Service Oriented Computing and Applications (2019) 13:237–249
https://doi.org/10.1007/s11761-019-00270-0

ORIGINAL RESEARCH PAPER

A focus on future cloud: machine learning-based cloud security

E. K. Subramanian[1] · Latha Tamilselvan[1]

Received: 7 June 2019 / Revised: 12 July 2019 / Accepted: 29 July 2019 / Published online: 12 August 2019

Abstract
Recent days have seen an apparent shift in most of the organizations moving towards using cloud environment and various cloud-based services. In order to protect and safeguard the transactions made by organizations over cloud environment, it is highly essential to provide a secure and robust environmental solution across cloud space. Existing approaches such as linear regression and support vector machine have been tried to promote cyber-security in the market by performing static verification of cloud user behaviour in order to identify pre-defined threats. Due to their static nature, these security solutions are restricted in their functionality. When it comes to access control, the decision making involves performing a permit or block operation. Also, the earlier methods face difficulties in terms of data protection over the endpoints which are not managed by the cloud. In order to solve the above-said problems, this paper is focused on designing a novel security solution for cloud applications using machine learning (ML) approaches. The main objective of this paper is to shape the future generation of cloud security using one of the ML algorithms such as convolution neural network because CNN can provide automatic and responsive approaches to enhance security in cloud environment. Instead of focusing only on detecting and identifying sensitive data patterns, ML can provide solutions which incorporate holistic algorithms for secure enterprise data throughout all the cloud applications. The proposed ML algorithm is experimented, results are verified and performance is evaluated by comparing with the existing approaches.

Keywords Machine learning · Cloud security · Malicious threats

1 Introduction

Cloud computing is one form of distributed computing paradigm that involves using the Internet to deliver a host of services. The services may be in the form of simple software which is developed to perform a specific task, or it may be an infrastructure that is shared across the Internet or any software-specific platform that is distributed across the Internet. Based on the above definition, cloud computing services can be categorized into three divisions, namely Software as a Service (SaaS), Infrastructure as a Service (IaaS) and Platform as a Service (PaaS). Typical cloud services are characterized by some common features. An important characteristic of cloud computing is service on demand which means that the end-user can request for the desired number of resources required to cater to the user request. In terms of SaaS, PaaS and IaaS, the service provides services like software, platform and infrastructures in accordance with the user demand. In all the services, the user's business data are uploaded where they comprise of personal information and business rules. The personal information and the business rules need to be secured and it increases the user satisfaction. Hence, it is essential to provide security in SaaS, PaaS and IaaS. Not only in services, in terms of unauthorized user, access permission, link or route, data and data storage, the security level is very low. Another striking aspect of cloud computing is that it is highly elastic in nature and thus provides organizations the flexibility to scale up or scale down the resources based on demand and hence brings about substantial cost reductions. Another significant aspect observed in cloud computing is that the resources are not permanently attached to the user; rather, the resources are measured at a highly granular level in terms of its utilization while catering to the cloud service and is charged based on its usage. This pay per use concept observed in cloud computing has significantly cut down the cost invested in procuring static

✉ E. K. Subramanian
eksdeal@gmail.com

Latha Tamilselvan
latha.tamilselvan94@gmail.com

[1] B S Abdur Rahman Crescent Institute of Science and Technology, Chennai, India

resources. These characteristics have seen an apparent shift of different industries towards using cloud-based services.

Cloud computing deployment models comprise of using private cloud models such as OpenStack and VMWare which cater to internal user requests, public cloud models such as Microsoft Azure, Google Cloud services and Amazon Web Services (AWS) and a combination of public and private cloud models termed as hybrid cloud models. However, when public-based cloud services are used, it caters to a large population base, and as such, the public cloud environment is multitenant in nature. The multitenant nature of cloud environment promotes sharing of information which in turn increases the threat of accessing other user contents and information. Several top organizations and industries still are hesitant to use cloud environment just for the fear that the business-sensitive information does not get shared across or compromised while using cloud environment. This aspect has in turn underlined the significance and importance of cloud security and the need for having sophisticated and advanced approaches to promote security in cloud environment. However, several cloud-based data encryption standards and policies deployed in recent years have played a vital role in promoting cloud security. Also, several access management techniques and advanced tools for identity detection, management and tracking have promoted enhanced security measures in cloud environment.

Machine learning (ML) is one of the interdisciplinary areas of artificial intelligence (AI) that is used for imparting decision-making capabilities into an artificial intelligence-based system [1]. Machine learning is an extension of convolution neural networks (CNNs) and enables the device or system to acquire knowledge to make decision by training the system with relevant data and making the system well equipped to handle different scenarios and make smart decisions. The primary advantages of machine learning are that they help in easy identification of patterns and trends in the existing flow without any human intervention. Also, the use of machine learning approaches provides a scope for continuous improvement especially when the system is volatile to changing data. Machine learning has been used across a wide range of applications that handle multivariant and multidimensional data. The recent trend observed is the fusion of advanced computing paradigms such as artificial intelligence and machine learning along with cloud computing to accomplish various tasks and also from the perspective of enhancing cloud security. Several research works are directed towards using machine learning algorithms and approaches to promote cloud security as indicated in the surveys conducted by the authors of [2, 3].

The general convention followed while developing machine learning-based cloud security models is to train the model with labelled intrusions and anomalies observed across the cloud network along with the normal behaviour. This is accomplished using several standard datasets and involves extracting the features of the data and exploiting these features to study the underlying patterns and behaviour. Some common datasets available are DRAPA, KDD, UNSW and ISOT. Several machine learning techniques and classifiers such as decision trees, regression analysis, support vector machines (SVM) and naïve Bayes classifier are used in conjunction with other security measures to enhance cloud security. Most of the machine learning applications are generally geared towards emerging fields like marketing, finance, sales and so on. From the literature survey, it is identified that there is no earlier research works have used CNN for security applications. The biggest enterprises like Facebook, Salesforce and Google already use advanced machine learning for their business to provide cyber-security. Though security provision is a challenging task, it needs full attention. Hence, this paper motivated to use advanced machine learning model CNN for security provision in cloud environment.

1.1 Contribution of the work

- Construct a CNN model for analysing the network traffic data.
- Initially CNN is used for training the model with a dataset (UNSW dataset).
- Test the model with a dataset (ISOT dataset).
- Experiment and evaluate the performance by comparing the obtained results.

A convolution neural network model is created for analysing the network traffic. Initially the CNN model is trained with UNSW dataset and will used for testing on ISOT dataset. Finally the performance is compared in terms of malicious/abnormal detection rate.

2 Literature review

Despite the apparent shift observed among users and industries towards using cloud enabled services, cloud security has been a serious challenge and a topic of long-standing debate. A detailed analysis and study of several threats in cloud environment have been carried out in [4, 5]. Current trends indicate the usage of machine learning approaches to promote cloud and network security as discussed in [6, 7]. Support vector machine (SVM)-based classifier is deployed in [8, 9] to enhance cloud security. Another machine learning classifier is commonly used to promote cloud security, and using this, cloud anomalies were detected in [10, 11]. Cloud network anomalies can be easily detected using a machine learning-based multilayer perceptron-based approach as discussed in [11]. As discussed in [12, 13], decision tree-based machine learning models have also been developed to classify the anomalies observed in cloud environment.

