Automated Software Engineering (2021) 28:19
https://doi.org/10.1007/s10515-021-00298-7

Deep learning approach for intrusion detection in IoT-multi cloud environment

D. Selvapandian[1] · R. Santhosh[1]

Received: 4 May 2021 / Accepted: 13 September 2021 / Published online: 21 September 2021

Abstract
The possibility of connecting billions of smart end devices in the Internet of Things (IoT) provides wide range of services to the user. But, the unlimited connectivity of devices in IoT brings security issues when it is connected to wireless networks. Integrating cloud with IoT networks gains more attention as it reduces the sensor node resource limitations. However, the network complexity, open broadcast characteristics of IoT networks are vulnerable to attacks. To ensure network security and reliable operations, Intrusion Detection Systems (IDS) are widely preferred. IDS identifies the anomalies effectively in complex network environments and ensures the security of the network. Traditional intrusion detection systems based on neural networks consume long training time and low classification accuracy. Recently, deep learning methods are widely used in various image and signal processing, security applications. This research work presents a deep learning-based intrusion detection system for multi-cloud IoT environment to overcome the limitations of neural network-based intrusion detection models. The proposed intrusion detection model improves the detection accuracy by improving the training efficiency. Experimental evaluation of proposed model using NSL-KDD dataset provides improved performance than conventional techniques attaining 97.51% of detection rate, 96.28% of detection accuracy, and 94.41% of precision.

Keywords Internet of Things (IoT) · Multi-cloud computing · Intrusion detection system (IDS) · Deep learning technique · Detection accuracy

✉ D. Selvapandian
selvapandian79@gmail.com

R. Santhosh
santhoshrd@gmail.com

[1] Department of Computer Science and Engineering, Faculty of Engineering, Karpagam Academy of Higher Education, Coimbatore, India

1 Introduction

The technology development opens up the way for innovation of wide range of applications using smart devices collectively in terms of Internet of Things. Recent studies report that approximately 2.5 quintillion bytes of data are generated by IoT devices every day and it increases every year. IoT becomes a part of the future internet where billions of devices are used to sense, compute, communicate, actuate various physical and virtual attributes from anywhere and it can be accessed by anyone from any place. The self-configured paths, networks, and services are the major benefits of IoT. However, the smart devices in the IoT network have resource constraints. The limited capacity can accommodate a specific memory and that can be used to actuate and process the data. Similarly, open broadcast communication and wireless networks in IoT are vulnerable to user data privacy and network security (Meneghello et al. 2019). The hardware platforms, network interconnection topologies have flaws and vulnerable to attacks. Due to this, the overall security of the IoT is questionable when it is connected to heterogeneous networks. To overcome this cloud computing infrastructure is incorporated with IoT networks.

Cloud computing provides better storage and processing abilities to the IoT networks through its software and hardware resources. The reasonable operating cost attracts IoT users to move their data to the cloud and all the needs of IoT networks can be satisfied through cloud computing (Rafique et al. 2020). Data processing, streaming, and managing are possible in cloud computing and it can support IoT networks through its geographically distributed data sources. All the requirements of IoT applications can effectively be solved by cloud computing. Essentially cloud act as a transport layer between IoT and applications so that the scalability, flexibility is improved and complexities between the IoT and applications get reduced. Integration of IoT and cloud introduces new paradigms as sensing as a service, actuation as a service, surveillance as a service, data analytics as a service, sensor event as a service, etc.,

Integrating cloud and IoT provides various benefits however this integration imposes several challenges in terms of device discovery, device management, quality of service, mobility issues, security, and service level agreement management (Li et al. 2019). The integration of IoT module with cloud services may be a choice of single cloud or multi-cloud depends on user requirements. The difference between single cloud and multi-cloud is in its architecture. Single cloud refers to either a private cloud, public cloud which is accessed by the user. Whereas multi-cloud can incorporate mixed cloud architectures which include multiple public clouds, or private clouds or combination of public and private clouds. When it is used as a service, multiple independent operators need to provide various services across cloud layers and it must be integrated based on the IoT applications as a multi-cloud environment. The detection model is used as the network level intrusion detection and it perform detection before the data transfer into the multi-cloud environment.

A multi-cloud environment consists of several data centers that are distributed across the network. Data centers are distributed either geographically or topologically will facilitate various challenges for IoT applications that used multiple service providers. Moving all those to a single centralized data center will increase the communication overhead of the network and a single data center could not able to fulfill the storage and computation requirements of IoT applications (Porambage et al. 2018). Enabling third party access to the data significantly increases the possibility of threats that might affect the quality of services of multi-cloud IoT applications. So it is essential to include an intrusion detection system for IoT module that accommodates multi-cloud environment and detects threats and attacks.

Generally, intrusion detection system is categorized into two types as signature-based intrusion detection and anomaly-based intrusion detection (Benkhelifa et al. 2018). Other than these two types hybrid detection models are also available that works based on the specific application. In signature-based anomaly detection, pattern matching techniques are used to identify the attacks. By matching the signature of the previous intrusion the present intrusion is verified and alert the system for malicious activities. Whereas anomaly-based intrusion detection systems are introduced that utilizes statistical or knowledge-based techniques and detects malicious activities in the network. The deviation between the present model and observed behavior is used to list the intrusion. The training and testing phase of anomaly-based intrusion detection systems learns the normal traffic profile in the training phase and checks the abnormal behavior in the testing phase.

