Proceedings of the 2020 39th Chinese Control Conference (CCC), 27-29 July, Shenyang, China
DOI:10.23919/CCC50068.2020.9188872

A Novel Industrial Intrusion Detection Method based on Threshold-optimized CNN-BiLSTM-Attention using ROC Curve

Mindi Lan[1],Jun Luo[1],Senchun Chai[1],Runqi Chai[2],Chen Zhang[1],Baihai Zhang[1]

1. School of Automation, Beijing Institute of Technology, Beijing 100081, P. R. China
E-mail: chaisc97@bit.edu.cn

2. School of Aerospace, Transport and Manufacturing, Cranfield University, Bedfordshire MK43 0AL, United Kingdom
E-mail: 57545813@qq.com

Abstract: In recent years, many researchers have proposed many intrusion detection methods to protect the industrial network. However, there are two existing problems among them: one is that they only consider the overall accuracy rate (AC) while ignoring the problem of class imbalance; another one is that they have considered the problem of class imbalance, but the detection rate (DR) is low and false positive rate (FR) is high for minority classes. In order to improve AC and DR of minority classes, we propose a method called threshold-optimized CNN-BiLSTM-Attention that combines CNN-BiLSTM-Attention model, with threshold modification method based on receiver operating characteristic (ROC) curve. In this method, we use CNN-BiLSTM-Attention model as a classifier and modify threshold of the classifier through ROC curve. To evaluate the proposed method, we have performed experiments on the standard industrial data set. And the experimental results show that the proposed method can improve AC and the DR of minority classes at low FR, which is better than other intrusion detection methods.

Key Words: Industrial intrusion detection, Class imbalance, CNN-BiLSTM-Attention, Threshold modification, ROC curve

1 Introduction

According to the proposal of "Made in China 2025" strategy, industrial control systems begin to transfer from physically isolated networks to open networks [1]. Although industrial control system is developing towards networking and intelligence, it gradually exposes its vulnerability in the network for its mutual integration with the internet, which results in a series of attacks against industrial control networks. Especially, the attacks on Iranian nuclear power plants, namely Stuxnet[2], marked a turning point in the history of industrial cyber security. Since then, the attacks of industrial control system have emerging in endlessly, which seriously affected the national security and social stability[3]. Therefore, it is necessary to design related anomaly detection system specifically for industrial control systems.

Since machine learning algorithms have the capability to acquire knowledge from a number of new data, they are widely used in industrial intrusion detection systems[4]. Considering about the data characteristics of industrial control system[5], which have fewer abnormal attack samples, high dimensions and so on, many traditional machine learning algorithms have been proposed. For example, Wang et al.[6] proposed an industrial control intrusion detection method combining principal component analysis (PCA) and PSO-SVM, they firstly applied PCA to reduce the dimension of the obtained intrusion data, and then used PSO to optimize SVM parameters to obtain the optimal SVM industrial intrusion detection model, which improves the accuracy of detection. But they just consider the problem of high-dimensional data, and ignore the problem of class imbalance. In addition, other traditional machine learning algorithms mentioned by Beaver et al[7], such as Naive Bayes and SVM, and the decision tree proposed by Moon et al[8], can perform well in AC of intrusion detection but did not take the problem of class imbalance into account.

Furthermore, due to deep learning algorithms have strong capabilities of feature extraction, they are widely applied to industrial intrusion detection[9–11]. For example, an industrial intrusion detection method based on the Long and Short-Term Memory(LSTM) algorithm was proposed by Bing et al[9], which was applied to solve the problem of High-dimensional network intrusion data and class imbalance, and it obtained high AC by reducing the number of data set. Furthermore, since Convolutional Neural Network(CNN) is widely used in image recognition for its excellent feature extraction performance, it has been used in intrusion detection of standard Information and Communication Technologies (ICTs) by Wu et al[10]. Meanwhile, it has also been applied to detect cyber attacks of the industrial control system by Kravchik et al[11], which enhances the AC of intrusion detection of Industrial Control System (ICS).