The authors of [14] have explored the possibilities and feasibility of using supervised machine learning algorithms for promoting security in cloud-based environment. In the work carried out in [15], the authors have analysed the vulnerabilities and explored the possibilities of securing cloud-based email systems. The authors in [16] have used an unsupervised learning-based automated model selection approach in cloud environment for cloud network analysis and auto-tuning. The authors of [17] have proposed a fuzzy-based semi-supervised learning approach for intrusion detection in cloud network. The authors of [18] have gone to the next level of using deep learning for cyber-physical intrusion detection across vehicles. The above works clearly underline the significant shift in trends towards using machine learning algorithms for promoting cloud security and also highlights the essential need for enhancing cloud security. The authors in [2, 3, 5] have used machine learning algorithms for cloud and network security. The authors in [3, 4] provided a detailed survey about ML-based cloud security. The survey was explained about the popularity of the ML approaches. The authors in [5] introduced a general approach derived from ML algorithms for security. Using the general model, a training process is carried out to label the abnormal and normal patterns present in the data. Similarly, while thinking of deep learning algorithms, the authors in [18] proposed convolution neural network for analysing different kinds of data such as data, images and texts posted in the social websites. The convolution neural network is applied on the multimodel data for learning and classifying the data to provide user certification. Based on the classification results, the users are evaluated and categorized. The overall precision obtained using existing CNN is 79.93%. And the authors in [19] proposed CNN is proposed for providing security to mobile applications. The authors proposed CNN for analysing the file directory for analysing the class files to check whether it is a malicious code or not. From the experimental results, the accuracy of code verification is 94%.

3 Limitations

Machine learning-based cloud security systems require huge volumes of datasets for training. Any new form of data entering into the system involves intensive training. In order to develop machine learning-based applications with high degree of accuracy and precision, it incurs considerable time and enormous resource utilization. At the same time, it is highly essential to ensure that the machine learning-based system is frequently tuned to identify new anomalies coming across the network. Also, the choice of machine learning algorithm plays a vital role in predicting the result. Any change of algorithm when applied over same dataset may interpret the result differently. Hence, careful choice of machine learning algorithm is highly needed. Also, machine learning-based algorithms are highly sensitive to errors and outlier data. Thus, proper data pre-processing and cleaning are highly essential.

4 Motivation

The proposed system is developed with an objective to enhance cloud security across distributed cloud-based environment using machine learning techniques. The system should be robust enough to detect any form of anomalies intruding into the network and should be smart enough to diagnose uncharacteristic behavioural patterns observed across the cloud network. At the same time, the system should be scalable to adapt to the changes observed across the cloud environment as well. The primary motivation behind using machine learning-based approach for cloud security is that they are highly accurate once they are properly trained and yield best possible results with high degree of precision and accuracy. At the same time, the machine learning-based cloud security system can be easily adaptable to the evolving changes by mere changing and tuning of input parameters without changing the core logic. Machine learning algorithms learn the entire input data automatically and extract the features for classification. Also, the CNN of machine learning algorithm has a greater number of hidden layers to learn and fetch the entire features, hidden information and other additional meta-information of the data. It helps to do accurate classification. Hence, this CNN is suggested as easier method than the current technologies.

So, this paper focused on implementing the advanced machine learning method such as convolution neural network (CNN) for analysing and classifying the abnormality. Various network traffic data are analyzed using CNN for classifying the data belongs to normal or abnormal activities. If the traffic data are abnormal, then it will not be permitted in the network; otherwise, the normal data will be permitted to transmit in the network.

4.1 Problem statement

The main problem considered in this paper is to provide security using CNN by analysing the network traffic data. The objective of this work is to devise a robust and highly secure machine learning-based cloud security by observing the data patterns and detecting the outliers and anomalies observed in the cloud environment using these machine learning approaches. The primary factor behind using machine learning-based approach to promote cloud security is to detect anomalies with high degree of precision and accuracy.

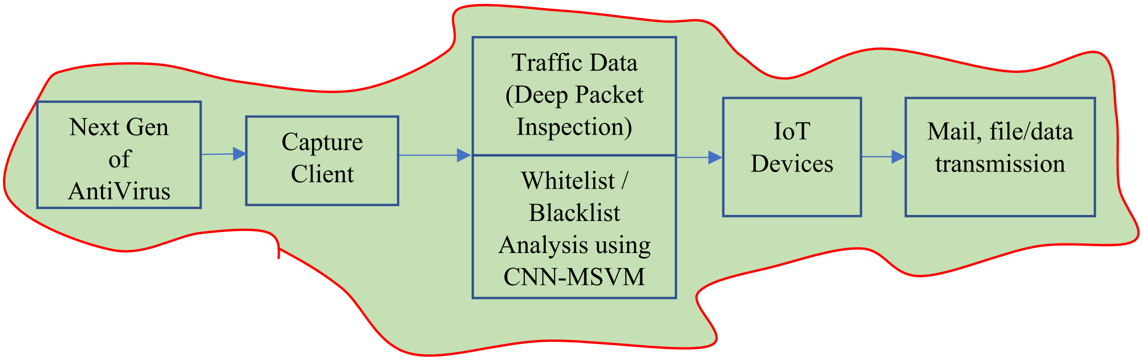
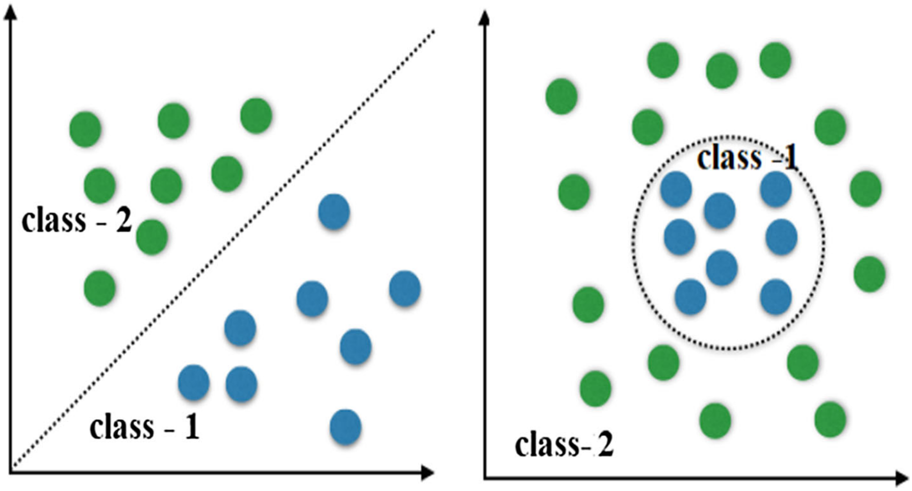

Next Gen of AntiVirus
Capture Client
Traffic Data (Deep Packet Inspection)
Whitelist / Blacklist Analysis using CNN-MSVM
IoT Devices
Mail, file/data transmission

Fig. 1 Proposed CNN-MSVM cloud security system

Fig. 2 Support vector machine (SVM)



The dataset is analysed by verifying the data flow rate, time interval during the communication and other parameters for identifying the threats occur in the dataset. The mitigation of the malicious threats is obtained by computing the true positive, true negative, false positive and false negative measures over UNSW and ISOT datasets.

The overall environmental structure is illustrated in Fig. 1. Figure 2 provides the overall architecture of this machine learning-based cloud security system. The striking aspect of this system is that it reaps the advantages provided by advanced decision-making and response techniques provided by machine learning classifiers and artificial intelligence (AI) at various points on the cloud network.

As shown in the figure, apart from securing the data and the underlying cloud environment using conventional antivirus software, operating system firewalls, etc., this proposed system uses the next-generation artificial intelligence methods to capture the client input and process them even at the entry point level itself to ensure no anomalous data seep into the cloud-based system. It must be understood that the security is provided at three levels, namely at the entry point where the client request is captured, security at the network edge, cloud platform and end point where the application user interface resides. Once the data are pre-processed and fetched into the system, each of the data packets are subjected to both encryption and decryption and analysed using machine learning algorithms to detect and alert anomalous packets. These packets are then sent across to the cloud layer, thus ensuring only the relevant packets are transmitted across the cloud platform.

