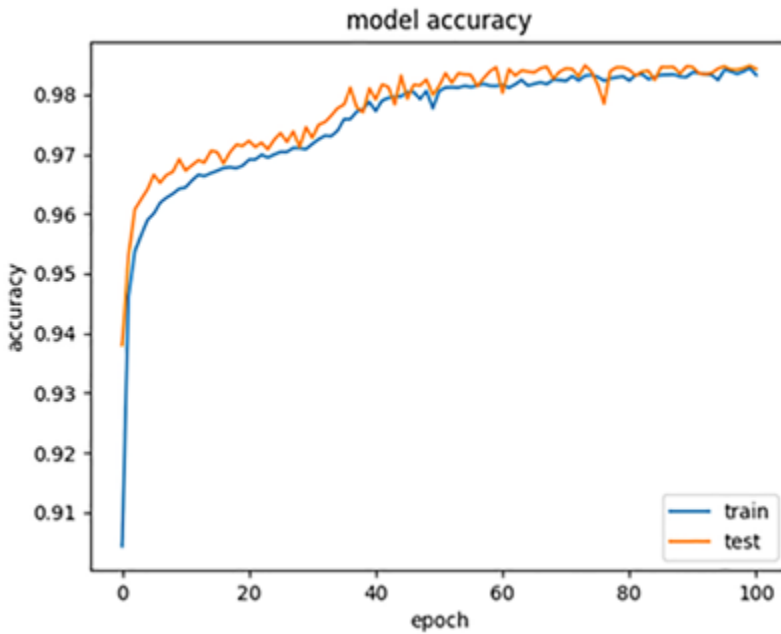
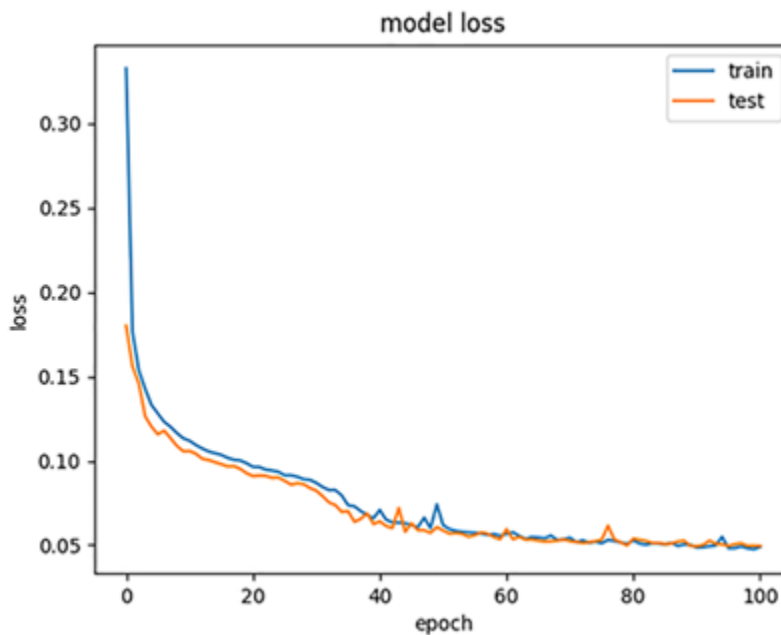


Figure 6. Loss curves of the CNN-LSTM



After comparison, the results evidence that, in the same environment, the authors' model performs better than a single traditional model in all indicators in the binary classification experiment on the NSL-KDD dataset, and the overall performance is the best. In the multiclassification experiment,

Table 9. Binary classification accuracy of different models (KDD)

Models	Accuracy	F1 Score	Precision	Recall
Random forest	95.44%	95.41%	95.72%	95.32%
naïve Bayes	86.95%	86.91%	87.01%	86.88%
SVM	89.94%	89.88%	90.32%	89.79%
MLP	96.10%	96.10%	96.10%	96.15%
CNN-LSTM	98.51%	98.50%	98.50%	98.51%

Table 10. Multiclassification accuracy of different models (KDD)