In the methods mentioned above, there are two main problems: one is that they only consider the overall accuracy rate (AC) while ignoring the problem of class imbalance; another one is that they have considered the problem of class imbalance, but the detection rate (DR) is low and false positive rate (FR) is high for minority classes. Thus, we propose a method called Threshold-optimized CNN-BiLSTM-Attention to enhance the AC and the DR of minority classes in this paper.

The rest of this paper is formed as follows: the proposed method is described in Section 2, which includes preprocessing of data sets, threshold-optimized CNN-BiLSTM-Attention and evaluation metrics. In Section 3, we perform two sets of experiments on the standard industrial data set respectively. Finally, we give the conclusions in Section 4.

2 Proposed Method

The threshold-optimized CNN-BiLSTM-Attention used in the intrusion detection of industrial control system can be implemented following three steps, as shown in Fig. 1. First of all, although the standard industrial data set we used is numerically processed, we need to do some other preprocessing on the data set to make it adaptive to the threshold-

optimized CNN-BiLSTM-Attention model. And then, the training data set is used to train threshold-optimized CNN-BiLSTM-Attention until it obtains the most optimal parameters. Finally, we utilize the testing data set to evaluate the quality of the trained threshold-optimized CNN-BiLSTM-Attention.

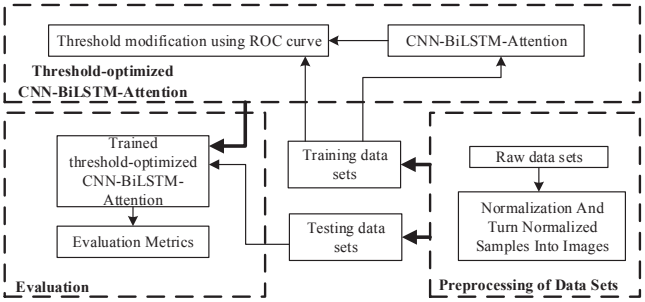

Fig. 1: Industrial intrusion detection based on threshold-optimized CNN-BiLSTM-Attention

2.1 Preprocessing of Data Sets

One standard industrial data set is used in this paper, which contains 26 features, such as IP address, message length, and function code, and a label. Although the data set we used is numerically processed, it is not effective if we directly apply it to train the proposed model. Consequently, it is necessary to preprocess the data set, making it standard training data set and testing data set so as to apply to the model. As shown below, there are three steps of data preprocessing.

2.1.1 One-hot encoding of Data Labels

There are 8 types of data samples in the industrial control intrusion detection standard data set, including one normal data and 7 different types of attack data, which means its data labels are divided into 8 types from 0-7. In order to make it suitable for the trained model, the one-hot encoding is used for the data labels. The dimension of one hot encoding is determined by the number of types of the data labels, and only one bit is valid in each vector, represented by 1, others are 0.

2.1.2 Normalization of Data Features

Although the data set we used is numerically processed, the range of the numerical values of the data features is quite different, which tends to affect the accuracy of the model we proposed. Therefore, normalizing the feature values is needed, which converts the different numerical values of the features into range of [0, 1]. We use the Min-Max standardization for normalization, and it can be expressed by formula (1).

$$\hat{X}_{ij} = \frac{X_{ij} - \min(X_j)}{\max(X_j) - \min(X_j)} \tag{1}$$

Where X_{ij} represents value of feature j in sample i of the data sets, $\max(X_j)$ means the maximum value of feature j of the data sets, and $\min(X_j)$ is the minimum value of feature j of the data sets, meanwhile, $\hat{X}_{ij}$ is the normalized value of feature j in sample i of data sets.

2.1.3 Turn Normalized Samples Into Images

Considering that the proposed method is more efficient for image processing, we transfer the normalized data into grayscale image. There are 26-dimensional features in the data set mentioned above, and in order to preserve all the features of each sample in the data set, we fill it with 10 zeros, which makes it suitable for a 6×6 grayscale image. And we randomly select 10 samples from the data set, whose categories are CMRI, Normal, Normal, Recon, CMRI, Recon, MPCI, Normal, MPCI, CMRI from top to bottom and then left to right, and then transfer them into grayscale images. As in Fig. 2, the pictures of the same class are almost the same, but there are great differences between images of different classes.