In this research work, a deep learning-based intrusion detection system for an IoT- multi-cloud environment is presented that integrates the end-user applications and identifies the intrusion level in the network. The major contribution of this research work is given as follows:

- Improving the Intrusion detection performance using Deep learning-based approach for IoT multi-cloud environment
- The proposed model has increased detection accuracy with minimum computation time.
- The proposed deep learning based approach reduces the training time that is the major limitation of conventional intrusion detection models.
- Provided secure data transfer in IoT multi-cloud architecture with proposed intrusion detection model that prevents the security issues among the end-users.

The rest of the paper is organized as follows: Sect. 2 provides a brief literature survey, Sect. 3 presents the proposed multi-cloud IoT intrusion detection model, Sect. 4 presents the experimental results and its discussion and conclusion are given in Sect. 5.

2 Related works

A brief literature analysis in the field of intrusion detection systems is presented in this section. privacy and security are the major considerations in IoT applications as all the collected information is more sensitive. Researchers provided various solutions to ensure data security in IoT applications by identifying anomalies or intrusions in the network. An intrusion detection system is used to identify malicious activities so that the security of the application can be maintained in a better manner. Traditional firewalls could not able to identify the malicious network traffic and unauthorized access and it can be detected through intrusion detection systems. IDS increases the system's integrity, availability, and confidentiality. Two types of intrusion detection systemsare signature-based intrusion detection and anomaly-based intrusion detection (Chaabouni et al. 2019). By matching the signature of the previous intrusion the present intrusion is verified and alert the system for malicious activities in the signature-based anomaly detection. However, signature-based intrusion detection is less effective and could not able to detect attacks due to the absence of the signature. So, anomaly-based intrusion detection models are introduced.

Anomaly-based intrusion detection models are categorized into supervised learning, unsupervised learning, reinforcement learning, and deep learning. In this supervised learning, intrusions are detected using labeled training data. The relevant features and its classes are identified in the training phase and through the learning process, the intrusion or normal behavior is categorized. Decision tree based intrusion detection system (Luo et al. 2020) is a supervised learning model that initially identifies the test attributes from the decision node. from the test attributes the possible decisions are obtained as a branch and the instances are comprised as leaves. The tree structure classifies the normal and abnormal nodes in the network. The probability of attacks and normal behavior is considered in the Naïve Bayes approach-based intrusion detection system (Pajouh et al. 2019) has better calculation efficiency. However, if the independent assumption used in the model is not valid it will affect the detection performance. Moreover, the complex attribute dependencies reduce the overall performance of the detection module.

Genetic algorithm based intrusion detection system provides quality solutions through its evolution principle (Zhang et al. 2019). Through selection and reproduction operators the quality of solutions is improved in the genetic algorithm based detection models. Chromosome encoding types used in the genetic algorithm are based on clustering and cluster centers. However, the computation complexity of genetic based intrusion detection system is high which is considered as a major demerit. In machine learning based intrusion detection systems, the artificial neural network is widely used to detect different malware. The backpropagation algorithm is a supervised learning model that assesses the network error based on weights to detect the malware (Qiu et al. 2020). However, the detection accuracy of ANN-based intrusion detection models (Shenfield et al. 2018) is less and it needs to be improved. compared to more frequent attacks, the training dataset for less frequent attacks is less that makes the network to learn the attacks correctly.

The major limitation of ANN-based intrusion detection systems is time consuming learning process, local minima, and less detection accuracy. Fuzzy logic based intrusion detection systems (Smys et al. 2021) provides anomaly detection based on the degree of uncertainty. Multiple classes of intrusions are detected in the research work. However, the intrusion detection model needs to handle multiple numeric features that introduce high false alarms as it fails to recognize the minor changes in the normal activities. Support vector machine-based intrusion detection model (Teng et al. 2018) uses a kernel function to map the training data into high dimensional space. Linear, Gaussian, Polynomial, and radial basis functions are used as kernels to separate the hyperplanes. In the intrusion detection process, the redundant and less influent features are separated into data points and multiple class classification results are obtained in the support vector-based intrusion detection systems. Multiple machine learning algorithms are widely as ensemble methods (Moustafa et al. 2019) to detect intrusions in the network. Ensemble methods improve the predictive performance and increase the detection rate.

Unsupervised learning in intrusion detection systems utilizes input data without any class labels to extract essential information. The learning process is used to group the data into various classes and intrusions are identified by train the model. Based on the similarity the groups are created and the outliers are considered as anomalies in the unsupervised approach. k-means (Al-Yaseen et al. 2017), PCA (Ali et al. 2018), hierarchical clustering(Yahalom et al. 2019) based intrusion detection systemsare few familiar intrusion detection systems in which k-means have the benefit of identifying different behaviors. Whereas PCA obtains the low dimensional features from a large dataset that reduces the computational complexity of the intrusion detection system. In hierarchical clustering, bottom-up or iterative clustering is used to produce sub-clusters and identifies the abnormalities from large feature datasets.