Models	Accuracy	F1 Score	Precision	Recall
Random Forest	94.53%	81.77%	93.91%	76.59%
Naive Bayes	83.60%	62.18%	60.55%	72.58%
SVM	89.76%	54.44%	61.33%	53.45%
MLP	95.15%	72.89%	89.17%	70.25%
CNN-LSTM	98.50%	91.51%	90.58%	93.08%

the accuracy rate, F1 score, and recall rate were all better than other traditional models, but the precision of 90.58% was slightly lower than the precision of 93.91% of the random forest algorithm. Table 11 shows the multiclassification performance comparison between NSL and KDD and existing research models. The results showed that the model has higher accuracy, F1 score, and recall rate on NSL-KDD. The precision of S-NDAE and SAVAER-DNN is higher than this model, reaching 100% and 95.98%, respectively. However, the overall indicators of this model are guaranteed to be 90% or above, and the overall performance is the best.

In addition, the authors chose the industrial internet traffic dataset and the natural gas pipeline dataset to validate the selected model. The dataset has a total of 274,628 samples, divided into 214,580 normal data and 60,048 abnormal data. Each piece of data corresponds to 20 characteristic attributes, and about 75% of the data are missing 10 or 11 characteristic attribute values. Due to the large number of missing values and missing dimensions, it is inconvenient to use related interpolation filling methods such as mean interpolation or similar mean interpolation. Because a large amount of data is complemented by interpolation or padding, it is difficult to effectively express the authenticity and validity of the data. Therefore, the authors performed the missing value removal process on the Gas dataset and removed data with missing attribute values greater than or equal to 10. After the data filtering process, the remaining data lacked the value of the attribute column “pressure

Table 11. Multiclassification accuracy of different models (KDD)

Paper	Models	Accuracy	F1 Score	Precision	Recall
Shone et al., 2018	S-NDAE	85.42%	87.37%	100%	85.42%
Yang et al., 2020	SAVAER-DNN	89.36%	90.08%	95.98%	84.86%
Alfoudi et al., 2022	DBSCAN	86.82%	87.87%	88.95%	86.82%
Yang et al., 2023	NRS-SSA	78.75%	74.9%	80.41%	78.75%
Model in this paper	CNN-LSTM	98.50%	91.51%	90.58%	93.08%

measurement,” so the authors deleted this column and ignored the impact of this column on the data. After the above operations, they finally obtained 64,100 complete pieces of data, including four attack categories, namely Normal, MSCl, MPCl, and Dos. The authors performed the same data preprocessing on the cleaned data to obtain a training set with a ratio of 0.8 and a test set with a ratio of 0.2. Figure 7 illustrates the generated distribution diagram, where 0-7 represent different peaks, and the corresponding sample categories are Normal, NMRI, CMRI, MSCl, MPCl, MFCl, Dos, and Reconnaissance. Then, the authors replaced the corresponding input and output, performed training fitting, and compared with other models. Table 12 shows the results of multiclassification experiments on the Gas dataset.

The comparison results show that the CNN-LSTM intrusion detection classification algorithm the authors proposed in this paper has better classification performance in the Gas industrial network dataset. The accuracy and precision of the model were 97.09% and 97.01%, respectively, which are higher than the existing models. The F1 score of 93.75% was slightly lower than that of the DNN-DT model (93.83%). The recall rate was 90.84%, which is lower than that of the DNN-DT model (93.72%) and of the LSTM-AE-OCSVM model (96.28%). In summary, the overall performance of the model was better, which proves the effectiveness and portability of the proposed method.