4.1.1 SVM versus MSVM

SVM classifier can classify only two classes, but MSVM classifier can classify multiple classes present in the dataset. Figures 2 and 3 show the SVM classification, whereas Fig. 2 shows the multiclass SVM classifier.

Based on the hyperplane, the SVM classifier classifies two or more classes. Each hyperplane in MSVM classifies the data into multiple classes as class 1, class 2, class 3, etc. In this paper, MSVM is used to classify the multiple classes obtained from the dataset.

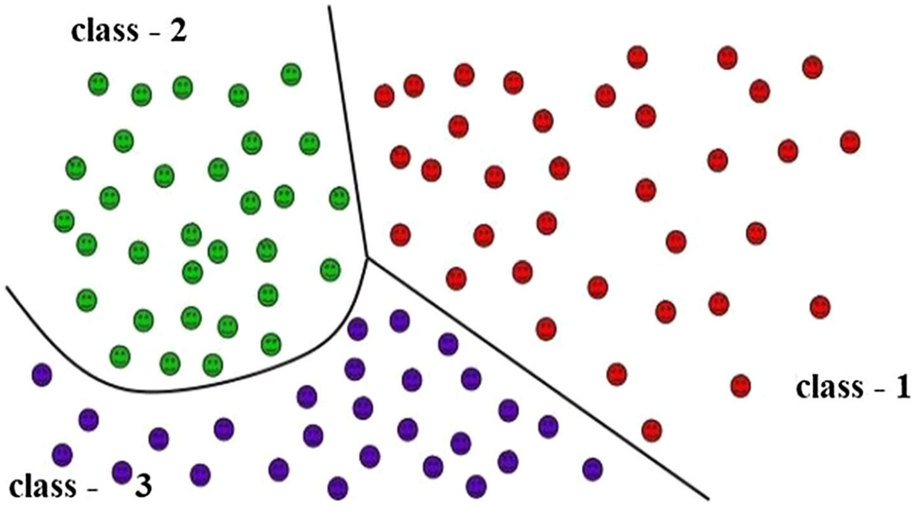


Fig. 3 Multiclass support vector machine (MSVM)

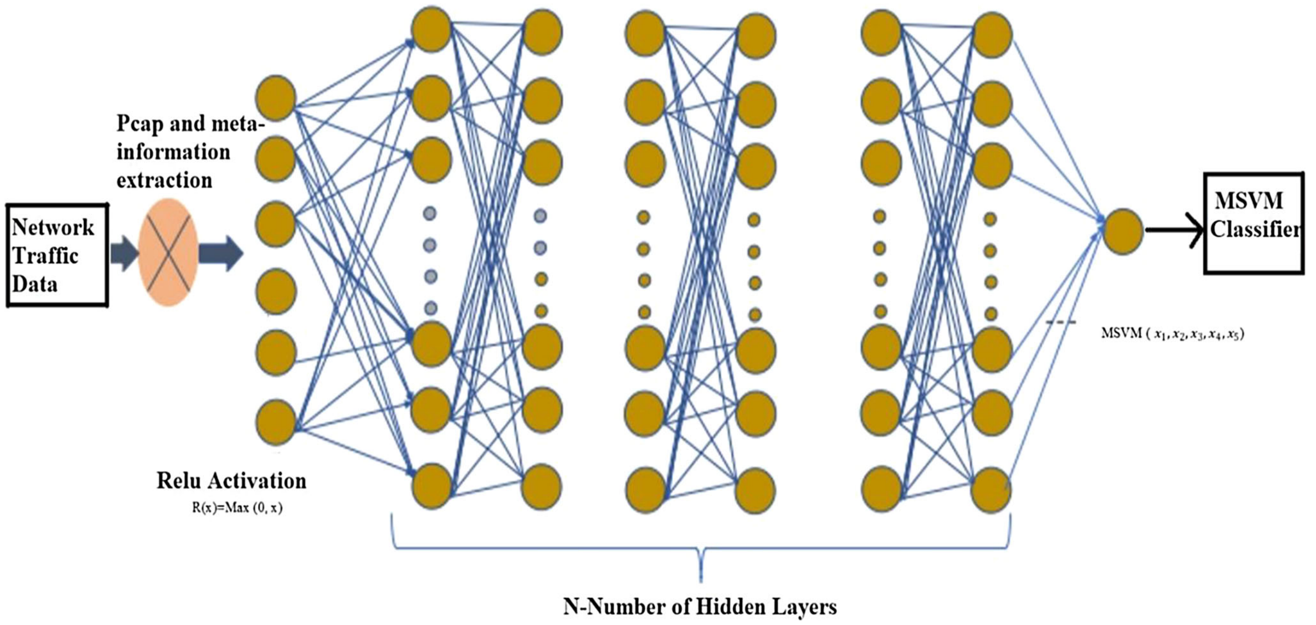

Network Traffic Data
Pcap and meta-information extraction
Relu Activation
R(x)=Max (0, x)
MSVM Classifier
MSVM (x_1, x_2, x_3, x_4, x_5)
N-Number of Hidden Layers

Fig. 4 CNN-MSVM functionality

4.2 Proposed CNN architecture

The proposed system comprises of using machine learning system-based classification to detect any anomalous behaviour based on the data analysis across the cloud environment. In this case, convolution neural network (CNN)-based architecture is used for classification of anomalies. The input information from cloud is fed to the convolution neural network system. This comprises of applying multiple layers of pooled resources as convolution layers, and the data are transformed into several forms across each convolution stage. The output of CNN comprises of different feature vectors that serve as the input to the multiclass support vector machine classifier. Generally, the fully connected layer in the CNN classifies the final vector data where its size is two or three, and it determines the abnormal classes automatically. But in this paper, the abnormality is defined as five classes in terms of source node, destination node, link reliability, time interval and request–response pattern. The number of output vector is also more. Hence, to classify the abnormal classes MSVM is applied on the output vector comes from CNN. It is well known that the classified result obtained from CNN is highly accurate. The output vector obtained from the CNN, is labeled using MSVM to identify the exact class of the data. This multiclass support vector machine classifier is highly trained to categorize the events across the cloud platform based on the data available and when the feature vectors are provided as inputs, the events in real-time cloud environment are categorized as normal events and anomalous events.

Table 1 Meta-data of the data [21]

Parameter	Parameter explanation
FIAT	Forward interarrival time, the time between two packets sent forward direction (mean, min, max, std)
BIAT	Backward interarrival time, the time between two packets sent backwards (mean, min, max, std)
FLOWIAT	Flow interarrival time, the time between two packets sent in either direction (mean, min, max, std)
ACTIVE	The amount of time a flow was active before going idle (mean, min, max, std)
IDLE	The amount of time a flow was idle before becoming active (mean, min, max, std)
FB PSEC	Flow bytes per second. Flow packets per second. Duration: the duration of the flow

The architecture of the CNN is illustrated in Fig. 4. The number of hidden layers is not limited, but in this paper, the number of hidden layers is varying from 1 to 10. When the number of hidden layers is 5, it is optimal. Relu is used for activating all the hidden layers. All the convolution and pooling layers are dense in nature, and the number of dimensions is 100. While data are processed in the CNN, the size of the data is reduced from matrix to vectors. Final vector represents the selected feature vector which is applied to MSVM for classifying whether the data are normal or abnormal.

