Journal of Scientific & Industrial Research
Vol. 80, September 2021, pp. 809-816

A Hybrid Classification Approach for Intrusion Detection in IoT Network

Sarika Choudhary* and Nishtha Kesswani

Department of Computer Science, Central University of Rajasthan, Ajmer, Rajasthan, India 305 817

Received 12 December 2020; revised 09 August 2021; accepted 01 September 2021

With the increase in number of IoT devices, the capabilities to provide reliable security and detect the malicious activities within the IoT network have become quite challenging. We propose a hybrid classification approach to detect multi-class attacks in the IoT network. In the proposed model, Principle Component Analysis (PCA) is used to extract the useful features and Linear Discriminant Analysis (LDA) is used to reduce the high dimension data set into lower dimension space by keeping less number of important features. This was assisted by use of a combination of neural network and Support Vector Machine (SVM) classifiers to improve the detection rate and decrease the false alarm rate. The neural network, a multi-class classifier, is used to classify the intruders in the network with more accuracy. The SVM is an efficient and fast learner classifier which is used to classify the unmatched behavior. The proposed method needs less computation complexity for intrusion detection. The performance of the proposed model was evaluated on two benchmark datasets for intrusion detection, i.e., NSL-KDD and UNSW-NB15. Results show that our model outperforms existing models.

Keywords: Internet of Things (IoT), Intrusion Detection Systems (IDS), Linear Discriminant Analysis (LDA), PCA, SVM

Introduction

A collection of heterogeneous devices that exchange information with each other over the internet is called Internet of Things (IoT). The IoT devices are resource constraint, i.e., they have less computation power and less storage space. The devices could be wearable, vehicles, cellphones, smart appliances, smart infrastructure, industry robots etc. People frequently use IoT devices, remotely monitor them, and carry sensitive information like personal data, health-related data, etc. The result is an increased number of attack surface area and possibilities. As we are using increasing number of smart devices, it is imperative to develop smart Intrusion Detection System (IDS) that is efficient in detecting known and unknown attacks.[1]

As the IoT devices are part of smart infrastructure, they are vulnerable to cyber-attacks. According to a report more than 25 billion smart devices will be in operation by the year 2020, which is continuously increasing with time.[2] An FBI report warn about the compromized IoT devices those have been used as proxies. By sensing these compromized devices risk, FBI published the report in 2018. The IoT devices are also used as mediators for computer network manipulation and internet requests used to find out the malicious traffic.[3] It means that attackers are aiming to use IoT devices to perform cyber-attacks and exploits the connected infrastructure.

Most IoT technologies were not designed to keep security in mind and that is the reason behind the wider adoption of the IoT services till now. Traditional internet system uses authentication, cryptography, hash function, etc. as security mechanisms but IoT is a collection of smart devices, so, the security mechanism should also be smart. The IDS is one of the security processes to detect the malicious activities in the IoT network. The IDS should be placed on the networklayer of IoT architecture. The network layer is the backbone of IoT network for connecting heterogenous devices. It also provides chances to implement Network Intrusion Detection Systems (NIDS), i.e., a network-based intrusion detection mechanism which analyze the flow of the network against malicious activities.[4]

The three categories signature-based, anomaly-based, and specification-based are the main categories of IDS[1]. The signature-based IDS is able to detect pre-defined or known attacks in the network. The anomaly-based IDS is able to detect unknown attacks with or without the repository of known attacks. The specification-based IDS is the hybrid version of signature-based and anomaly-based IDS but user can specify any term and condition manually. Hence, with the heterogenous

*Author for Correspondence
E-mail: scpreety98@gmail.com

nature of IoT networks, it is good to choose the hybrid version, i.e., specification-based IDS.

Most of the existing IDS methods calculate the complete accuracy than the detection rate of all the attacks presents in the datasets. For example, NSL-KDD dataset, four attack categories are there such as Denial of Service (DoS), Probe, User to Root (U2R), and Remote to Local (R2L). The last two attack categories have very less number of attack instances compared to first two categories. With fewer instances, these attacks may be dangerous instead of widely known attacks. Although all kind of attacks are important from security perspective, but many methods suffers from addressing rare and dangerous attacks. Hence, it is desired to effectively detect all attack categories, i.e., multi-class attacks. Kim *et al.*[5] proposed a method based on Hierarchical Feature Reduction-Multinomial Logistic Regression (HFR-MLR). It has better accuracy and detection rate (DR) results against most known attacks but the DR results are not promising against the rare kind of attacks.