The principles of deep learning and reinforcement learning are used in the intrusion detection system trains the deep neural network and obtains possible results for the given environment. further, the performance is enhanced through a deep Q-network that combines the deep neural network (Naseer et al. 2018) and reinforcement learning principles for intrusion detection (Smys and Haoxiang 2021). The learning policy of reinforcement learning faces overestimation problems in the detection process and it is eliminated in the double Q-learning model. other than this Recurrent Neural Network (RNN) (Yin et al. 2017) based IDS functions efficiently over series of data. For current state prediction, the information of prior state is used in RNN based IDS that makes the approach appropriate for any network with high detection accuracy compared to conventional machine learning based approaches.

In literature (Sivaganesan 2021), blockchain based attack detection in IoT sensor networks is presented to detect black and grey hole attacks. The presented blockchain based trust model avoids single point failure and reduces communication and computation overhead of the network. However, the system is limited to detect specific black and grey hole attacks which is the demerit of the presented model. From the above, it is observed that traditional machine learning method-based intrusion detection system performance can be improved in terms of accuracy and computation time. Deep learning-based intrusion detection models provide multiclass intrusion detection opportunities compared to conventional machine learning based intrusion detection systems. Considering this as a research gap, this research work presents a deep learning based intrusion detection system for multi-cloud IoT environment to improve the detection accuracy with minimum time consumption.

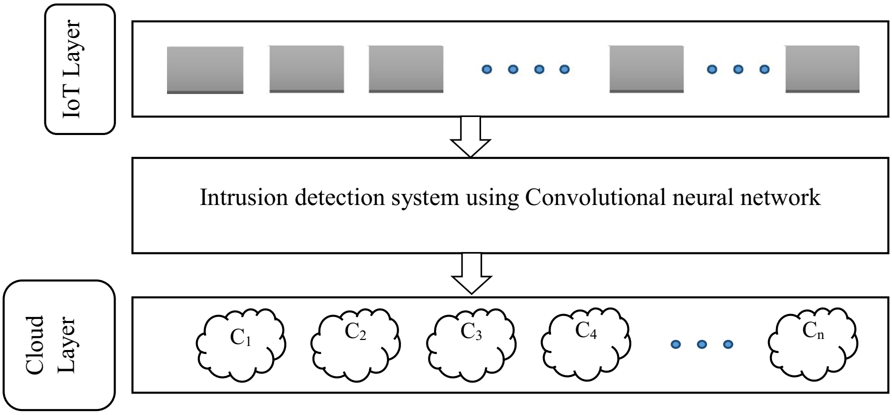

IoT Layer
Intrusion detection system using Convolutional neural network
Cloud Layer
C1
C2
C3
C4
Cn

Fig. 1 Overview of proposed work

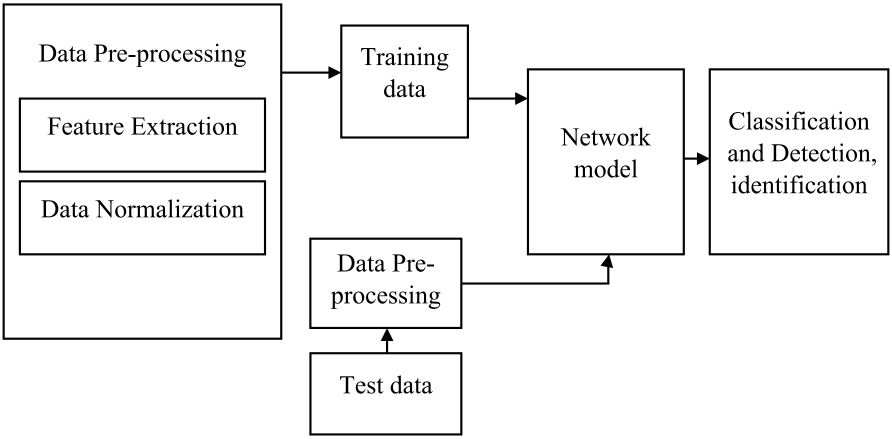


Fig. 2 Proposed Intrusion Detection Model

3 Proposed work

The proposed multi-cloud IoT intrusion detection system is presented in this section. The convolutional neural network is used in the proposed intrusion detection model. The detection model is present between the cloud and IoT gateway so that attacks in the IoT network are identified and secure the cloud network also. Figure 1 depicts the overall process flow of the proposed model.

The IoT layer represents the n number of IoT devices are connected to the intrusion detection system and further it is connected to multiple cloud environments. The research work doesn't discuss the routing strategy so a simple overview is presented in Fig. 1. The process in the intrusion detection system is depicted in Fig. 2. It includes feature selection, feature extraction, data normalization, training, testing, and classification. Initially, in the preprocessing steps, the features are extracted and it is normalized to balance the feature dimensions. It is essential to analyze both low-level and high-level features this normalization is required in the proposed approach.