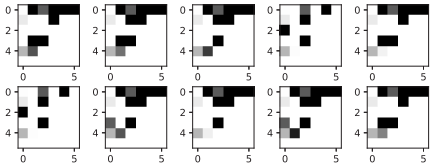

Fig. 2: Grayscale images of 10 samples

2.2 Threshold-optimized CNN-BiLSTM-Attention

The threshold-optimized CNN-BiLSTM-Attention proposed in this paper is based on CNN-BiLSTM-Attention and the threshold modification implemented by ROC curves. The CNN-BiLSTM-Attention is actually a CNN-BiLSTM model structure that introduces the attention mechanism, whose structure is shown in Fig. 3.

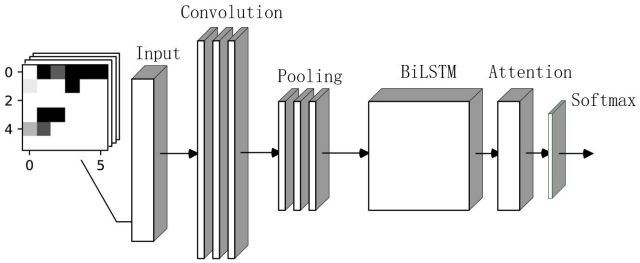

Fig. 3: Structure of CNN-BiLSTM model

Firstly, the input of CNN is a number of 6×6 grayscale intrusion image samples, and the images are then passed through CNN, BiLSTM and attention layer in order. Finally, the output need to put into the softmax layer. In this way, we can get better classification results.

From Table 3, we can find that there are minority classes, such as NMRI, MSCI, MFCI and Dos. Thus, we utilize the threshold modification method using ROC curve, to reduce FR and improve DR. As shown in Fig. 4, we firstly suppose that there are five classes $\{A, B, C, D, E\}$ in the data set, and the number of these five classes is recorded as $\{n_A, n_B, n_C, n_D, n_E\}$, satisfying formula (2), and $\{E\}$ means minority classes of the intrusion data set. And then,

we take $P0$ in Fig. 4 as the training threshold, which is represented by formula (3), P^i is the output of the CNN-BiLSTM-Attention model, which is shown in formula (4).