Contribution

In this paper, we proposed a less computation and less storage based hybrid classification algorithm for the IoT network where multi-class attacks are detected. Main idea is to detect rare kind of attacks effectively. We used neural network and Support Vector Machine (SVM) to classify the different attacks. We used dimension reduction modules like Principal Component Analysis (PCA) for extraction of the useful features and Linear Discriminant Analysis (LDA) for reducing dimension of the dataset. These help to reduce the computation complexity of the proposed scheme. Performance of the proposed work was evaluated by using NSL-KDD[6], UNSW-NB15[(7)] datasets.

Related Work

Many researchers worked on anomaly based intrusion detection using Machine Learning (ML) and Artificial Intelligence (AI) in IoT.[8] Several approaches such as SVM[9], random forest (RF)[10], decision tree[11,12], naive bayes[11,13,14], auto-encoders[15], deep learning[16], artificial neural network[11], K-nearest neighbors[9, 12–14] and many more[17–19] have been used to detect the vulnerabilities in the network.

Gümüşbaş *et al.*[20] surveyed on ML methods for cyber security and datasets for IDS. They provided a detail description of deep learning techniques which included Deep Belief Networks (DBN), autoencoders, Convolutional Neural Networks (CNN), Long-Short Term Memory (LSTM) networks, and Generative Adversarial (GAN) networks and suitable datasets, i.e., AWID2018, CICIDS2017, KDD99, NSL-KDD, Kyoto, and UNSW-NB15 for cyber security.

Another study[21] presented hierarchical clustering and SVM for IDS. They used clustering algorithm for feature selection and SVM for classification on KDD-Cup'99 dataset. Their results are good for DoS and probe attacks but not good for U2R and R2L attacks. These two kinds of attacks are rare attacks and have importance in real-time networks. For performance evaluation, NSL-KDD and KDD-Cup'99 datasets were used. Their work did not show the promising performance on the U2R attack and got 100% false alarm rate. The detection of anomalous behavior in the network has been detected by using Artificial Neural Network (ANN). The authors used NSL-KDD dataset and did the binary-class and multi-class attack classification.

Khan *et al.*[15] presented a two-stage deep learning model for network intrusion detection. Their model use stacked autoencoder with soft-max classifier for efficient classification. The model worked in two stages: first stage worked as an initial stage for detecting attacks and then in the second stage, final decision is to be made. Both stages use soft-max classifier. The model works well for KDD-Cup'99 dataset but is not efficient for UNSW-NB15 dataset while detecting multi-class attack classification. Zhang *et al.*[10] proposed a RF based network IDS and used KDD-Cup'99 dataset for evaluation. Choudhary *et al.*[22] proposed a cluster-based IDS for IoT. Their hybrid IDS approach was designed to detect the selective forwarding and sinkhole attack in the IoT network. The disadvantage is that their work is limited to detect only two kinds of attacks and they didn't use any real-time dataset for evaluation.

Pajouh *et al.*[14] proposed a two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks. They worked with the naive bayes and k-nearest neighbor techniques to identify the intrusions. For performance evaluation, NSL-KDD dataset has been used and the result shows good detection rate for binary and multiclass attack classification. Toosi *et al.*[24] integrated a fuzzy inference method, neuro-fuzzy network, and genetic algorithms (GA) to target an IDS. Their work is getting good DR on

major attacks such as DoS, Probe but still experiences low DR on rare attacks.