In the data preprocessing step, feature extraction and normalization are performed. Since most of the learning models can able to handle numerical values for the training and testing process it is necessary to convert all values into numerical form through the normalization process. One hot encoding is one of the familiar models used for the normalization process. In the experimentation part, the NSL-KDD dataset is used and this normalization process identifies the features and maps them into dimensional features once the transformation is completed. For an instance, the nominal features in the dataset such as protocol type, flag, and service features are encoded into its binary values. The attributes on the protocol type are TCP, ICMP, and UDP and it can be encoded into (1,0,0) for TCP, (0,0,1) for ICMP, and (0,1,0) for UDP as feature vectors.

Similarly, the features are encoded into dimensional features are identified into 41 and it is mapped as 122-dimensional features after the transformation process. However, one hot encoding increases the number of features for every transformation that might increase the training and testing time. So in this proposed work, the nominal values are converted by assigning specific variables in alphabetic order. For example, ICMP is assigned with value 1, TCP is assigned with value 2, and UDP is assigned with value 3. This process doesn't increase the number of features and reduces the training and testing time of the detection model. Using min–max transformation the numeric features in the dataset are mapped into binary values in the range 0 to 1 in the normalization process and it is given as

$$i = \frac{n_i - Min}{Max - Min} \tag{1}$$

where the numeric feature is represented as n_i for the ith sample and the maximum, minimum values of numeric features are represented as Max and Min respectively. The deep learning model is constructed by tuning the optimal hyperparameters. Figure 3 depicts the network model used in the intrusion detection system in detail. Various CNN models are already evolved for intrusion detection, however, the research towards improved performance is still in progress. The ability in feature selection and processing, high accuracy and computational efficiency are the major reason for selecting CNN model for this proposed approach. In this research work LeNet based

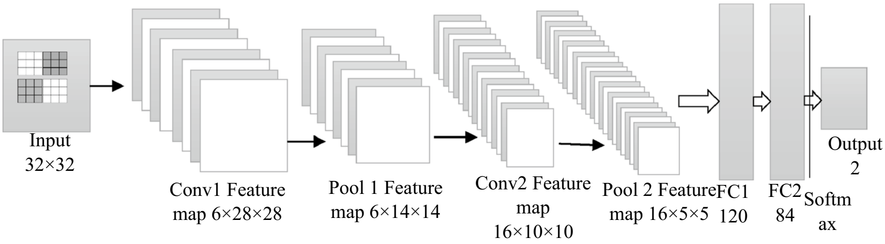


Fig. 3 Proposed LeNetmodel for intrusion detection

intrusion detection system is proposed for IoT multi-cloud environment. Compared to other methods such as GoogleNet and AlexNet, the proposed LeNet based model provides better performance and simple to compute the inputs. The training speed and convergence rate are high compared to other CNN models.

The normalized input is given to the proposed detection model and convolution layer uses a series of convolution kernels and identifies the features of the network packet data. Mathematically it is expressed as

$$n_k^h = \sum_{h \in m_k} n_k^{h-1} * m_{kh}^h + b_k^h \tag{2}$$

where the input feature matrix is represented as n and the convolutional kernel is represented as m and the bias term is represented as b. Pooling layer is used after convolution layer to obtain high feature map. The pooling layer reduces the dimension of original input feature from convolution layer and avoid overfitting issues. The mathematical expression for pooling model is given as

$$S_k^h = \mu_{k=1} max_{h=1}\left(F_{kh}\right) + b_k^h \tag{3}$$

where μ is the pooling factor which is used to optimize the pooling function and it is given as

$$\mu = \varphi \frac{A\left(n_{max} - A\right)}{n_{max}^2} + \theta \tag{4}$$

where the average pooling element is given as A, the maximum pooling element is represented as n_{max} and alignment error term is represented as θ. The feature coefficient φ used in Eq. (4) is obtained as follows

$$\varphi = \frac{l}{1 + \left(n_{epoc} - 1\right)c^{n_{epoc}^2 + 1}} \tag{5}$$

where the length of the pooling layer is represented as l and the number of iterations during training is represented as n_{epoc}. An activation function is generally used between each layer and in the proposed model ReLU activation function is used after the convolution layer. compared to other activation function like sigmoid, tanh (Mugunthan and Vijayakumar 2021) the training speed of ReLU is high and it is expressed as

$$f(s) = \max(0, s) \tag{6}$$

By reducing the feature size, the training speed of the system is increased. Two layers of convolution and pooling layers are used in the proposed work to reduce the feature and finally, a fully connected layer is used to classify the given features.