4.3 CNN-based network: traffic data analysis

In this paper, the data feed into CNN are network traffic data. For comparison and performance evaluation, the data are taken from Habibi Lashkari et al. [20] and used in the experiment. The data comprise of several features extracted from the analysis. The data demonstrate the network/Internet traffic having meta-information on the data which are given in Table 1. Along with the above parameters, data flow parameters also involved. To understand the entire data structure and the experimental results, a sample model of the data is given in Fig. 5. The convolution neural network (CNN) has one input layer, output layer and a greater number of convolutions with pooling layers. As it overfits with the model, the input port (source IP port) and the output port (destination IP port) and the protocol fields are removed.

The set of all input data is feed into the input layer, where it converts the data into convolution vectors. Also, the entire features of the data are learned by the convolutional and pooling layers. All the convolutional pooling layers are also called as hidden layers, since they learn and extract the entire hidden information from the input data after some iteration of process. The output node is activated by MSVM function. MSVM is used to classify the output data using binary classification such as “Normal” or “Abnormal”.

In the CNN-MSVM, Keras and TensorFlow is used in the backend to train the CNN model efficiently. Finally, a cross-entropy is applied for calculating the loss in order to optimize the CNN. The proposed model is trained many times with several iterations. Figure 4 shows the functionality of CNN-MSVM model for increasing the performance of predicting the abnormal data and normal data with decreased loss for a greater number of epochs.

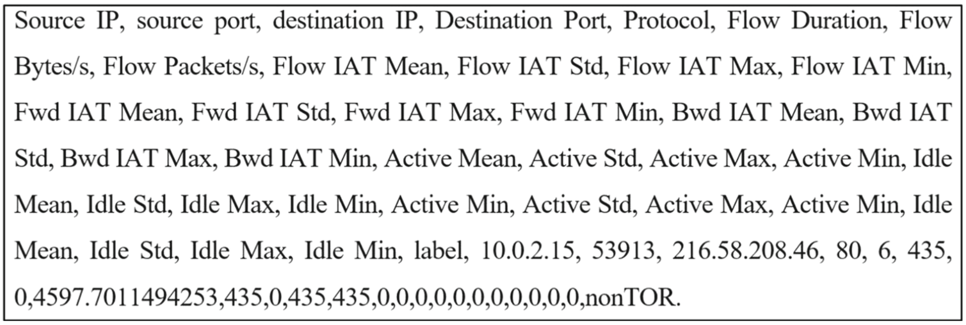
The output obtained using CNN-MSVM method is compared with the various existing supervised machine learning methods such as LR, SVM, NB and RF (Fig. 6). The performance measures compared with each other are precision, recall and F-score which determine the efficiency of the classification ability. The proposed CNN-MSVM has ability to identify and detect TOR class. Similarly, detecting the NON-TOR class is also important and classified. From the classified results, it is identified that the proposed CNN-MSVM method reduces the number of FP cases for NON-TOR. The comparison result is illustrated in Fig. 7.

4.4 Performance evaluation

Research works focused on creating machine learning for training and testing various datasets are very less. It is essential to investigate the robustness of ML models used in various real-time applications. This problem is also addressed in this paper by analysing the performance of ML algorithms by experiment on three different datasets such as UNSW [22, 23] and ISOT [24]. The above-said datasets are evaluated by cloud simulations. These datasets provide traffic information collected in a big cloud environment as a good example. To evaluate the performance of the dataset recent and popular methods such as regression, naïve Bayes, decision trees and support vector machines, these methods have been used widely in various fields including abnormality classification and cloud security [4–7, 11].

Initially the methods are trained using the UNSW dataset. The trained methods are again used to test the UNSW and ISOT datasets. Both training and testing processes are applied in the same simulation setup. Not only these two datasets, but also multiple datasets are applied in the model for training the ML methods; due to more comprehensiveness in testing the ML methods, it is not been applied in the earlier research works [25, 26]. This kind of testing of ML methods can increase the applicability and robustness for real-time scenarios. The experimental results obtained in this paper can be used for further research works focused in the similar supervised ML applications under cloud security and network security. So, it is concentrated and insisting that the trained ML methods over certain datasets must test different datasets for verifying ML method’s robustness. Hence, several ML methods are trained and labelled for UNSW datasets.

Fig. 5 Sample data



Source IP, source port, destination IP, Destination Port, Protocol, Flow Duration, Flow Bytes/s, Flow Packets/s, Flow IAT Mean, Flow IAT Std, Flow IAT Max, Flow IAT Min, Fwd IAT Mean, Fwd IAT Std, Fwd IAT Max, Fwd IAT Min, Bwd IAT Mean, Bwd IAT Std, Bwd IAT Max, Bwd IAT Min, Active Mean, Active Std, Active Max, Active Min, Idle Mean, Idle Std, Idle Max, Idle Min, Active Min, Active Std, Active Max, Active Min, Idle Mean, Idle Std, Idle Max, Idle Min, label, 10.0.2.15, 53913, 216.58.208.46, 80, 6, 435, 0,4597.7011494253,435,0,435,435,0,0,0,0,0,0,0,0,0,0,nonTOR.

Fig. 6 CNN-MSVM accuracy versus loss

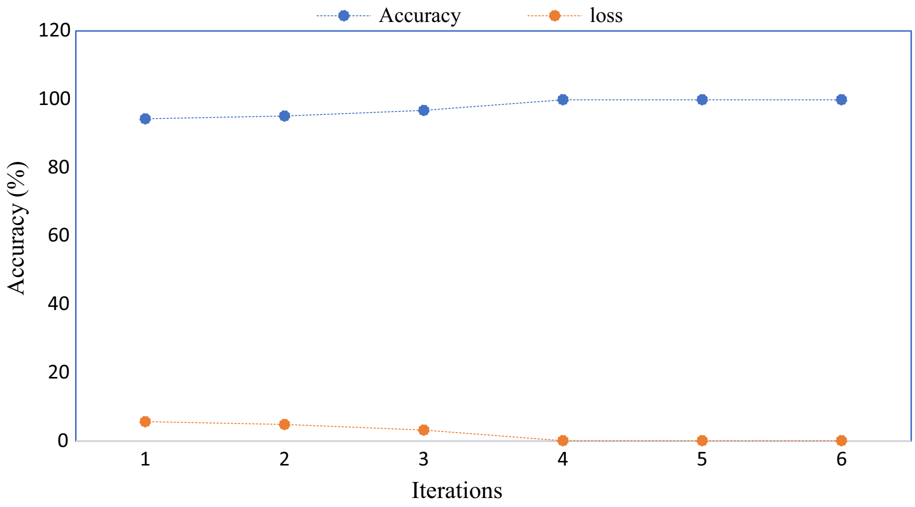
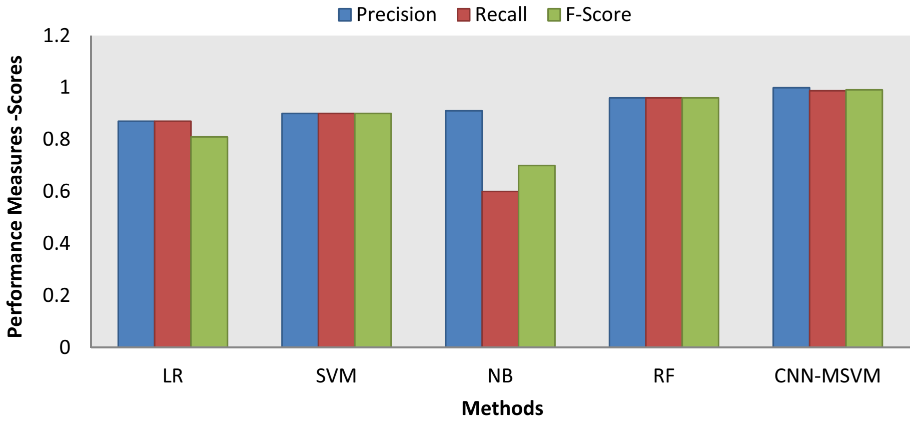


Fig. 7 Performance comparison



Then it will be used for testing other datasets taken from other research works.