Hajisalem *et al.*[25] used hybridization of two classification approaches such as artificial bee colony (ABC) and artificial fish swarm (AFS). In this work, fuzzy c-means clustering and correlation-based feature selection techniques were applied to remove the unimportant features. If-then rules were generated through the Classification and Regression Trees (CART) technique in order to separate the normal and malicious instances. Their performance was evaluated by using NSLKDD and UNSW-NB15 datasets. Kim *et al.*[5] proposed logistic regression-based anomaly detection system which utilized hierarchical feature reduction to discriminate anomalous behaviors from normal ones. Their model shows the increment in the well-known attacks (i.e., DoS and Probe) as well as rare attacks (i.e., U2R and U2R) but the disadvantage is the high false alarm rate. Moustafa *et al.*[7] proposed a Geometric Area Analysis (GAA) technique based on Trapezoidal Area Estimation (TAE) for each examination calculated from the parameters of the Beta Mixture Model (BMM). Geometric area analysis based mechanism reclined on the anomaly-based intrusion detection technique. PCA used to reduce the high-dimension space to the lower one and then evaluate the results with GAA. Two benchmark datasets have been used for performance analysis. Security issues are hurdle to adopt the smart IoT devices. Some authors worked on to provide authentication and other work on security methods to IoT system. Teixeira *et al.*[26] presented a scheme for foiling attacks by crosschecking the flow of data transmission of each IoT mote. Chen *et al.*[27] highlighted a cybersecurity management approach based on automatic model which used to detect, estimate and response to cyber attacks without (or a little) human involvement.

The disadvantages in the above studies are the less detection rate and high false alarm rate. Authors have not detected the rare kind of attacks with best accuracy as well as well-known attacks. The performance varies with datasets and selection of the best features for training. For example, a classifier acts different for different datasets based on the training, and another classifier acts differently. Sometimes, a classifier does not able to detect correct behavior effectively and it leads to high false alarm rate. So, we are trying to fill these gaps by this approach.

Proposed Model

To conquer the shortcomings of past works, i.e., low Detection Rate (DR) of rare attacks, high False Alarm Rate (FAR), and low overall accuracy, we proposed a hybrid classification model as shown in Fig. 1. The proposed model consists of three step process such as dimension reduction, classification using neural network, and classification of the outcomes from neural network with SVM. It is a two stage classification model, at the first stage, it uses neural network to classify the attack classes and at the second stage, it uses SVM for better classification of attack/normal classes.

Dimension Reduction

The use of connected devices accumulates a large amount of data. As data is increasing, visualizing and showing inferences become more difficult and challenging. One of the most used method to visualize data through graphs or charts however, it is not an effective approach. We should use some dimensionality reduction to reduce storage space, computation time, and to observe patterns clearly. The main goal of using dimension reduction techniques on dataset is to remove the redundant or dependent features from higher dimension and bring the dataset to lower dimension space. We deployed PCA and LDA to overcome the high dimensionality issue.

Principle Component Analysis (PCA)

Principle component analysis is an unsupervised dimension reduction technique used in ML. It is used to reduce high dimension to lower one. High

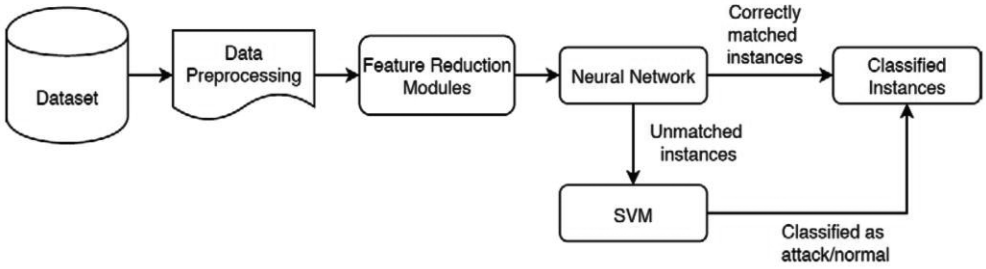

Fig. 1 — Proposed Model

dimensionality means that the dataset has a large number of features. The main problem with high dimension data is model over fitting or inefficient classification with higher computational cost and higher storage space. It is usually used to perform feature selection or feature extraction, i.e., choose and extract more efficient features while preserving the information as much as possible.[28] However, in the proposed work, PCA is used as a feature extraction technique to map the UNSWNB15 and NSL-KDD dataset which consists 47 and 41 main features respectively. The PCA uses linear transformation. The proposed method has three steps for transformation operation as follows:

First, normalization is done which means normalize the range of the continual initial variables so that each one of them contributes equally to the analysis. Let x_i is the random variable in the d-dimensional original dataset, i = 1, 2, ..., n, and n is the total number of values in a variable. Dimension of original data set would be 25192 × 42 for NSL-KDD dataset. Thus the normalization can be done as:

$$z_i = \frac{x_i - \mu}{\sigma} \quad \ldots (1)$$

where,

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2} \quad \ldots (2)$$

Here, σ is the standard deviation, μ is the mean value.

In second step, the covariance matrix is computed to understand how the variables of dataset are varying from the mean value to see the relationship between the variables. Let z_i be the variable from the normalized dataset and i = 1, 2, 3, ..., n. To identify the correlations, compute the covariance between the variables as:

$$C = \sum_{i=1}^{n} (z_i - \mu)(z_i - \mu)^T \quad \ldots (3)$$

where mean μ is defined as:

$$\mu = \frac{1}{n} \sum_{i=1}^{n} z_i \quad \ldots (4)$$

In the covariance matrix, if we have positive values, it means two variables are correlated, and if values are negative, it means two variables are inversely correlated.

In third step, eigenvectors and eigenvalues are computed from the covariance matrix to identify the principle components by using the following expression.

$$Cv = \lambda v \quad \ldots (5)$$

Here, C is the covarience matrix, v is the eigenvector, and λ is the eigenvalue.

Principle Components (PC) mean the new variable set obtained in such a way that variables are uncorrelated and most informative. It is used to leave the redundant features or variables. After getting the eigenvectors and eigenvalues, take an average of them and check whether the value is positive or negative then keep the values according to that. Therefore, if the average value is positive then take positive eigenvectors otherwise choose negative values. After getting filtered eigenvector and eigenvalues, arrange them in descending order then choose whether to keep the features (high significant eigenvalues) or discard (the low significant eigenvalues). After feature selection, the matrix is called feature vector. It is basically a matrix that has selected components. For example, in NSL-KDD, there are 41 features and after applying PCA, we got 22 efficient and informative features, then, feature vector matrix would have 22 columns of components. In UNSW-NB15 dataset, there are 47 features and after applying PCA, we got 28 selective features.

Linear Discriminant Analysis (LDA)

Linear discriminant analysis is a dimension reduction technique used as a pre-processing or for pattern classification approach. It is supervised classification technique that takes data labels as input. Main goal of LDA is to project the features from higher dimension space onto a lower dimension space while preserving the important information.

There are two scatter matrix that need to be gained, first is between-class scatter matrix S_b and second is within class scatter matrix S_w. Let us consider that we have n d-dimensional dataset samples $z_1, z_2, \ldots, z_n$ and they are dived into c different classes. Each class A_i, i = *1, 2, ..., c* has n_i cases such as in our proposed work, c = 5 for NSL-KDD and c = 10 for UNSW-NB15 dataset. Projection matrix P is assessed to minimize the within-class scatter matrix (Eq. 6) and maximize the between-class scatter matrix (Eq. 7). The scatter matrix S_w and S_b are defined as:

$$S_w = \sum_{t=1}^{c} \sum_{j=1}^{n} (z_j - \mu_t)(z_j - \mu_t)^T \quad \ldots (6)$$

$$S_b = \sum_{t=1}^{c} (\mu_t - \bar{z})(\mu_t - \bar{z})^T \quad \ldots (7)$$

where, $\bar{z}$ is the mean of the dataset denoted by:

$$\bar{z} = \frac{1}{n} \sum_{i=1}^{n} z_i \quad \ldots (8)$$

and, μ_t is the simple mean for class A_i

$$\mu_t = \frac{1}{n_c}\sum_{i=1}^{n} z_i \quad , z_i \epsilon A_i \qquad \ldots (9)$$

Now, construct the lower dimension space which maximize the S_b and minimize the S_w. Let P be the lower dimensional space projection which is called Fisher's criterion. Let R be the ratio of these two projections:

$$R = \frac{P^T S_b P}{P^T S_w P} \qquad \ldots (10)$$

All these operations performed on the input dataset to get the reduced transformation matrix. After applying two dimensional reduction techniques, we get modified dataset M which have lower dimension than original dataset N where $M<N$.