The two-dimensional features are converted into one-dimensional features in the fully connected layers followed by a SoftMax classifier is used to classify the intrusion in the dataset. SoftMax classifies the n number of neurons into multiple classes and its mathematical formulation is obtained as

$$f_k(z) = \frac{e^{z_k}}{\sum_k e^{z_l}} \tag{7}$$

The flowchart to explain the process of the proposed intrusion detection system is given in Fig. 4. Initially, the process starts from the conversion of numerical data followed by normalization. Then the feature vectors are transformed into two-dimensional features are the system is trained with the given data. once the training process is completed the test data are loaded to the LeNet model and intrusions are detected based on the classifier results. Most of the intrusion detection system in deep learning models used traditional CNN model which requires more training time But, the selection of minimal feature and detect the attack types using LeNet model are considered to be the novelty of the research work. Compared to traditional CNN based intrusion detection model the performance of proposed deep learning has greatly increased and it is discussed in the following section.

4 Result and discussion

The proposed intrusion detection model is verified through simulation and the results are discussed in this section. NSL-KDD dataset is used for the experimentation (Mohammed and Ahmed 2019; Akey and Sharma 2021; Raj 2021). Dataset has 41 attributes of different features and is assigned with one label as either an attack or normal. Based on the attack it is categorized into DoS, Probe, R2L, and U2R. Training and testing records in NSL-KDD datasets are listed in Table 1.

The proposed system has experimented in Tensorflow version is 1.11.0 and the Anaconda version is 4.5.11 in an i5 Intel processor CPU with 2.5Ghz frequency and 8 GB of memory. The operating system is Ubuntu 16.04 (Table 2). The parameter setting for the LeNet model is given in Table 3.

The simulation is performed with different epoch values and at 60 epoch the proposed model attains stable results. As the epoch increases the accuracy increases and loss function decreases. The training and testing accuracy of proposed model is 99% for both with minimum false alarm rate. The relationship between the metrics are observed as a confusion matrix and it is depicted in

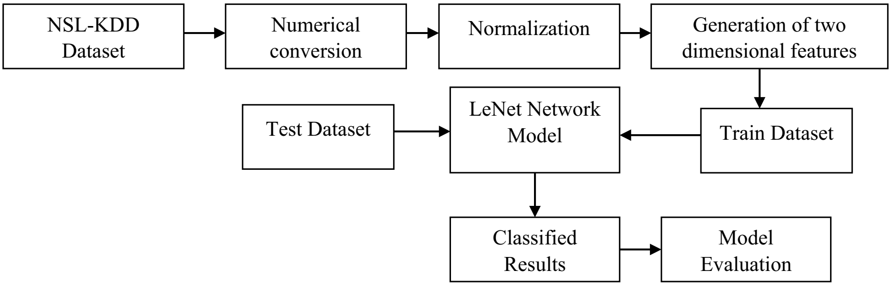


Fig. 4 Flowchart of proposed LeNet model for Intrusion detection

Table 1 Training and testing records in NSL-KDD Dataset

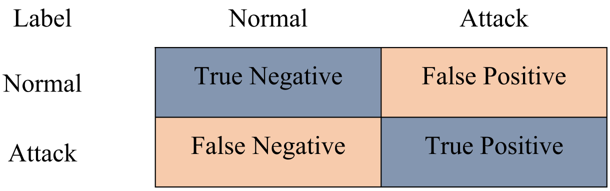
Attack	Train	Test
Normal	13,357	9690
Dos	9234	7435
Probe	2289	2421
R2L	209	2754
U2R	11	200
Total	25,100	22,500

Table 2 Parameter setting of LeNet model

Layer	Attribute	Size	Strides	Activation function
L11	Conv 1	6×28× 28	1	ReLU
L12	Pool 1	6×14× 14	1	ReLU
L21	Conv 2	16×10× 10	1	ReLU
L22	Pool 2	16×5× 5	1	ReLU
L3	FC1	–	1	ReLU
L4	FC2	–	1	Dropout
L5	–	–	–	Softmax

Table 3 Performance comparative analysis

S. no	Performance metrics	SVM	RNN	Proposed
1	Accuracy (%)	88.84	91.17	96.28
2	Precision (%)	76.8	89.4	94.41
3	Detection rate (%)	89.4	93.6	97.51
4	False positive rate (%)	10.2	9.4	6.2



Label	Normal	Attack
Normal	True Negative	False Positive
Attack	False Negative	True Positive

Fig. 5 Metrics relationship- confusion matrix

Fig. 5 for the proposed network model. based on the label the values are categorized into True positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). The predicted and actual values are used to create the confusion matrix.

The parameters used to evaluate the performance of the proposed model are accuracy, detection rate, False Positive Rate (FPR), False Negative Rate (FNR), precision, and training time. The essential formulations for the above-mentioned parameters are given as follows.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$Detection\, rate = \frac{TP}{TP + FN} \tag{9}$$

$$FPR = \frac{FP}{FP + TN} \tag{10}$$

$$FPR = \frac{FN}{FN + TP} \tag{11}$$

$$Precision = \frac{TP}{TP + FP} \tag{12}$$

The detection rate of the proposed intrusion detection system and existing Support Vector Machine (SVM), Recurrent Neural Network (RNN) based intrusion detection systems are compared and depicted in Fig. 6. It is observed that the detection rate of the proposed model is better than other models. the performance of RNN model is 4% lesser than the proposed model whereas the performance of SVM model is 8% lesser than the proposed approach. The performance of proposed model maintains an average of 97.5% for the entire iteration and there is a minor variation in the detection rate when it reaches 1000. Whereas support vector machine exhibits major variations in its detection rate due to its feature handling characteristics in the intrusion detection system.