Figure 7. Diagram of balanced KDE of gas dataset

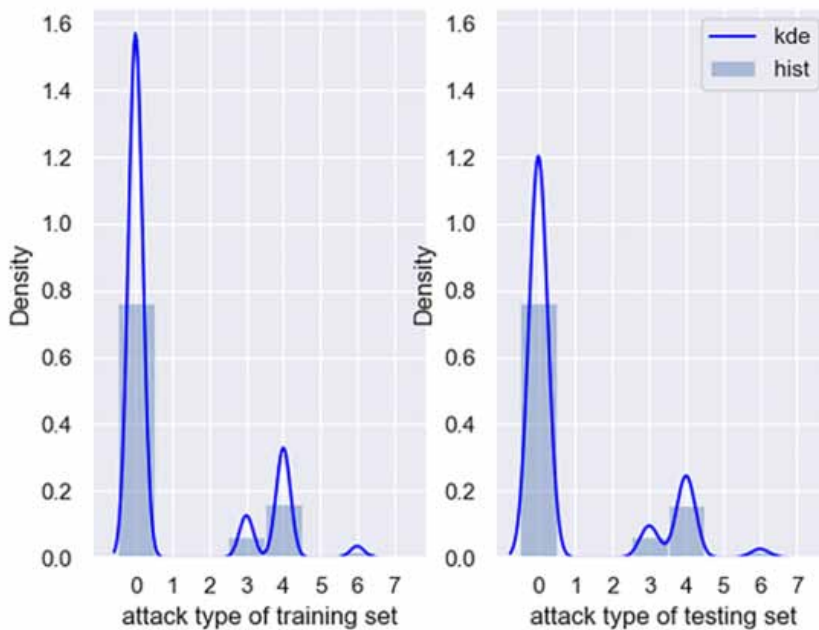


Table 12. Multiclassification accuracy of different models (gas)

Paper	Models	Accuracy	F1 Score	Precision	Recall
Feng et al., 2017	LSTM	92%	85%	94%	78%
Al-Abassi et al., 2020	DNN-DT	96%	93.83%	94.63%	93.72%
Chang et al., 2019	CAE	95.53%	89.08%	95.43%	83.52%
Ha et al., 2022	LSTM-AE-OCSVM	-	90.12%	84.70%	96.28%
Model in this paper	CNN-LSTM	97.09%	93.75%	97.01%	90.84%

CONCLUSION

In this paper, the authors proposed an industrial Internet network intrusion detection method integrating CNN and LSTM. The researchers tested the performance of the proposed model on the NSL-KDD dataset. They performed the KDE on the dataset to analyze the data distribution, processed the relevant features through the Pearson correlation coefficient, and continuously convoluted the local low-level features into high-level features to cover more data information. Then, they used the LSTM to extract timing features, better consider the impact of timing, and reduce the FP rate. Finally, they compared the proposed model with machine learning algorithms (e.g., SVM, random forest, MLP, and naive Bayes) and existing models. In the KDD dataset, the proposed model has higher precision and recall than the machine learning algorithm; compared with the existing model, it has better classification performance, but the precision index is poor, and the overall index remains at more than 90%. In the Gas dataset, the proposed model is almost the same as the existing model; it performs better in accuracy and precision, slightly worse in F1 and recall, and the overall index can also be maintained above 90%. Overall, the performance of the model on KDD and Gas fully demonstrates its effectiveness and generalization. At this stage, the authors deployed the proposed algorithm model in the laboratory local area network (LAN) to realize real-time traffic detection in the laboratory LAN. However, due to insufficient abnormal data in the LAN dataset, the detection effect is mediocre. In the future, the authors will consider combining the concept of cloud-edge collaboration to optimize learning performance and efficiency and improve on this model to make it more applicable to actual industrial Internet scenarios.

COMPETING INTERESTS

All the authors of this article declare there are no competing interests.

FUNDING AGENCY

The work was sponsored by the National Natural Science Foundation of China Grant No. 61972133 and No. 12101195, Project of Leading Talents in Science and Technology Innovation in Henan Province Grant No. 204200510021, Henan Province Natural Science Fund Grant No. 202300410146 and No. 232300420148, Henan Province Key Scientific and Technological Projects Grant No. 222102210177, No. 212102210383, No. 202102210162, and No. 222102210072.