Proposed Algorithm

Now, model is trained by modified dataset and classification can be done in the next step to identify the anomalies. For this, multi-class and binary-class classifiers can be used. The choice of selecting classifiers is based on the efficiency to detect different classes, good similarity count of less instances classes, and speed to detect the correct intruded class. The proposed classification modules and how they have applied in the model is shown in Fig.1. Neural network is the multi-class classifier that is used to classify the intruded behavior and unmatched behavior is classified by using SVM. Support vector machine is an efficient classifier since it is a fast learner and classified results are more efficient and accurate. A detail of proposed model is shown in the Algorithm 1.

Algorithm 1: Hybrid Classification Model for Intrusion Detection.

1 Upload Dataset.
2 Data preprocessing.
3 *Data* = Extract feature vector for each class and org_labels.
4 P_c = Apply PCA on *Data* for principle components to select high correlation values.
5 Classify P_c using LDA.
6 for result in LDA_{class}do
7 ifLDA_{class}= = org_labelsthen
8 *Data(Result_counter)*→ append $Data_{LDA}$ and $label_{LDA}$
9 else
10 dump *Data(Result_counter)*
11 Train_neural($Data_{LDA}$, $label_{LDA}$, $neuron_{count}$)
12 *classify_trained*→*classified_neural*
13 Th_1 = *labelLDA*+$\frac{label_{LDA} * vr}{100}$
14 Th_2= *labelLDA*-$\frac{label_{LDA} * vr}{100}$where vr = [20%-80%]
15 If Th_2<*classified_neural*<Th_1then
16 do nothing
17 else
18 train and classify using SVM.
19 Replace result labels with SVM labels.
20 Plot confusion matrix to calculate performance parameters.

We used NSL-KDD and UNSW-NB15 datasets for performance evaluation. At first, we upload the datasets and extract all the features of each class with original labels. Preprocess the dataset means remove the features which contain alphabetic data. Then we apply PCA on the dataset for feature extraction to pick features by removing less significant features. After the processes described in (Eq. 1) to (Eq. 5), it gives output in the form of eigenvectors and eigenvalues. After having PCA matrix, we apply LDA to classify the PCA results as multi-class classifier. It gives output as LDA labels. It examines the class labels with original labels of dataset to reduce the size of dataset. If original class labels match with the LDA class labels then we save those data in the training set otherwise dump the data.

Pass this training data to neural network with data and labels of LDA. Here, we are taking threshold values Th_1 and Th_2 to best match the classified label results from the neural network. We are taking a range of it. For example, if the class label value is 4 and *vr* (some constant value) is 20% then Th_1 would be 4 + 0.2 = 4.2 and Th_2 would be 4 − 0.2 = 3.8. Thus, range of threshold would be [3.8–4.2] for label value 4. We are using *vr* = 0.20 for NSL-KDD dataset and 0.10 for UNSW-NB15 dataset for our work.

So, for the best fit, if classified labels from neural are less than Th_1 and greater than Th_2 then do nothing means we got true value otherwise we find out the unmatched labels and their class and pass them to SVM. Support vector machine classify the unmatched labels of classes thoroughly. Then, replace the unmatched labels from the neural with the SVM output labels for that class. Finally, we calculate performance parameters on the basis of SVM and neural classification results.

Simulation Results

In this section, the comprehensive examination of the applied datasets is discussed, and then model

performance metrics is explained, and finally assessment of the proposed model is stated. Simulation is performed by using MATLAB R2016 brunning on the macOS Catalina powered by 1.8 GHz Dual-Core Intel Core i5 and 8 GB 1600 MHz DDR3 RAM memory.

NSL-KDD Dataset

The extended and refined version of KDD Cup'99 dataset which consist selected records is NSL-KDD.[29] It is a standard dataset for IDS. A total of 42 features are there in NSL-KDD from which 41 are main different features including label such as duration, service, protocol type, flag, etc. and one for label. There are four attack categories, i.e., DoS, Probe, R2L, U2R in NSL-KDD and one normal condition. Class distribution of NSL-KDD dataset is shown in Table 1.