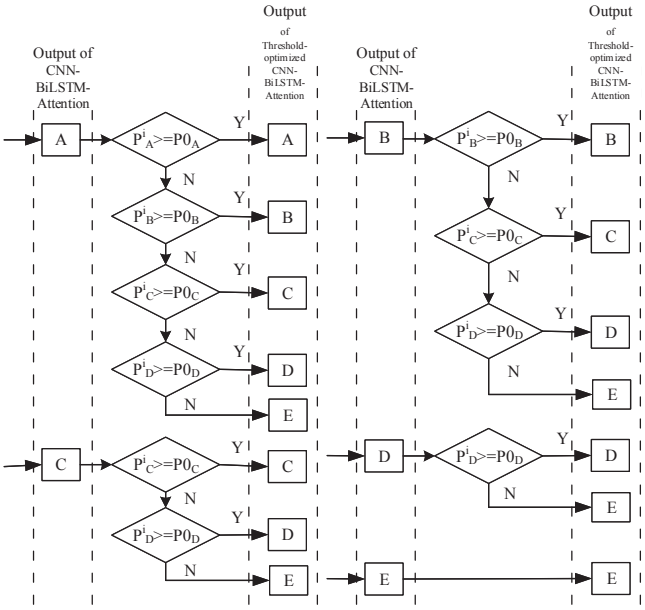

Fig. 4: Process of threshold modification

$$n_A > n_B > n_C > n_D > n_E \tag{2}$$

$$P0 = [P0_A, P0_B, P0_C, P0_D, P0_E] \tag{3}$$

$$P^i = \left[P_A^i, P_B^i, P_C^i, P_D^i, P_E^i\right] \tag{4}$$

When the training samples are not balanced, the output of the classifier is always biased towards a large number of samples, resulting in a low DR of minority class. In order to solve this problem, we introduce a ROC curve to set a new threshold, which is not affected by the number of samples, so as to ensure the classification accuracy of the minority class. We can see from Fig. 4 that when the prediction result of the CNN-BiLSTM-Attention is class A, we compare the optimal threshold $P0_A$ with the predicted probability P_A^i of class A on the output of the CNN-BiLSTM-Attention. If P_A^i is not less than $P0_A$, the output of Threshold-optimized CNN-BiLSTM-Attention is class A. Otherwise, the optimal threshold $P0_B$ of class B is used to compare with the the predicted probability P_B^i of class A on the output of the CNN-BiLSTM-Attention, and the output of Threshold-optimized CNN-BiLSTM-Attention is class B if P_B^i is not less than $P0_B$. And we use the same comparison method as above to get the output of Threshold-optimized CNN-BiLSTM-Attention, including class C, D, E. Similarly, when the prediction result of the CNN-BiLSTM-Attention is class B, C, D, we firstly compare $P_B^i/P_C^i/P_D^i$ to the optimal threshold $P0_B/P0_C/P0_D$, and when the former is less than the latter, continuing to compare with the rest samples as the method mentioned above. Finally, when the prediction result of the CNN-BiLSTM-Attention is the minority class E, the prediction of the the CNN-BiLSTM-Attention is equal to the output of Threshold-optimized CNN-BiLSTM-Attention. Also, we can see from Fig. 4 that the larger $P0$, the larger penalty for the classifier, so as to improve the DR of minority classes.

In this paper, we use ROC curve to find the optimal threshold $P0$ and give a proof. The ROC curve is a way to judge the performance of two classifiers[12]. The abscissa uses false positive rate (FPR) to represent specificity, and the ordinate uses true positive rate (TPR) to represent sensitivity. Related concepts are defined in Table 1, which mainly includes TP,FN,FP,TN. And FPR and TPR are represented as formula (5) and (6). And the ROC curve is used as a more balanced evaluation method, which takes the trade-offs between positive and negative samples into consideration. And the performance of the classifier will be best if $TPR = 1$ and $FPR = 0$.

Table 1: The Confusion Matrix for a Two-Class Classification Test

Real lable	Predicted label	
	Positive Samples	Negative Samples
Positive samples	TP	FN
Negative samples	FP	TN

$$FPR = \frac{FP}{FP + TN} = \frac{FP}{N} \tag{5}$$

$$TPR = \frac{TP}{TP + FN} = \frac{TP}{P} \tag{6}$$

We transfer the problem from a multi-class classification into multiple two-class classifications. And then use multiple ROC curves to obtain multiple thresholds for comparison. Table 2 is defined as a multi-class classification confusion matrix. We assume that m and n belong to $\{A, B, C, D, E\}$ and m is not equal to n. In table 2, TP_m means the number of positive samples correctly predicted, FN_m means the number of positive samples classified as negative samples, FP_m means the number of positive samples classified as zero samples, TN_m means the number of negative samples correctly predicted, and E_{mn} means the number of misclassified samples that m is classified as n. We can calculate the FN_A =E_{AB}+E_{AC} + E_{AD}+E_{AE}, and $FN_{B,C,D,E}$ can be obtained in the same way. Also, FP_A =E_{BA}+E_{CA} + E_{DA}+E_{EA}, and $FP_{B,C,D,E}$ can be obtained similarly. Then, we compute TN_A = S-FN_A-FP_A-TP_A, and $TP_{B,C,D,E}$ can be obtained similarly.

Table 2: The Confusion Matrix for a Multi-Class Classification Test

Real label	Predicted label				
	A	B	C	D	E
A	TP_A	E_{AB}	E_{AC}	E_{AD}	E_{AE}
B	E_{BA}	TP_B	E_{BC}	E_{BD}	E_{BE}
C	E_{CA}	E_{CB}	TP_C	E_{CD}	E_{CE}
D	E_{DA}	E_{DB}	E_{DC}	TP_D	E_{DE}
E	E_{EA}	E_{EB}	E_{EC}	E_{ED}	TP_E

Since TP_m, FP_m, TN_m, FN_m are obtained above, we use it to compute $TPR_m$and FPR_m as formula (7), (8)

$$TPR_m = \frac{TP_m}{TP_m + FN_m} \tag{7}$$

$$FPR_m = \frac{FP_m}{FP_m + TN_m} \tag{8}$$

In order to find $P0$ through the ROC curve, as shown in Fig. 5, we define the equation for circle C_m with (0, 1) as formula (9)