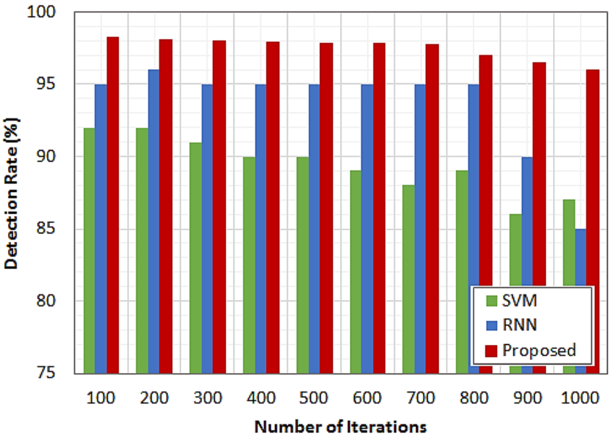


Fig. 6 Detection rate comparison

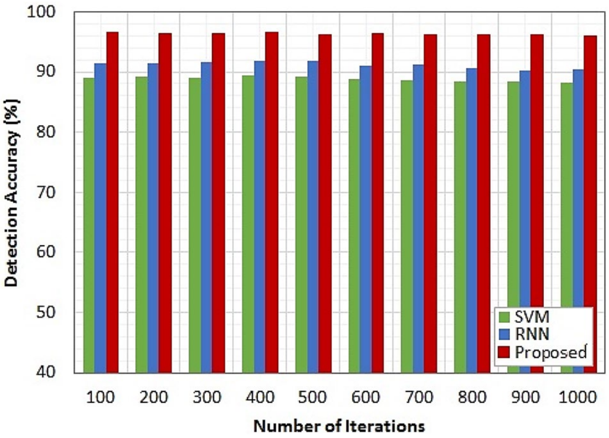


Fig. 7 Detection accuracy comparison

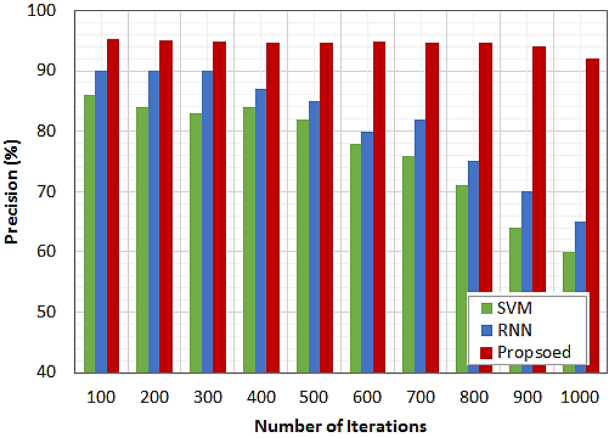


Fig. 8 Precision comparison

Figure 7 depicts the detection accuracy comparison for all three models. The average maximum accuracy attained by the proposed model is 96.28% whereas SVM and RNN attain 88.84% and 91.17% respectively. Detection accuracy given in the figure defines how efficiently the proposed approach detects the attacks. Whereas classification accuracy defines the type of attack as correctness score. The feature transformation and training process improves the detection accuracy of the proposed model. Though the detection rate is high, slightly reduced detection accuracy indicates the classification performance of the proposed intrusion detection model. It classifies the attacks efficiently than other models.

The detection precision of the proposed model is compared with existing models and depicted in Fig. 8. It is observed from the analysis proposed model exhibits maximum precision whereas support vector machine and RNN models attain only half of the precision values for 1000 iterations. Due to better feature extraction and processing through deep neural network functions, maximum precision is obtained by the proposed model. Compared to proposed approach the performance of SVM is poor which attains minimum precision. Whereas the precision values of RNN is quite acceptable for number of iterations but it reduces when the number of iterations is increased.

Figure 9 depicts the comparison of runtime with respect to the number of tasks for all three models. Run time is the time taken by the system to perform an operation. Run time is different from training time, since training time is defined based on how much time the system takes for learning the dataset whereas run time defines how much time the system takes while executing the test inputs. It could be observed from the figure that the proposed model takes minimum run time to handle the tasks. The maximum run time was occupied by the support vector machine and the performance of RNN is better than the SVM model.

The detection rate comparison for the proposed and existing intrusion detection systems are depicted in Fig. 10 based on the number of requests. It is observed that the proposed model has a maximum detection rate even the number of requests is maximum whereas SVM exhibits poor performance. Though the performance of RNN is better but it is lesser than the proposed model.

Figure 11 depicts the training time comparison for the proposed model and existing techniques. One of the objectives of the research work is to reduce the training time which is considered as the major limitation of machine learning based models. Due to minimal number of features the training time of proposed model is minimum compared to other techniques. It could be observed from the figure that the proposed deep learning-based intrusion detection model consumes minimum training time compared to other techniques.