Although, NSL-KDD dataset is refined from KDD-Cup'99 but have some redundancy and due to this classification problem occurs.

UNSW-NB15 Dataset

A new dataset for intrusion detection system is UNSW-NB15 and was published in 2015.[41] It has 49 total features from which 47 main features, one for labels, and one for attack category. Dataset has 5,40,044 total records in which normal records are 2,21,876 and 3,21,283 are the attack records. It has nine kinds of attack categories and one normal. Attack categories fall into Analysis, Backdoor, Fuzzers, DoS Exploits, Reconnaissance, Generic, Shellcode, and Worms.

Performance Metrics

For performance evaluation of an IDS model, we have calculated Detection Rate, False Alarm Rate, and Accuracy.

Detection Rate (D_r) is the degree of classifier that it correctly identified the malicious instances of all anomalous instances and is computed as:

$$D_r = \frac{TP}{TP+FN} \quad \ldots (11)$$

False Alarm Rate (F_r) is the amount of classifier that it wrongly detected the genuine (or normal) instances as malicious of all genuine instances and computed as:

$$F_r = \frac{FP}{FP+TN} \quad \ldots (12)$$

Table 1 — NSL-KDD dataset classes distribution

Datasets	Total Records	Normal	DoS	Probe	R2L	U2R
Train_20%	25,192	13,449	9,234	2,289	209	11
Train$^+$	1,25,973	67,343	45,927	11,656	995	52
Test$^+$	22,544	9,711	7,458	2,421	2,887	67

Accuracy (A_c) is the measure of classifier that it correctly identified the genuine/malicious instances as genuine/malicious out of all instances and computed as:

$$A_c = \frac{TP+TN}{TP+TN+FP+FN} \quad \ldots (13)$$

Evaluation

The proposed model is evaluated using two datasets named NSL-KDD and UNSW-NB15. For NSL-KDD, Train_20% dataset is used which has 25192 records. For UNSW-NB15, a part of dataset is used. Our model has improved performance in detecting less featured attacks such as R2L, U2R (in NSL-KDD dataset), Shellcode, and Worms (in UNSW-NB15 dataset). Comparative analysis on NSL-KDD dataset is shown in Table 2.

We have calculated DR and FAR for each class of NSLKDD and UNSW-NB15 dataset. Comparative summary of overall DR and FAR on NSL-KDD is shown in Table 3.

We have used a part of UNSW-NB15 dataset for performance evaluation and calculated the detection rate and false alarm rate for multi-class attacks. Comparative analysis of detection rate on UNSW-NB15 dataset is shown in Table 4. It is shown that DR

Table 2 — Comparative analysis of multi-class classification Detection Rate (%) on NSL-KDD dataset

Methods	Normal	DoS	Probe	R2L	U2R
Proposed	95.01	89.94	91.98	81.54	85.71
TDTC[14]	94.43	88.20	87.32	42	70.15
Two-tier[13]	94.56	84.68	79.76	34.81	67.16
HFR-MLR[5]	93.70	89.70	80.2	34.50	29.50
ESC-IDS[24]	98.2	84.1	99.5	14.1	31.5

Table 3 — A comparative summary of DR and FAR (%) on NSL-KDD dataset

Methods	Dataset	Detection Rate (%)	FAR (%)
Proposed	Train_20%	92.85	2.99
Two-tier[13]	Train_20%	83.24	4.83
TDTC[14]	Train_20%	84.82	5.56
ANN[30]	Train_20%	81.20	3.23

Table 4 — DR (%) and FAR (%) on UNSW-NB15 dataset

Class Category	Detection Rate	False Alarm Rate (FAR)
Normal	100	12.70
Analysis	83.78	1.03
Backdoor	100	0.94
DoS	100	7.45
Exploits	99.16	32.32
Fuzzers	93.24	13.93
Generic	100	30.13
Reconnaissance	100	8.51
Shellcode	100	0.95
Worms	100	0.15