The main dataset UNSW-NB-15 [8] has been created in Cyber Range Lab at Australian Centre for Cyber Security (ACCS) which is applied to generate the realistic normal and abnormal (synthetic contemporary) activities from the network traffic. The network traffic of 100 GB raw data is monitored and recorded by a TCP-DUMP tool. The monitored dataset comprises of nine types of malicious activities. Bro-IDS tool [23] is used with twelve algorithms implemented for generating K number of features ($k = 49$) including class label. In order to reduce the computational complexity, a portion of the dataset is considered for training process and the remaining portion of the dataset is taken for testing process. The training and testing datasets are stored in the name of “UNSW_NB15_training-set.csv” and “UNSW_NB15_testing-set.csv”. These two datasets are used for implementing and training the ML methods. The size

Table 2 Dataset information (UNSW)

Dataset	Total records	Normal	Abnormal
Training process	180,000	60,000	120,000
Testing process	83,000	40,000	43,000
Total data size	260,000	95,000	165,000
Data size in %	100	40	60

Table 3 Dataset information (ISOT)

Traffic type	Unique flows	Percentage
Malicious	56,000	3.5
Normal	170,000	96.5
Total	226,000	100

Table 4 Performance measures used as prediction constraints of ML

Total population	Predicted condition positive (anomalous)	Predicted condition negative (normal)
Positive (1, anomalous)	True positive (TP)	False negative (FN)
Negative (0, normal)	False positive (FP)	True negative (TN)

of the training dataset is 180,000, and the testing is 83,000 records. The complete statistics of the dataset is given in Table 2. The table shows the normal as well as abnormal data packets exist in the dataset and can be used for evaluating the performance of our proposed approach. The normal data of ISOT dataset (see Table 3) are taken from [24, 27]. The abnormal data of ISOT are taken from Traffic Lab at Ericsson Research [28] and Lawrence Berkeley National Lab [29]. The total data are recorded for 4-month time period, from a network having 22 subnets. The entire details of the data in ISOT dataset considered for our experiment are given in Table 3. The total number of malicious data existing in the ISOT dataset is 56,000, and normal is 170,000, that is, 3.5% of malicious from 100% of total data, and the remaining 96.5% is normal. Hence, the malicious data are less; we can obtain the malicious from ML classification speedily. The performance of the ML classification is increased by binary classification, where 1 indicates the abnormal and 0 indicates the normal. The set of all performance measures used and calculated from the experiment is given in Table 4.

5 Results and analysis

The ML implementation is experimented over the above-said datasets and compared with the performance on various aspects. To experiment the supervised machine learning methods, two data sets UNSW and ISOT are used. One dataset is used for training process, and other dataset is used for testing process. From the existing SVM method, it is extended as MSVM for implementation. Also, other algorithms from ML methods such as J48 in decision trees [30], naïve Bayes [31], logistic regression [32] and various kernels of SVM such as SVM-RBF, SVM-linear and SVM-polynomial [8, 30] are chosen for comparison. All these algorithms are selected from major supervised ML methods, and all the methods have been used extensively in network security applications [33, 34]. The training process is applied using ML method on UNSW dataset using WEKA software [34]. Finally, the performance of the method is compared with one another and the results are given in Fig. 8.

The obtained results determine the performance of the proposed ML method by comparing with the other results [9, 11]. Figure 8 shows the overall accuracy obtained from the experiment on UNSW dataset, which is also used by the existing approaches. It is observed that compared with the other ML methods the proposed MSVM method obtained better accuracy (96.45%) and greater value and the lowest value is obtained using SVM-polynomial method. The reason for the highest accuracy obtained using MSVM is the feature extraction and selection done by the convolution neural network, one of the best architectures of the ML methods. MSVM simply classifies the final feature vector given by the output layer in the CNN architecture. Including this, the performance comparison is also obtained by calculating and comparing separately using TP, FP, TN and FN values (see Table 4). In the binary classification results, 0 represents normal and 1 represents abnormal data packet. The comparison results using the performance measures are given in Fig. 9. Figure 9 shows that the % of TP obtained using MSVM is 98.6% for the abnormal data packet and FN is 1.4% for normal data packet. Also, it is identified that the accuracy of TP and FN classification using MSVM is high compared with the other methods. True positive is the measure which classifies the number abnormal data available in the dataset as the abnormal data correctly. FN is the measure which classifies the number of abnormal data available in the dataset as normal data incorrectly. In order to verify the performance of using TP and FN, the classified output using the implementation is cross-comparison with the dataset, since the dataset has pre-classified data. Now the accuracy obtained by calculating the overall performance is compared with the accuracy obtained by performance measure calculation. The accuracy obtained using both the ways is merely equal to one another. Thus, it identified that the proposed method is efficient and the implementation procedure is correct.

Similarly, TN and FP calculation-based performance comparison is obtained for UNSW dataset using the proposed CNN-MSVM model given in Fig. 10. It is observed that the accuracy of TN and FP calculation using CNN-MSVM classifier is high than the other existing ML models. CNN-