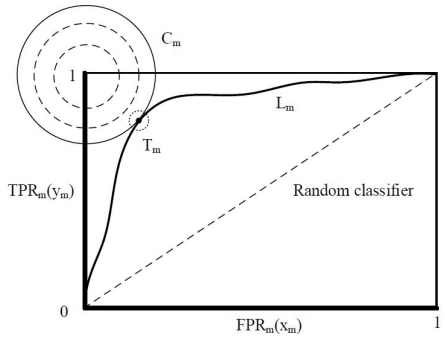

Fig. 5: Find the threshold by ROC curve

$$\begin{aligned}(1-y_m)^2+(0-x_m)^2=r_m^2\\ r_m\geq 0, x_m, y_m\in\mathbb{R}.\end{aligned} \tag{9}$$

When the circle C_m becomes larger with the radius r_m , we define the first intersection point of the circle with the ROC curve as T_m and T_m is not unique. And the distance from T_m to (0, 1) is D_m. Then, since the curve L_m is drawn by different (FPR_m, TPR_m), formula (10) holds true at T_m point.

$$\begin{aligned}(1-TPR_m)^2+(0-FPR_m)^2=D_m^2\\ \text{s.t. } D_m\geq 0, TPR_m, FPR_m\in[0,1].\end{aligned} \tag{10}$$

Formula (11) holds on curve L_m:

$$\begin{aligned}\min_{TPR_m,FPR_m}\quad (1-TPR_m)^2+(0-FPR_m)^2=D_m^2.\\ s.t.\quad 0\leq TPR_m\leq 1, 0\leq FPR_m\leq 1.\end{aligned} \tag{11}$$

Since the curve L_m is composed of different (FPR_m, TPR_m), formula (12) can be established.

$$\begin{aligned}t_{opt}^m=\arg\min_{t^m}\sqrt{(1-TPR_m)^2+(0-FPR_m)^2}.\\ s.t.\quad 0\leq t^m\leq 1, 0\leq t_{opt}^m\leq 1.\\ 0\leq TPR_m\leq 1, 0\leq FPR_m\leq 1.\end{aligned} \tag{12}$$

where t^m represents the threshold value corresponding to each point on the curve, and t_{opt}^m is the threshold value corresponding to the T_m point.

Generally, we obtain the optimal threshold from the ROC curve by solving the formula (13).

$$\begin{aligned}T_{opt}^m=\arg\max_{t^m} Y_m=\arg\max_{t^m}(TPR_m-FPR_m).\\ s.t.\quad 0\leq t^m\leq 1, 0\leq T_{opt}^m\leq 1, 0\leq Y_m\leq 1.\\ 0\leq TPR_m\leq 1, 0\leq FPR_m\leq 1.\end{aligned} \tag{13}$$

where Y_m is the Youden Index[24], and t_{opt}^m is the optimal threshold. According to it, we propose the following Theorem 1.

Theorem 1 *The optimal threshold t_{opt}^m is the threshold t_{opt}^m corresponding to the first intersection point T_m where the circle C_m and ROC curve intersect. It means that formula (12) is equal to formula (13), and t_{opt}^m = T_{opt}^m.*

Proof 2.1 *From Theorem 1, we know that formula (12) is equal to formula (13). Under this condition, we can get formula (14):*

$$\begin{aligned}&\min_{t^m}\sqrt{(1-TPR_m)^2+(0-FPR_m)^2}\\ &=\min_{t^m}(1-TPR_m)^2+(0-FPR_m)^2.\\ &s.t.\quad 0\leq t^m\leq 1.\\ &\qquad 0\leq TPR_m\leq 1, 0\leq FPR_m\leq 1.\end{aligned} \tag{14}$$

Also, we can get the formula (15) from formula (13):

$$TPR_m=FPR_m+Y_m. \tag{15}$$

Then, simultaneously with formula (14) and formula (15), we can get formula (16):

$$\begin{aligned}&\min_{t^m}(1-TPR_m)^2+(0-FPR_m)^2.\\ &=\min_{Y_m}[\min_{FPR_m}(1-Y_m-FPR_m)^2+(FPR_m)^2]\\ &=\min_{Y_m}[\min_{FPR_m}2(FPR_m)^2-2(1-Y_m)FPR_m+(1-Y_m)^2]\\ &\quad s.t.\quad 0\leq Y_m\leq 1, 0\leq FPR_m\leq 1.\end{aligned} \tag{16}$$