REFERENCES

- Abu-Khzam, F. N., Abd El-Wahab, M. M., Haidous, M., & Yosri, N. (2022). Learning from obstructions: An effective deep learning approach for minimum vertex cover. *Annals of Mathematics and Artificial Intelligence*, 1–12. doi:10.1007/s10472-022-09813-2
- Al-Abassi, A., Karimipour, H., Dehghantanha, A., & Parizi, R. (2020). An ensemble deep learning-based cyber-attack detection in industrial control system. *IEEE Access : Practical Innovations, Open Solutions*, 8, 83965–83973. doi:10.1109/ACCESS.2020.2992249
- Al Sobhahi, R., & Tekli, J. (2022). Comparing deep learning models for low-light natural scene image enhancement and their impact on object detection and classification: Overview, empirical evaluation, and challenges. *Signal Processing Image Communication*, 109, 116848. doi:10.1016/j.image.2022.116848
- Alanazi, M., Mahmood, A., & Chowdhury, M. J. M. (2022). SCADA vulnerabilities and attacks: A review of the state-of-the-art and open issues. *Computers & Security*, 125, 103028. doi:10.1016/j.cose.2022.103028
- Alfoudi, A. S., Aziz, M. R., Alyasseri, Z. A. A., Alsaeedi, A. H., Nuijaa, R. R., Mohammed, M. A., Abdulkareem, K. H., & Jaber, M. M. (2022). Hyper clustering model for dynamic network intrusion detection. *IET Communications*, 10, 1–13. doi:10.1049/cmu2.12523
- Alghamdi, R., & Bellaiche, M. (2023). A cascaded federated deep learning based framework for detecting wormhole attacks in IoT networks. *Computers & Security*, 125, 103014. doi:10.1016/j.cose.2022.103014
- Anthi, E., Williams, L., Rhode, M., Burnap, P., & Wedgbury, A. (2021). Adversarial attacks on machine learning cybersecurity defences in industrial control systems. *Journal of Information Security and Applications*, 58, 102717. doi:10.1016/j.jisa.2020.102717
- Chang, C. P., Hsu, W. C., & Liao, I. E. (2019). Anomaly detection for industrial control systems using k-means and convolutional autoencoder. In *Proceedings of the 2019 International Conference on Software, Telecommunications, and Computer Networks (SoftCOM)* (pp. 1–6). IEEE. doi:10.23919/SOFTCOM.2019.8903886
- Chhetri, S. R., Faezi, S., Rashid, N., & Faruque, M. A. (2018). Manufacturing supply chain and product lifecycle security in the era of industry 4.0. *Journal of Hardware and Systems Security*, 2(1), 51–68. doi:10.1007/s41635-017-0031-0
- Chui, K. T., Gupta, B. B., Alhalabi, W., & Alzahrani, F. S. (2022). An MRI scans-based Alzheimer's disease detection via convolutional neural network and transfer learning. *Diagnostics (Basel)*, 12(7), 1531. doi:10.3390/diagnostics12071531 PMID:35885437
- Feng, C., Li, T., & Chana, D. (2017). Multi-level anomaly detection in industrial control systems via package signatures and LSTM networks. In *Proceedings of the 2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)* (pp. 261–272). IEEE. doi:10.1109/DSN.2017.34
- García, S., Derrac, J., Triguero, I., Carmona, C. J., & Herrera, F. (2012). Evolutionary-based selection of generalized instances for imbalanced classification. *Knowledge-Based Systems*, 25(1), 3–12. doi:10.1016/j.knsys.2011.01.012
- Gauthama Raman, M. R., & Mathur, A. P. (2022). AICrit: A unified framework for real-time anomaly detection in water treatment plants. *Journal of Information Security and Applications*, 64, 103046. doi:10.1016/j.jisa.2021.103046
- Gupta, B. B., Joshi, R. C., & Misra, M. (2009). Defending against distributed denial of service attacks: Issues and challenges. *Information Security Journal: A Global Perspective*, 18(5), 224–247.
- Ha, D. T., Hoang, N. X., Hoang, N. V., Du, N. H., Huong, T. T., & Tran, K. P. (2022). Explainable anomaly detection for industrial control system cybersecurity. *IFAC-PapersOnLine*, 55(10), 1183–1188.
- Hammad, M., Alkinani, M. H., Gupta, B. B., & Abd El-Latif, A. (2021). Myocardial infarction detection based on deep neural network on imbalanced data. *Multimedia Systems*, 28, 1373–1385.
- Hassan, M. M., Gumaedi, A., Alsanad, A., Alrubaian, M., & Fortino, G. (2020). A hybrid deep learning model for efficient intrusion detection in big data environment. *Information Sciences*, 513, 386–396. doi:10.1016/j.ins.2019.10.069