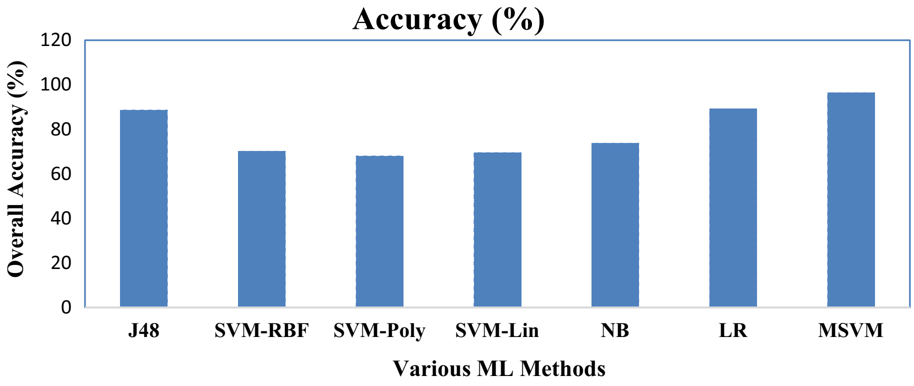


Fig. 8 Overall accuracy comparison on UNSW dataset

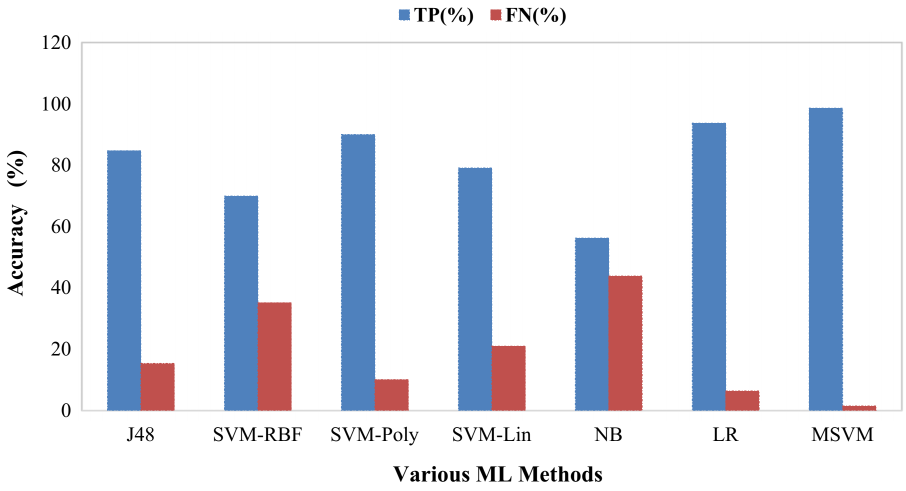


Fig. 9 % of TP and FN comparison on UNSW dataset

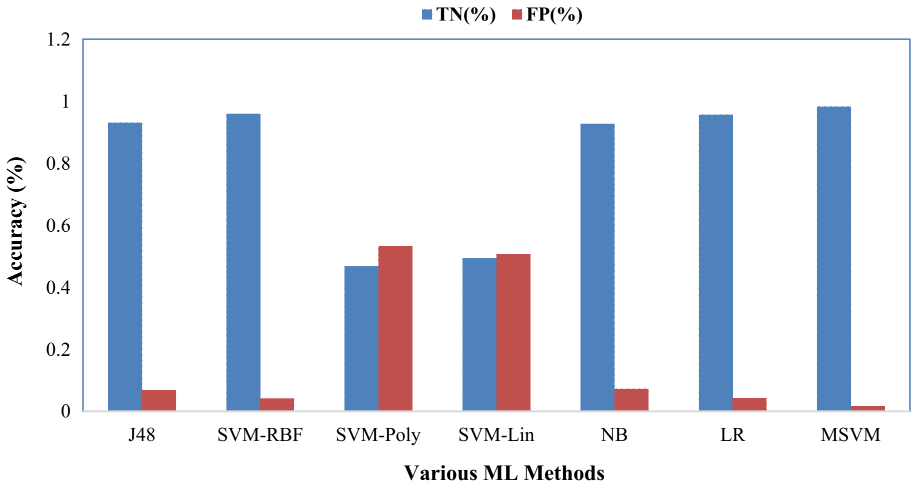


Fig. 10 % of TN and FP comparison on UNSW dataset

MSVM classifier obtained 98.3% in classifying normal data packets. TN is the measure of classifying entire amount of normal data as normal correctly and it is cross-comparison with the pre-classified results given in the dataset. Now the accuracy obtained by calculating the overall performance is compared with the accuracy obtained by performance measure calculation. The accuracy obtained using both the ways is merely equal to one another. Thus, it identified that the proposed method is efficient and the implementation procedure is correct. Also, it is identified that the value of TP is increased; due to classifying, the abnormal data packets have the most importance in security applications.

For large volume of data, and data with a greater number of features, a probability threshold is used for categorizing the normal and abnormal data packets. For example, the value 0.5 is used to compare the data. If the final probability of packet is greater than 0.5, then it is normal, else abnormal. Though the false negative chances are high in cloud, the probability

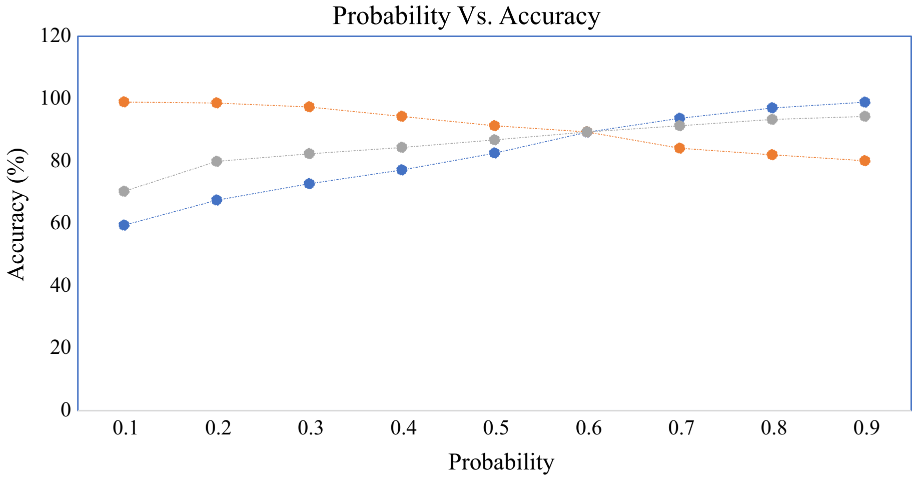


Fig. 11 Varying frequency threshold with UNSW dataset

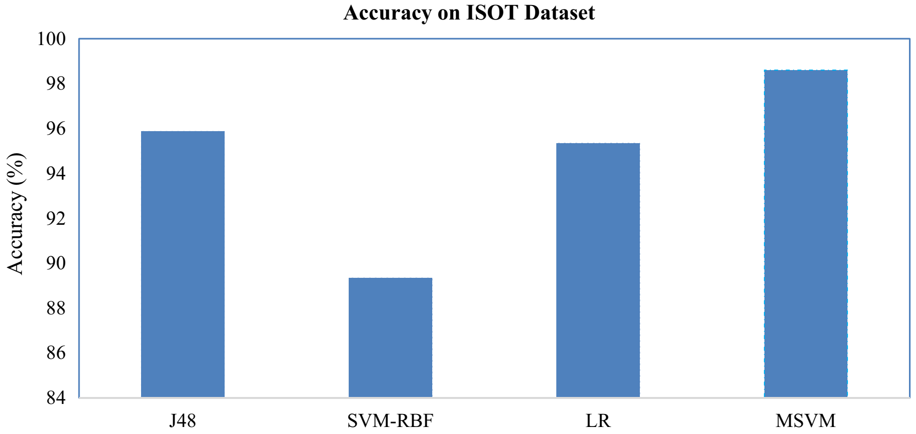


Fig. 12 Accuracy comparison on ISOT dataset

threshold value is assigned from 0.1 to 0.9, and based on this, the TP and TN are calculated and given in Fig. 11. It is observed that the threshold value increases; then, the TP rate increases, where TN decreases. For the threshold values 0.7 and 0.8, the % of TP is high and the accuracy is acceptable. Hence, it is concluded that the accuracy of identifying normal packets is 80.9% which is acceptable. From the obtained results given in Figs. 8, 9, 10 and 11, it is satisfactory on UNSW dataset regarding training- and testing-based classification. To verify the robustness of the ML approaches the experiment is carried out on different scenarios using ISOT Data set.

The results obtained on ISOT dataset is given in Figs. 12, 13, 14 and 15. The overall accuracy obtained from proposed CNN-MSVM is compared with the J48, SBM-RBF and LR. From the compared results, it is noticed that the proposed CNN-MSVM obtained 98.6% and outperforms than the others. The least % of accuracy obtained by LR is 90%. Then the based on the threshold value the TP rate is calculated and the result is given in Fig. 8. CNN-MSVM obtained 99.87% of TN value.

Figure 12 shows the overall accuracy obtained from the experiment on ISOT dataset, which is also used by the existing approaches. It is observed that compared with the other ML methods, the proposed MSVM method obtained better accuracy (98.6%) and greater value and the lowest value is obtained using SVM-RBF method. The reason for the highest accuracy obtained using MSVM is the feature extraction and selection is done by the convolution neural network, one of the best architectures of the ML methods. MSVM simply classifies the final feature vector given by the output layer in the CNN architecture. Including this, the performance comparison is also obtained by calculating and comparing separately using TP, FP, TN and FN values (see Table 4).