Furthermore, If and only if FPR_m=$\frac{(1-Y_m)}{2}$ $\in$ [0,$\frac{1}{2}$], we can obtain formula (17):

$$\begin{aligned}&\min_{Y_m}[\min_{FPR_m}2(FPR_m)^2-2(1-Y_m)FPR_m+(1-Y_m)^2]\\ &=\min_{Y_m}\frac{(1-Y_m)^2}{2}.\\ &\quad s.t.\quad 0\leq Y_m\leq 1, 0\leq FPR_m\leq 1.\end{aligned} \tag{17}$$

For $\frac{(1-Y_m)^2}{2}$ is monotonically decreasing in [0,1], formula (17) is equivalent to the following formula (18):

$$\begin{aligned}\max_{t^m} Y_m.\\ s.t.\quad 0\leq t^m\leq 1.\end{aligned} \tag{18}$$

Thus, T_m is the point of the optimal threshold, and we generally use the threshold t_{opt}^m corresponding to T_m point as the training threshold $P0_m$. And it can be adjusted in real situation as shown in formula (19).

$$P0_m\approx t_{opt}^m \tag{19}$$

2.3 Evaluation Metrics

In order to evaluate the threshold-optimized CNN-BiLSTM-Attention more objectively, we introduce the confusion matrix method depicted in table 2. And we use AC, DRs and FRs as the evaluation metrics, which can be calculated from the values in table 2 and defined as formula (20), (21) and (22).

$$AC = \sum_{m} \frac{TP_m}{S} \tag{20}$$

$$DR_m = TPR_m \tag{21}$$

$$FR_m = FPR_m \tag{22}$$

From formula (20), (21) and (22), we can conclude that if the intrusion detection method can obtain larger AC, DRs and smaller FRs, it is state-of-the-art.

3 Experiment And Results Analysis

This paper uses the standard industrial data set proposed by Mississippi State University in 2014[13]. The data has been numerically processed, and it can be divided into 8 classes, including Normal, Nave Malicious Response Injection (NMRI), Complex Malicious Response Injection (CMRI), Malicious State Command Injection (MSCI), Malicious Parameter Command Injection (MPCI), Malicious Function Code Injection (MFCI), Denial of Service (DoS) and Reconnaisance (Recon). Meanwhile, the data set has 26 features, such as IP address and message length, and a label. The distribution of it is shown in table 3. As we can see from the table 3, among 8 classes of the data set, normal class has the most samples, and other four classes, NMRI, MSCI, MFCI, DOS, have a smaller number of samples, especially MFCI, only 563, which refer to $\{E\}$ mentioned in section2.2.

Table 3: Data Distribution

The Standard Industrial Intrusion Data Set			
Attack Name	Training Data	Testing Data	Label
Normal	48925	7750	0
NMRI	2210	553	1
CMRI	12373	3093	2
MSCI	626	156	3
MPCI	6109	1528	4
MFCI	458	105	5
DOS	1470	252	6
Recon	5444	1361	7
Total	77615	14798	−−

The proposed algorithm is implemented by Tensorflow, Ubuntu 16.04, CPU computing and 10G running memory. We set the size of the convolution kernel as 5×5, whose step size is 1, and the number of the hidden layer is 1. In addition, we set the number of hidden layer units as 128 and the hidden layer dropout as 0.5. Also, we set the learning rate as 0.01 and the time step of BiLSTM as 6. Finally, 128 batch sizes and 5 epochs are set.