- Helwan, A., Ma'aitah, M. K. S., Uzelaltinbulat, S., Sonyel, B., Altobel, M. Z. Z., & Darwish, M. (2021). Stacked autoencoders deep learning approach for left ventricular localization in magnetic resonance slices. In *Proceedings of the International Conference on Emerging Technologies and Intelligent Systems: ICETIS 2021* (pp. 225-234). Springer.
- Islam, M. Z., Islam, M. M., & Asraf, A. (2020). A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images. *Informatics in Medicine Unlocked*, 20, 100412. doi:10.1016/j.imu.2020.100412 PMID:32835084
- Kou, L., Ding, S., Rao, Y., Xu, W., & Zhang, J. (2022). A lightweight intrusion detection model for 5G-enabled industrial Internet. *Mobile Networks and Applications*, 27(6), 2449–2458. doi:10.1007/s11036-021-01891-6
- Kravchik, M., & Shabtai, A. (2021). Efficient cyber attack detection in industrial control systems using lightweight neural networks and PCA. *IEEE Transactions on Dependable and Secure Computing*, 19(4), 2179–2197. doi:10.1109/TDSC.2021.3050101
- Li, Y., Xu, Y., Liu, Z., Hou, H., Zheng, Y., Xin, Y., Zhao, Y., & Cui, L. (2019). Robust detection for network intrusion of industrial IoT based on multi-CNN fusion. *Measurement*, 154, 107450. doi:10.1016/j.measurement.2019.107450
- Malik, S., Amin, J., Sharif, M., Yasmin, M., Kadry, S., & Anjum, S. (2022). Fractured elbow classification using hand-crafted and deep feature fusion and selection based on whale optimization approach. *Mathematics*, 10(18), 3291. doi:10.3390/math10183291
- Mishra, A., Gupta, B. B., & Joshi, R. C. (2011). A comparative study of distributed denial of service attacks, intrusion tolerance, and mitigation techniques. In *Proceedings of the 2011 European Intelligence and Security Informatics Conference* (pp. 286–289). IEEE. doi:10.1109/EISIC.2011.15
- Morris, T., & Gao, W. (2014). Industrial control system traffic data sets for intrusion detection research. In *Proceedings of the Critical Infrastructure Protection VIII: 8th IFIP WG 11.10 International Conference, ICCIP 2014* (pp. 65–78). Springer. doi:10.1007/978-3-662-45355-1_5
- Morris, T. H., Thornton, Z., & Turnipseed, I. (2015). Industrial control system simulation and data logging for intrusion detection system research. In *Proceedings of the 7th Annual Southeastern Cyber Security Summit*, 2015 (pp. 3–4). Uah.edu.
- Ren, J., Zhang, Y., Zhang, B., & Li, S. (2022). Classification method of industrial Internet intrusion detection based on feature selection. *Journal of Computer Research and Development*, 59(5), 1148–1159.
- Sayour, M. H., Kozhaya, S. E., & Saab, S. S. (2022). Autonomous robotic manipulation: Real-time, deep-learning approach for grasping of unknown objects. *Journal of Robotics*, 2022, 2585656. doi:10.1155/2022/2585656
- Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (2018). A deep learning approach to network intrusion detection. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2(1), 41–50. doi:10.1109/TETCI.2017.2772792
- Tavallae, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. In *Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications* (pp. 1-6). IEEE. doi:10.1109/CISDA.2009.5356528
- Yang, Y., Zheng, K., Wu, B., Yang, Y., & Wang, X. (2020). Network intrusion detection based on supervised adversarial variational auto-encoder with regularization. *IEEE Access : Practical Innovations, Open Solutions*, 8, 42169–42184. doi:10.1109/ACCESS.2020.2977007
- Yang, Z., Liu, Z., Zong, X., & Wang, G. (2023). An optimized adaptive ensemble model with feature selection for network intrusion detection. *Concurrency and Computation*, 35(4), e7529. doi:10.1002/cpe.7529
- Zhang, C., Chen, Y., Meng, Y., Ruan, F., Chen, R., Li, Y., & Yang, Y. (2021). A novel framework design of network intrusion detection based on machine learning techniques. *Security and Communication Networks*, 2021, 6610675. doi:10.1155/2021/6610675
- Zhang, C., Sun, G., Fang, Z., Zhou, P., Pan, P., & Cong, J. (2018). Caffeine: Toward uniformed representation and acceleration for deep convolutional neural networks. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 38(11), 2072–2085. doi:10.1109/TCAD.2017.2785257

Jinhai Song is currently pursuing his master's degree with College of Information Engineering, Henan University of Science and Technology and Henan International Joint Laboratory of Cyberspace Security Applications, Luoyang, China. His research interests focus on cyber security and industrial Internet security.