The binary classification results 0 represents normal and 1 represents abnormal data packet. The comparison results using the performance measures are given in Fig. 13. Figure 13 shows that the % of TP obtained using MSVM is 63.9% which denotes the abnormal data packet and FN is 36.1% of normal data packet. Also, it is identified that the accuracy of TP and FN classification using MSVM is high compared with the other methods. True positive is the measure which classifies the number abnormal data available in

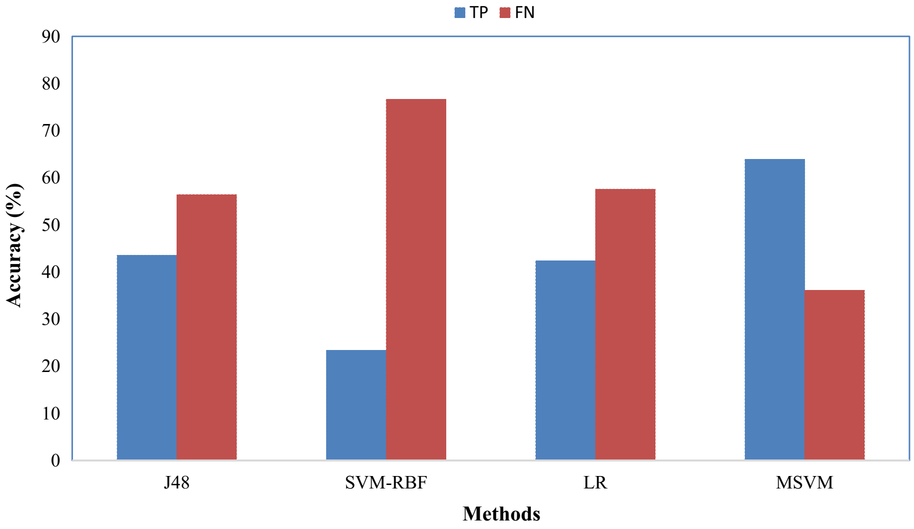


Fig. 13 % of TP and FN comparison on ISOT dataset

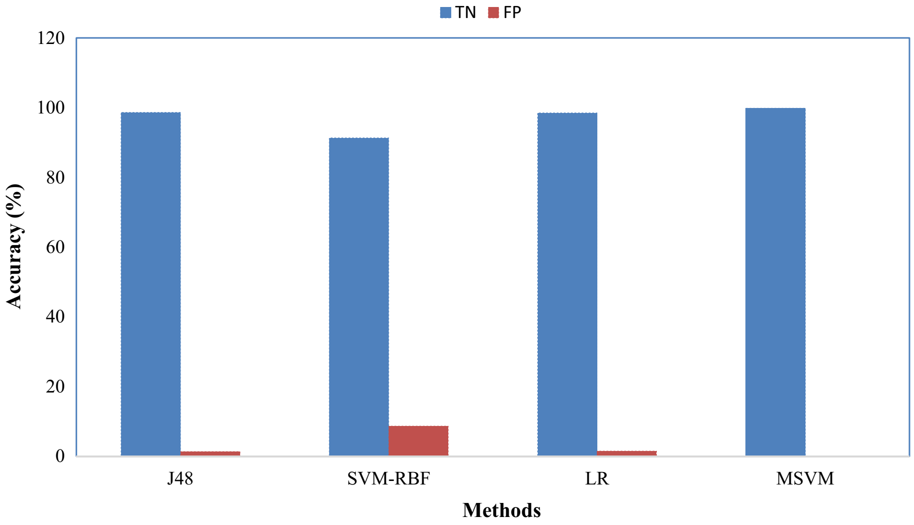


Fig. 14 % of TN and FP comparison on ISOT dataset

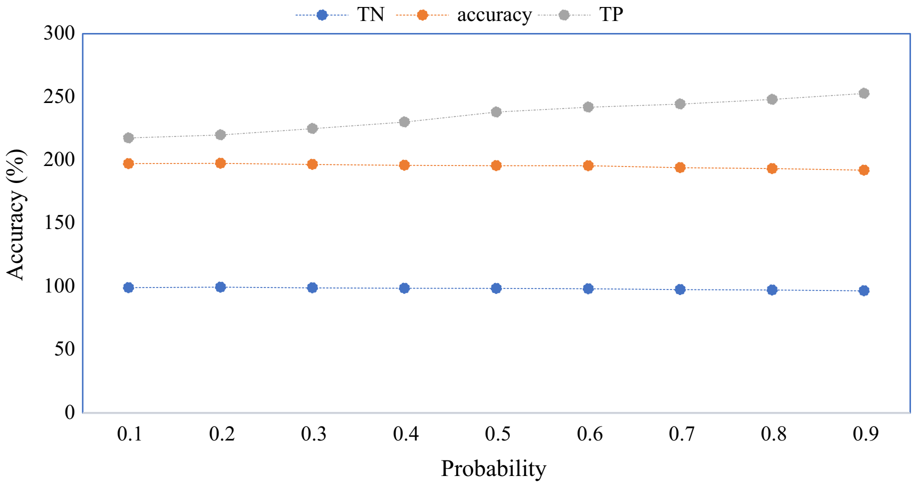


Fig. 15 Varying frequency threshold with ISOT dataset

the dataset as the abnormal data correctly. FN is the measure which classifies the number of abnormal data available in the dataset as normal data incorrectly. In order to verify the performance of using TP and FN, the classified output using the implementation is cross-comparison with the dataset, since the dataset has pre-classified data. Now the accuracy obtained by calculating the overall performance is compared with the accuracy obtained by performance measure calculation. The accuracy obtained using both the ways is merely equal to one another. Thus, it identified that the proposed method is efficient and the implementation procedure is correct.

Similarly, TN and FP calculation-based performance comparison is obtained for UNSW dataset using the proposed CNN-MSVM model given in Fig. 14. It is observed that the accuracy of TN and FP calculation using CNN-MSVM classifier is high than the other existing ML models. CNN-MSVM classifier obtained 99.87% in classifying normal data packets. TN is the measure of classifying entire amount of normal data as normal correctly and it is cross-comparison with the pre-classified results given in the dataset. Now the accuracy obtained by calculating the overall performance is compared with the accuracy obtained by performance measure calculation. The accuracy obtained using both the ways is merely equal to one another. Thus, it identified that the proposed method is efficient and the implementation procedure is correct. Also, it is identified that the value of TP is increased; due to classifying, the abnormal data packets have the most importance in security applications.

For large volume of data, and data with a greater number of features, a probability threshold is used for categorizing the normal and abnormal data packets. For example, the value 0.5 is used to compare the data. If the final probability of packet is greater than 0.5, then it is normal, else abnormal. Though the false negative chances are high in cloud, the probability threshold value is assigned from 0.1 to 0.9, and based on this, the TP and TN is calculated and given in Fig. 15. It is observed that the threshold value increases; then, the TP rate increases, where TN decreases. For the threshold values 0.7 and 0.8, the % of TP is high and the accuracy is acceptable. Hence, it is concluded that the accuracy of identifying normal packets is 80.9% which is acceptable. Finally, the proposed approach analyses the entire dataset and obtained 98.6% of accuracy in malicious detection in ISOT dataset and 98.87% of accuracy in UNSW dataset and it is shown in Table 5.

From the obtained results given in Figs. 12, 13, 14 and 15, it is satisfactory on UNSW dataset regarding training- and testing-based classification. Like we mentioned earlier in the paper that verifying the robustness and applicability of ML models the different experiments are carried out on different scenarios on ISOT and the results are compared with one another.