Table 5 — Comparative summary of DR (%) on UNSW-NB15 dataset

Methods	Normal	Analysis	Backdoor	DoS	Exploits	Fuzzers	Generic	Reconn	Shellcode	worms
Proposed	100	83.78	100	100	99.16	93.24	100	100	100	100
GAA[7]	93.0	76.4	64.8	84.3	65.4	58.6	90.3	45.6	73.8	56.2
ABC-AFS[25]	92.8	80.11	63.4	83.3	63.7	60.3	87.3	49.3	70.9	55.3
TSDL[15]	100	61.35	0	27.06	60.12	87.42	99.87	75.57	65.74	0

varies from 83% to 99% for different categories and 100% for normal instances. Comparative results with existing techniques that worked on UNSW-NB15 dataset for multi-class attack detection is shown in Table 5. It is to be noted that the proposed work shows the promising results with comparison to GAA[7] ABC-AFS[25], and TSDL[15] techniques. The average detection rate of the proposed model is 81.02% and false alarm rate is 2.22% which is quite good as compared to existing methods.

We gained DR and FAR 92.85% and 2.99% respectively for NSL-KDD Train 20% dataset. The comparison in Table 5 shows the multi-class DR results and we can see that our proposed model give good results than other existing methods. Detection rate and false alarm rate is also better for our proposed model than other models. It is worth noting that the model is getting good results and tried to remove the disadvantages of previous works, i.e., inefficiency in detecting the rare and lower instances attacks. The proposed work compared with the multi-class classification works offered the solution for the same classification problem. The studies represented their method to be efficient in some cases but for rare attacks, their results were not promising.

Conclusions

In this study, we proposed a hybrid intrusion detection model for detecting widely known and rare kind (low frequency) of attacks in IoT networks. Our model uses both supervised (i.e., LDA) and unsupervised (i.e., PCA) feature reduction and extraction methods that are able to categories the multi-class attacks and normal behavior. Then we applied combined classification algorithm, i.e., neural network and SVM for better the detection and false alarm rate. Results show our model's better performance on NSL-KDD and UNSW-NB15 datasets in comparison to existing methods.

References

1 Choudhary S & Kesswani N, Cluster-based intrusion detection method for internet of things, In *2019 IEEE/ACS 16th Int Conf Comput Syst App (AICCSA)*, IEEE, (2019, November) 1–8.

2 Kohler A, *In 2020, IoT Security Must Be Part of Your Threat Management Strategy, 2020* (accessed June 19, 2020). [Online], Available: https://securityintelligence.com/posts/in-2020-iot-security-must-be-part-of-your-threat-management-strategy/

3 FBI, *Cyber Actors use Internet of Things Devices as Proxies for Anonymity and Pursuit of Malicious Cyber Activities*, 2018 (accessed June 19, 2020), [Online]. Available: https://www.ic3.gov/media/2018/180802.aspx

4 Bhuyan M H, Bhattacharyya D K & Kalita J K, Network anomaly detection: methods, systems and tools, *IEEE Commun Surv Tutor*, **16(1)** (2013) 303–336.

5 Kim E & Kim S, A novel anomaly detection system based on HFR-MLR method, in *Mobile, Ubiquitous, and Intelligent Computing*, (2014) 279–286.

6 Tavallaee M, Bagheri E, Lu W & Ghorbani A A, A detailed analysis of the KDD CUP 99 data set, in *2009 IEEE Sympos Comput Intel Secur Defen Appl*, (2009) 1–6.

7 Moustafa N, Slay J & Creech G, Novel geometric area analysis technique for anomaly detection using trapezoidal area estimation on large-scale networks, *IEEE Trans Big Data*, **5(4)** (2017) 481–494.

8 Benkhelifa E, Welsh T & Hamouda W, A critical review of practices and challenges in intrusion detection systems for IoT: Toward universal and resilient systems, *IEEE Commun Surv Tutor*, **20(4)** (2018) 3496–3509.

9 Aburomman A A & Reaz M B I, A novel SVM-kNN-PSO ensemble method for intrusion detection system, *Appl Soft Comput*, **38** (2016) 360–372.