Figure 12 depicts a comparative analysis of proposed detection and existing detection models for different attacks. It could be observed from the results the

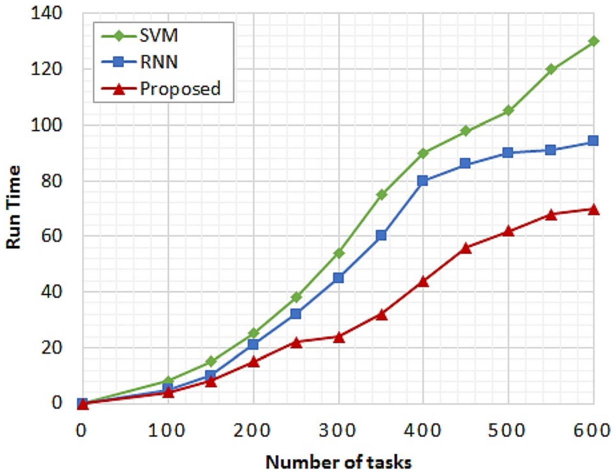


Fig. 9 Runtime comparison

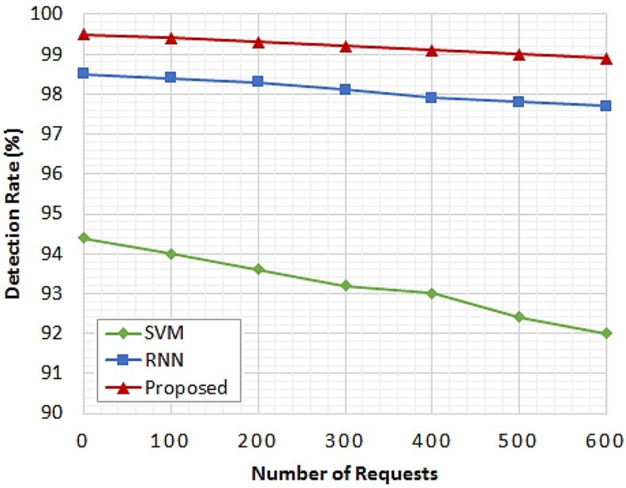


Fig. 10 Detection rate comparison vs number of requests

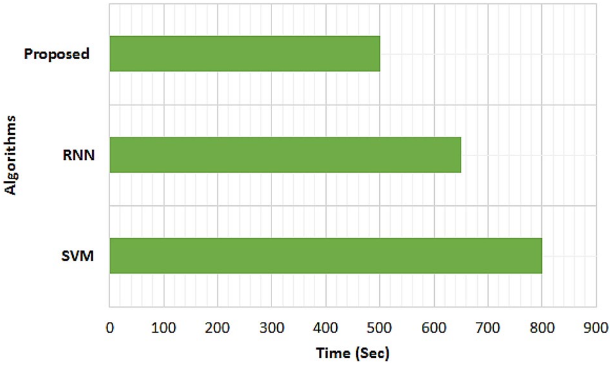


Fig. 11 Training time comparison

proposed approach exhibits a maximum detection rate for all kinds of attacks. The performance of RNN model is satisfactory for DoS and probe however it exhibits poor performance for R2L and U2R attacks.

The performance metrics used to evaluate the proposed model and existing models are summarized numerically in Table 3. It is observed from the comparative analysis that the proposed deep learning model attains better performance in all the aspects like detection accuracy, detection rate, precision, and less false positive rate. This improved performance over RNN indicates that the proposed model is suitable for detecting intrusions in the multi-cloud IoT environment.

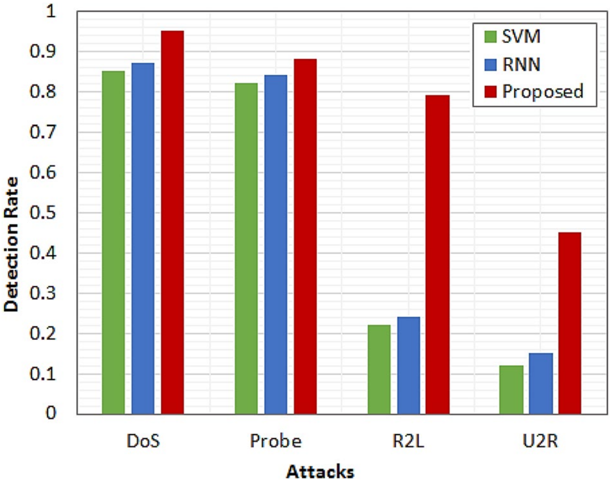


Fig. 12 Comparison of detection rates to different attack

5 Conclusion

This research work presents deep learning-based IoT multi-cloud intrusion detection system to enhance network security. The increased training time and less detection accuracy of the conventional machine learning algorithms are overcome by the proposed intrusion detection model. The proposed deep learning model deeply mint the features in the dataset and improves the classification accuracy. The multiscale features are extracted through convolution operations at different levels and attained better performance in terms of detection accuracy, detection rate, precision, false positive rate. Experimental verification utilized the NSL-KDD dataset and the observed values are compared with existing support vector machine-based and recurrent neural network-based intrusion detection systems. The proposed model attains better performance in all aspects. Further, this research work can be improved by implementing hybrid models to detect multi-class attacks in the cloud environment.