Table 5 Malicious detection accuracy

Dataset	Accuracy in malicious detection (%)
UNSW	98.87
ISOT	98.6

6 Conclusion

This paper proposed CNN-MSVM method for network traffic analysis. Since the data travel in various cloud scenarios, and more than one data centres under various operating conditions, it is necessary to verify the malicious activity over the data. To do that, the data are deeply analysed. Hence, extended machine learning method such as CNN-MSVM is used for providing cloud security. The main objective of this paper is to identify and detect abnormal activities by analysing the network traffic data. Two different sets of data such as TOR and UNSW and ISOT are taken for experimenting and performance evaluation on CNN-MSVM method. The experiment is carried out by feeding all the input data (network traffic data) into the input layer of the CNN model. The main objective of this work is to provide a better learning approach for data analytics regarding cloud security. Also, it is focused on applying extensive amount of research work using CNN incorporated with MSVM approach. From the experimental results and performance comparison, it is identified and concluded that the extended supervised machine learning methods are highly suitable and applicable in real-time cloud applications. The efficiency is verified by experiment on various datasets and identified that the proposed CNN-MSVM method is satisfactory.

7 Future work

In future, the cloud security is provided using various deep learning models and the performance is compared.

Some of the other datasets can also be used to verify the performance of the proposed approach. In future work, it can be experimented on the following datasets and the performance is evaluated.

- UCI ML datasets: University of California, Irvine, has a collection of machine learning datasets across many fields. Within their collection, there are security relevant datasets if you search for topics such as spam, phishing, etc.
- HTTP dataset CSIC 2010: “The HTTP dataset CSIC 2010 contains thousands of web requests automatically generated. It can be used for the testing of web attack protection systems. It was developed at the ‘Information Security Institute’ of CSIC (Spanish Research National Council)”.
- eXpose: Deep neural network This is an open-source deep neural network project that attempts to detect malicious URLs, file paths and registry keys with proper training. Datasets can be found in the data/model’s directory the in the sample_scores.json files.
- KDD Cup 1999: Computer network intrusion detection The goal of the KDD Cup competition in 1999 was to learn a predictive model (i.e. a classifier) capable of distinguish-

ing between legitimate and illegitimate connections in a computer network. This is a link to the large dataset used for that competition. The other tabs on the page provide additional context on the data.

References

1. https://www.analyticsvidhya.com/blog/2018/07/using-power-deep-learning-cyber-security/
2. Tsai C, Hsu Y, Lin C, Lin W (2009) Intrusion detection by machine learning: a review. Expert Syst Appl 36(10):11994–12000
3. Fernandes D, Soares L, Gomes J, Freire M, Inácio P (2014) Security issues in cloud environments: a survey. Int J Inf Secur 13(2):113–170
4. Kandukuri B, Paturi V, Rakshit A (2009) Cloud security issues. In: IEEE international conference on services computing, pp 517–520
5. Almulla S, Yeun C (2010) Cloud computing security management. In: 2nd international conference on ICESMA. IEEE, pp 1–7
6. Palivela H, Chawande N, Wani A (2011) Development of server in cloud computing to solve issues related to security and backup. In: IEEE CCIS, pp 158–163
7. Roshke S, Cheng F, Meinel C (2009) Intrusion detection in the cloud. In: Eighth IEEE international conference on dependable, autonomic and secure computing, pp 729–734
8. Laura A, Moro R (2008) Support vector machines (SVM) as a technique for solvency analysis. DIW Berlin discussion paper
9. Zhang X, Zhao Y (2013) Application of support vector machine to reliability analysis of engine systems. Telkomnika 11(7):3352–3560
10. Haykin S (2009) Neural networks: a comprehensive foundation, 2nd edn. Prentice Hall, Englewood Cliffs, NJ
11. Michalski R, Carbonell J, Mitchell T (2013) Machine learning: an artificial intelligence approach. Springer, Berlin
12. Breiman L, Friedman J, Olshen R, Stone P (1984) Classification and regressing trees. Wadsworth International Group, Belmont, CA
13. Stein G, Chen B, Wu A, Hua K (2005) Decision tree classifier for network intrusion detection with GA-based feature selection. In: Paper presented at the proceedings of the 43rd annual southeast regional conference, Kennesaw, GA
14. Bhamare D, Salman T, Samaka M, Erbad A, Jain R (2016) Feasibility of supervised machine learning for cloud security. In: 2016 international conference on information science and security (ICISS), Pattaya, pp 1–5
15. Ayodele T, Adeegbe D (2013) Cloud based emails boundaries and vulnerabilities. In: 2013 science and information conference, London, pp 912–914
16. Karn RR, Kudva P, Elfadel IA (2019) Dynamic autoselection and autotuning of machine learning models for cloud network analytics. IEEE Trans Parallel Distrib Syst 30(5):1052–1064
17. https://www.ixiacom.com/products/professional-services/test-as-a-service-taas
18. Hong T, Choi C, Shin J (2017) CNN-based malicious user detection in social networks. Concurr Comput. https://doi.org/10.1002/cpe.4163
19. Kuo W-C, Lin Y-P, Chang C-Y, Lin C-C, Lin H-H (2019) Malware detection method based on CNN. In: New trends in computer technologies and applications
20. Habibi Lashkari A, Draper Gil G, Mamun M, Ghorbani A (2017) Characterization of Tor traffic using time based features, pp 253–262. https://doi.org/10.5220/0006105602530262
21. Javaid A, Niyaz Q, Sun W, Alam M (2015) A deep learning approach for network intrusion detection system. EAI Endorsed Trans Secur Saf. https://doi.org/10.4108/eai.3-12-2015.2262516
22. Nour M, Slay J (2015) UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: Military communications and information systems conference (MilCIS). IEEE
23. Nour M, Slay J (2016) The evaluation of network anomaly detection systems: statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. Inf Secur J Glob Perspect 25:1–14
24. Sherif S, Traore I, Ghorbani A, Sayed B, Zhao D, Lu W, Felix J, Hakimian P (2011) Detecting P2P botnets through network behavior analysis and machine learning. In: Proceedings of 9th annual conference on privacy, security and trust (PST2011)
25. Shiravi A, Shiravi H, Tavallaee M, Ghorbani A (2012) Toward developing a systematic approach to generate benchmark datasets for intrusion detection. Comput Secur 31(3):357–374
26. Mchugh J (2000) Testing intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln laboratory. ACM Trans Inf Syst Secur 3:262–294
27. French Chapter of Honenynet. http://www.honeynet.org/chapters/france
28. Szabó G, Orincsay D, Malomsoky S, Szabó I (2008) On the validation of traffic classification algorithms. In: Proceedings of the 9th international conference on passive and active network measurement, PAM. Springer, Berlin, pp 72–81
29. LBNL Enterprise Trace Repository (2005) http://www.icir.org/enterprise-tracing
30. Peddabachigari S, Abraham A, Thomas J (2004) Intrusion detection systems using decision trees and support vector machines. Int J Appl Sci Comput 11:118–134
31. Jin B, Wang Y, Liu Z, Xue J (2011) A trust model based on cloud model and Bayesian networks. Procedia Environ Sci 11(Part A):452–459
32. Jordan A (2002) On discriminative vs. generative classifiers: a comparison of logistic regression and naïve Bayes. In: Advances in neural information processing systems, vol 14, p 841
33. Modi C et al (2013) A survey of intrusion detection techniques in cloud. J Netw Comput Appl 36(1):42–57
34. Waikato environment for knowledge analysis (WEKA) version 3.5.7. http://www.cs.waikato.ac.nz/ml/weka/. June 2008