10 Zhang J, Zulkernine M & Haque A, Random-forests-based network intrusion detection systems, *IEEE Trans Syst Man Cybern, Part C (Appl Rev)*, **38(5)** (2008) 649–659.

11 Moustafa N, Turnbull B & Choo K K R, An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things, *IEEE Internet Things J*, **6(3)** (2018) 4815–4830.

12 Azmoodeh A, Dehghantanha A, Conti M & Choo K K R, Detecting crypto-ransomware in IoT networks based on energy consumption footprint, *J Ambient Intell Humaniz Comput*, **9(4)** (2018) 1141–1152.

13 Pajouh H H, Dastghaibyfard G & Hashemi S, Two-tier network anomaly detection model: a machine learning approach, *J of Intelligent Info Syst*, **48(1)** (2017) 61–74.

14 Pajouh H H, Javidan R, Khayami R, Ali D & Choo K K R, A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks, *IEEE Trans Emerg Topics Comput*, 7(2) (2019) 314–323.

15 Khan F A, Gumaei A, Derhab A & Hussain A, A novel two-stage deep learning model for efficient network intrusion detection, *IEEE Access*, **7** (2019) 30373–30385.

16 Diro A A & Chilamkurti N, Distributed attack detection scheme using deep learning approach for Internet of Things, *Future Gen Comp Syst*, **82** (2018) 761–768.
17 Sheikhan M & Bostani H, A hybrid intrusion detection architecture for internet of things, in *8th Int Sympos Telecommun (IST)*, (2016) 601–606.
18 Bostani H & Sheikhan M, Hybrid of anomaly-based and specification-based IDS for Internet of Things using unsupervised OPF based on MapReduce approach, *Comput Commun*, **98** (2017) 52–71.
19 Raza S, Wallgren L & Voigt T, SVELTE: Real-time intrusion detection in the Internet of Things, *Ad hoc netw*, **11(8)** (2013) 2661–2674.
20 Gümüşbaş D, Yıldırım T, Genovese A & Scotti F, A comprehensive survey of databases and deep learning methods for cybersecurity and intrusion detection systems, *IEEE Syst J*, **15(2)** (2021) 1717–1731.
21 Horng S J, Su M Y, Chen Y H, Kao T W, Chen R J, Lai J L & Perkasa C D, A novel intrusion detection system based on hierarchical clustering and support vector machines, *Expert Syst Appl*, **38(1)** (2011) 306–313.
22 Choudhary S & Kesswani N, Cluster-based intrusion detection method for internet of things, in *IEEE/ACS 16th Int Conf Comput Syst Appl (AICCSA)* IEEE, November 2019, 1–8.
23 Panda M, Abraham A & Patra M R, Discriminative multinomial naive bayes for network intrusion detection in 6th *Int Conf Inform Assur Secur*, *IEEE*, August 2010, 5–10.
24 Toosi A N & Kahani M, A new approach to intrusion detection based on an evolutionary soft computing model using neuro-fuzzy classifiers, *Comput commun*, **30(10)** (2007) 2201–2212.
25 Hajisalem V & Babaie S, A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection, *Comput Netw*, **136** (2018) 37–50.
26 Teixeira F A, Vieira G M, Fonseca P M, Pereira F M Q, Wong H C, Nogueira J M S & Oliveira L B, Defending Internet of Things against exploits, *IEEE Lat Am Trans*, **13(4)** (2015) 1112–1119.
27 Chen Q, Abdelwahed S & Erradi A, A model-based validated autonomic approach to self-protect computing systems, *IEEE Internet things J*, **1(5)** (2014) 446–460
28 Abdi H & Williams L J, Principal component analysis, *Wiley Interdisciplinary Reviews: Computational Statistics*, **2(4)** (2010) 433–459.
29 Choudhary S & Kesswani N, Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 Datasets using Deep Learning in IoT, *Procedia Comput Sci*, **167** (2020) 1561–1573.
30 Ingre B & Yadav A, Performance analysis of NSL-KDD dataset using ANN, in *Int Conf Signal Process Commun Eng Syst*, (2015) 92–96.