Zhiyong Zhang (Senior Member, IEEE) received his master's and Ph.D. degrees in Computer Science from Dalian University of Technology and Xidian University, P. R. China, respectively. He gained a post-doctoral fellowship at School of Management, Xi'an Jiaotong University, China. Nowadays, he is Director of Henan International Joint Laboratory of Cyberspace Security Applications, Vice-Dean of College of Information Engineering, and full-time Henan Province Distinguished Professor at Henan University of Science and Technology, China. He is also a visiting professor of Computer Science Department of Iowa State University. His research interests include cyber security and frontier computing, social big data security and privacy, social computing and social intelligence. He has published over 150 scientific papers and edited six books in the above research fields; he also holds 20 authorized patents. He is Chair of IEEE MMTC DRMIG, IEEE Systems, Man, Cybernetics Society Technical Committee on Soft Computing, World Federation on Soft Computing Young Researchers Committee, Committeeman of China National Audio, Video, Multimedia System and Device Standardization Technologies Committee. Besides, he is editorial board member and associate editor of Multimedia Tools and Applications (Springer), Human-Centric Computing and Information Sciences (Springer), IEEE Access (IEEE), Neural Network World, EURASIP Journal on Information Security (Springer), leading guest editor or coguest Editor of Applied Soft Computing (Elsevier), Computer Journal (Oxford) and Future Generation Computer Systems (Elsevier). In addition, he is Chair/Cochair and TPC Member for numerous international conferences/workshops on digital rights management and cloud computing security.

Kejing Zhao received her master's degrees in Mathematics at Henan University of Science and Technology, Luoyang, China. She is currently pursuing her Ph.D. degree with the School of Information Engineering, Henan University of Science and Technology and Henan International Joint Laboratory of Cyberspace Security Applications, Luoyang, China. Her research interests include industrial Internet security, industrial situational analytics, and industrial big data.

Qinhai Xue is currently pursuing his master's degree with the College of Information Engineering, Henan University of Science and Technology and Henan International Joint Laboratory of Cyberspace Security Applications, Luoyang, China. His research interests focus on cyber security and blockchain technology.

Brij B. Gupta is working as Director of International Center for AI and Cyber Security Research and Innovations, and Full Professor with the Department of Computer Science and Information Engineering (CSIE), Asia University, Taiwan. In more than 17 years of his professional experience, he published over 500 papers in journals/conferences including 35 books and 11 Patents with over 21,000 citations. He has received numerous national and international awards including Canadian Commonwealth Scholarship (2009), Faculty Research Fellowship Award (2017), MeitY, GoI, IEEE GCCE outstanding and WIE paper awards and Best Faculty Award (2018 & 2019), NIT KKR, respectively. Prof. Gupta was recently selected for 2022 Clarivate Web of Science Highly Cited Researchers in Computer Science. He was also selected in the 2022, 2021 and 2020 Stanford University's ranking of the world's top 2% scientists. He is also a visiting/adjunct professor with several universities worldwide. He is also an IEEE Senior Member (2017) and also selected as 2021 Distinguished Lecturer in IEEE CTSoc. Dr Gupta is also serving as Member-in-Large, Board of Governors, IEEE Consumer Technology Society (2022-2024). Prof Gupta is also leading IJSWIS, IJSSCI, STE and IJCAC as Editor-in-Chief. Moreover, he is also serving as lead-editor of a Book Series with CRC and IET press. He also served as TPC members in more than 150 international conferences also serving as Associate/Guest Editor of various journals and transactions. His research interests include information security, Cyber physical systems, cloud computing, blockchain technologies, intrusion detection, AI, social media and networking.