3.1 Effect of Threshold Modification Using ROC Curve

The first experiment is designed for demonstrating effectiveness of threshold modification using ROC curve. Therefore, we firstly choose threshold-optimized CNN-BiLSTM-Attention (TO-CBLA) which is proposed in this paper and CNN-BiLSTM-Attention (CBLA) for comparison on the same data set. Since paper[10] proposed a method called CNN based on modifying loss function weights (CMLW) performing well in detection, we apply modifying loss function weights to CNN-BiLSTM-Attention (CBLA-MLW) similarly, which is the second comparison algorithm. We can clearly see from Fig. 6 that the AC of TO-CBLA is the highest, which is better than other algorithms. Also, we can get a conclusion from Table 4 that TO-CBLA performs well in improving the DRs of minority categories (NMRI, DOS). Meanwhile, it can make sure that other classes obtain the higher DRs. But, compared with other methods, the FRs of minority classes(NMRI, DOS) obtained by TO-CBLA is slightly higher while the FRs of the overall classes are still low. In a word, the threshold modification using ROC curve can obviously enhance the DRs of the minority class and obtain the higher AC.

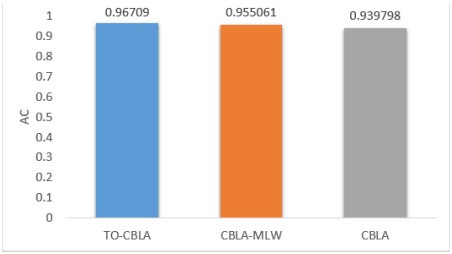

Fig. 6: ACs of TO-CBLA/CBLA-MLW/CBLA

3.2 State-of-the-art of the Threshold-optimized CNN-BiLSTM-Attention

The second experiment is designed for demonstrating the state-of-the-art of TO-CBLA. Therefore, we compare TO-CBLA with other five machine learning methods that have been used for industrial control intrusion detection, including CNN, BiLSTM, Decision Tree(DT), Naive Bayes(NB) and SVM. We can clearly see from Fig. 7 that the AC of TO-CBLA is higher than other five algorithms, which is as high as 96.7%. Also, we can get a conclusion from Table 5 that TO-CBLA can not only perform well in improving the DRs of minority categories (NMRI, DOS), but also make sure that the DRs of other classes is higher than other five methods. Meanwhile, compared with other five methods, the FRs obtained by TO-CBLA is more smaller. In brief, the TO-CBLA can obviously enhance the DRs of all the eight classes contained in the data set, especially the minority classes.

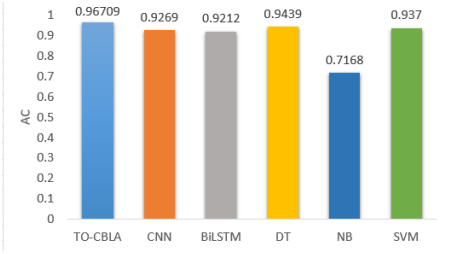

Fig. 7: ACs of TO-CBLA/CNN/BiLSTM/DT/NB/SVM

3.3 Results Analysis

From the above two sets of experiments, we can make an overall analysis of the results. Firstly, we can see from Fig. 6

Table 4: DRs and FRs of TO-CBLA/CBLA-MLW/CBLA Methods

Metrics	Method	Classes							
		Normal	NMRI	CMRI	MSCI	MPCI	MFCI	DOS	Recon
DRs	TO-CBLA	0.968	0.725	0.986	0.974	0.976	0.952	1	1
	CBLA-MLW	0.968	0.403	0.986	0.974	0.976	0.952	0.996	1
	CBLA	0.968	0	0.986	0.974	0.976	0.952	0.988	1
FRs	TO-CBLA	0.031	0.001	0.017	0.0002	0.004	0	0.0003	0
	CBLA-MLW	0.057	0.0006	0.017	0.0003	0.004	0	0.0002	0
	CBLA	0.091	0	0.017	0.0003	0.004	0	0.0003	0