Author contributions All authors are contributed in this work.

Funding We are used our own data and coding.

Declarations

Conflict of interest No conflict of interest.

Human and animal rights Humans and animals are not involved in this work.

References

Akey, S., Sharma, R.: Design an early detection and classification for diabetic retinopathy by deep feature extraction based convolution neural network. J. Trends Comput. Sci. Smart Technol. **3**(02), 81–94 (2021)

Ali, M.H., Al Mohammed, B.A.D., Ismail, A., Zolkipli, M.F.: A new intrusion detection system based on fast learning network and particle swarm optimization. IEEE Access **6**, 20255–20261 (2018)

Al-Yaseen, W.L., Othman, Z.A., Nazri, M.Z.A.: Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system. Expert Syst. Appl. **67**, 296–303 (2017)

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), 3496–3509 (2018)

Chaabouni, N., Mosbah, M., Zemmari, A., Sauvignac, C., Faruki, P.: Network intrusion detection for IoT security based on learning techniques. IEEE Commun. Surv. Tutor. **21**(3), 2671–2701 (2019)

Li, X., Wang, Q., Lan, X., Chen, X., Zhang, N., Chen, D.: Enhancing cloud-based IoT security through trustworthy cloud service: an integration of security and reputation approach. IEEE Access **7**, 9368–9383 (2019)

Luo, J.-L., Yu, S.-Z., Peng, S.-J.: SDN/NFV-based security service function tree for cloud. IEEE Access **8**, 38538–38545 (2020)

Meneghello, F., Calore, M., Zucchetto, D., Polese, M., Zanella, A.: IoT: internet of threats? A survey of practical security vulnerabilities in real IoT devices. IEEE Internet Things J. **6**(5), 8182–8201 (2019)

Mohammed, M.N., Ahmed, M.M.: Data preparation and reduction technique in intrusion detection systems: ANOVA-PCA. Int. J. Comput. Sci. Secur. **13**(5), 167–182 (2019)

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), 4815–4830 (2019)

Mugunthan, S.R., Vijayakumar, T.: Design of improved version of sigmoidal function with biases for classification task in ELM domain. J. Soft Comput. Paradigm **3**(02), 70–82 (2021)

Naseer, S., Saleem, Y., Khalid, S., Bashir, M.K., Han, J., Iqbal, M.M., Han, K.: Enhanced network anomaly detection based on deep neural networks. IEEE Access **6**, 48231–48246 (2018)

Pajouh, H.H., Javidan, R., Khayami, R., Dehghantanha, A., 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. Top. Comput. **7**(2), 314–323 (2019)

Porambage, P., Okwuibe, J., Liyanage, M., Ylianttila, M., Taleb, T.: Survey on multi-access edge computing for internet of things realization. IEEE Commun. Surv. Tutor. **20**(4), 2961–2991 (2018)

Qiu, X., Dai, J., Hayes, M.: A learning approach for physical layer authentication using adaptive neural network. IEEE Access **8**, 26139–26149 (2020)

Rafique, W., Qi, L., Yaqoob, I., Imran, M., Rasool, R.U., Dou, W.: Complementing IoT services through software defined networking and edge computing: a comprehensive survey. IEEE Commun. Surv. Tutor. **22**(3), 1761–1804 (2020)

Raj, J.S.: Security enhanced blockchain based unmanned aerial vehicle health monitoring system. J. ISMAC **3**(02), 121–131 (2021)

Shenfield, A., Day, D., Ayesh, A.: Intelligent intrusion detection systems using artificial neural networks. ICT Express **4**(2), 95–99 (2018)

Sivaganesan, D.: A data driven trust mechanism based on blockchain in IoT sensor networks for detection and mitigation of attacks. J. Trends Comput. Sci. Smart Technol. **3**(01), 59–69 (2021)

Smys, S., Haoxiang, W.: Security enhancement in smart vehicle using blockchain-based architectural framework. J. Artif. Intell. **3**(02), 90–100 (2021)

Smys, S., Wang, H., Basar, A.: 5G network simulation in smart cities using neural network algorithm. J. Artif. Intell. **3**(01), 43–52 (2021)

Teng, S., Wu, N., Zhu, H., Teng, L., Zhang, W.: SVM-DT-based adaptive and collaborative intrusion detection. IEEE/CAA J. AutomaticaSinica **5**(1), 108–118 (2018)

Yahalom, R., Steren, A., Elovici, Y.: Improving the effectiveness of intrusion detection systems for hierarchical data. Knowl.-Based Syst. **168**, 59–69 (2019)

Yin, C., Zhu, Y., Fei, J., He, X.: A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access **5**, 21954–21961 (2017)

Zhang, Y., Li, P., Wang, X.: Intrusion detection for IoT based on improved genetic algorithm and deep belief network. IEEE Access **7**, 31711–31722 (2019)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.