Table 5: DRs and FRs of TO-CBLA/CNN/BiLSTM/DT/NB/SVM Methods

Metrics	Method	Classes							
		Normal	NMRI	CMRI	MSCI	MPCI	MFCI	DOS	Recon
DRs	TO-CBLA	0.968	0.725	0.986	0.974	0.976	0.952	1	1
	CNN	0.968	0	0.986	0.974	0.976	0.571	0.374	1
	BiLSTM	0.968	0	0.986	0.878	0.956	0.381	0.313	1
	DT	0.968	0.128	0.985	0.974	0.970	0.952	1	1
	NB	0.995	0	0	0	0.968	0.571	0	1
	SVM	0.968	0	0.986	0.974	0.976	0.952	0.817	1
FRs	TO-CBLA	0.031	0.001	0.017	0.0003	0.004	0	0.0003	0
	CNN	0.118	0	0.017	0.0003	0.004	0	0	0
	BiLSTM	0.127	0	0.017	0.0008	0.004	0.0002	0	0.0003
	DT	0.082	0	0.017	0.0003	0.004	0.00006	0.00006	0
	NB	0.589	0	0	0	0.004	0	0	1
	SVM	0.097	0	0.017	0.0003	0.004	0	0.0002	0

and Fig. 7 that the AC of TO-CBLA is the highest, which is up to 96.7%, and it indicates that we have improved AC obviously. Secondly, we can find that in Table 4 and Table 5, there are cases where DRs and FRs of the mentioned methods, such as CNN, and so on, are 0, especially in the minority classes. And the reason can be derived from formula(7), (8), (21), (22), which indicates that the FRs and DRs will be both 0 if the method does not detect the attack class. And we can see that TO-CBLA can significantly improve the DR of minority classes.

4 Conclusion

In order to improve AC and solve the problem of class imbalance, we propose a method called threshold-optimized CNN-BiLSTM-Attention. The experiments obtained on the standard industrial data set achieve higher DR and lower FR in detection of the minority classes. Meanwhile, when comparing with other intrusion detection methods, the proposed method is state-of-the-art.

Although it is proved by the experimental results that the proposed method performs well in the industrial intrusion detection, it requires longer training time when comparing with other methods existing in the literature. This can be further studied in the future.

References

[1] L. Hu, H. Li, Z. Wei, et al, Summary of Research on IT Network and Industrial Control Network Security Assessment, in *IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC). IEEE* , 2019: 1203-1210.

[2] R. Langner, Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Security Privacy*, 9(3): 49-51, 2011.

[3] T. Macaulay, B. Singer, *Cybersecurity for industrial control systems: SCADA, DCS, PLC, HMI, and SIS*. Auerbach Publications, 2016.

[4] M. S. Papa, A behavioral intrusion detection system for SCADA systems, Southern Methodist University, 2013.

[5] S. Yasakethu,J. Jiang, Intrusion detection via machine learning for SCADA system protection, in *Proceedings of the 1st International Symposium for ICS & SCADA Cyber Security research*. 2013: 101-5.

[6] H. Wang, Z. Yang, B. Yan, et al, Application of fusion PCA and PSO-SVM method in industrial control intrusion detection. *Sci. Technol*, 33(1): 80-85, 2017.

[7] JM. Beaver, RC. Borges-Hink, MA. Buckner, An evaluation of machine learning methods to detect malicious SCADA communications, *2013 12th International Conference on Machine Learning and Applications. IEEE*,2: 54-59, 2013.

[8] D. Moon, H. Im, I. Kim, et al, DTB-IDS: an intrusion detection system based on decision tree using behavior analysis for preventing APT attacks, *The Journal of supercomputing*, 73(7): 2881-2895, 2017.

[9] YB. Bing, HZ. Wang, YB. Yong, Intrusion detection of industrial control system based on long-short-term memory network, *Information and control*, 47(1): 54-59, 2018.

[10] K. Wu, Z. Chen, W. Li, A Novel Intrusion Detection Model for a Massive Network Using Convolutional Neural Networks,*IEEE Access*, 6: 50850-50859, 2018.

[11] M. Kravchik, A. Shabtai, Detecting cyber attacks in industrial control systems using convolutional neural networks, in *Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and PrivaCy. ACM*, 2018: 72-83.

[12] A. Tharwat, Classification assessment methods. *Applied Computing and Informatics*, 2018.

[13] P. Nader, P. Honeine, P. Beauseroy, l_p-norms in one-class classification for intrusion detection in SCADA systems, *IEEE Transactions on Industrial Informatics*, 10(4): 2308-2317, 2014.

[14] J. Luo, et al, A novel intrusion detection method based on threshold modification using receiver operating characteristic curve, *Concurrency and Computation: Practice and Experience*, n/a(n/a): p. e5690, 2019.