Development of IDS Using Supervised Machine Learning

Indrajeet Kumar, Noor Mohd, Chandradeep Bhatt, and Shashi Kumar Sharma

Abstract In the era of modern lifestyle, the internet and networking are essential things for everyone. With the help these facilities everyone can exchange information between intranet and internet-connected people. During the information exchange, so many intermediate devices are involved, so that the security of information or data is primary concern for each and every involved system. Attackers or intruders belong to inside the network or outside of the network. To detect an intruder or attacker an intrusion detection system (IDS) has been proposed for the detection of normal and attack data packets for a network. In this work, KDD-99 dataset is used for the development of IDS. A total set of 32,640 samples are considered, in which 12,440 samples of normal and 20,200 samples of attack class are used. These samples are further bifurcated into training and testing set in balanced manner. Thus, 16,320 samples (normal: 6220 and attack: 10,100) are used for training set and same number of set is used for the testing set. For the supervised learning, SVM and kNN classifiers are used to differentiate between normal data packets and attack data packets with PCA as dimensionality reduction. After the successful completion of experiments, it has been found that PCA-kNN yields maximum accuracy of 90.07% at *pc* value of 5 using cosine distance.

Keywords Intrusion detection system · Supervised learning · SVM classifier · kNN classifier · Principal component analysis

I. Kumar (✉) · C. Bhatt · S. K. Sharma
Graphic Era Hill University, Dehradun, Uttarakhand, India
e-mail: erindrajeet@gmail.com

C. Bhatt
e-mail: bhattchandradeep@gmail.com

S. K. Sharma
e-mail: 12sksharma@gmail.com

N. Mohd
Graphic Era Deemed to be University, Dehradun, Uttarakhand, India
e-mail: noormohdcs@gmail.com

Research scholar, GBPIET, Pauri Garhwal, Uttarakhand, India

© Springer Nature Singapore Pte Ltd. 2020

M. Pant et al. (eds.), *Soft Computing: Theories and Applications*,
Advances in Intelligent Systems and Computing 1154,
https://doi.org/10.1007/978-981-15-4032-5_52

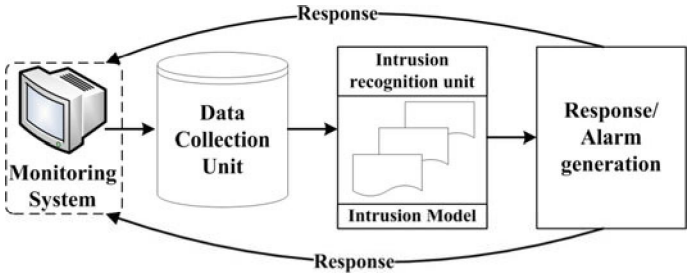

Fig. 1 IDS architecture

1 Introduction

The security and unauthorized accessed of data is still an open research area for the researchers. There are so many research communities working on this problem and so many systems had been already developed. Among them, Intrusion Detection System (IDS) is an authentic outcome of research development. It is a system used for monitoring traffic in the network and protecting the network from suspicious activity. It refers to as a tool, method, and software application used to find out the unauthorized network activities. It is typically a type of software installed around a system or network device that protects the overall network.

Intrusion Detection was initially introduced in the early 1980s succeeding the evolution of the internet with surveillance and monitoring. IDS was put in place as formal research when a technical report wrote by James Anderson for the U.S. Air force [1, 2]. Later on, Denning [3] focused on the implementation part of IDS and observed as major research recognition in 1987. Until the present day, it has been followed by many researchers. The basic building block and architecture of the IDS system is shown in Fig. 1.

2 Related Work

After the study of past research, it had been observed that IDS involve two broad categories on which researchers have worked named as signature-based and anomaly-based detection [4]. Signature-based IDS are also called as rule-based IDS, in which a set of predefined rules are used for differentiation between various attacks. If the behavior of the network activity is matched to any deviation from predefined rules, then it will be predicted as an attack. However, rule-based system is unable to detect new security attack or those attacks which have no predefined rules [5]. Therefore, the development of signature-based IDS is also suitable for the machine learning concept. In this research era, so many studies were performed and reported few major contributions.

Initially, an intrusion detection system is proposed by the author [6] that is capable of detecting known packet. This IDS model was based on the concept of genetic programming. It was effective against a variety of attacks such as a denial of services (DOS) and the unauthorized access. IDS used the variant of GP such as linear GP, multi-expression programming (MEP), and genetic expression (GEP). The proposed IDS model based on neural network is addressing the detection of anomaly-based attacks. A similar type of attack has been also explored in the study [7] and optimized results are reported. Further, a distributed intrusion detection system consists of multiple IDS over a big network with a centralized server that facilitates advanced network monitoring. The concept of fuzzy rule based IDS system was developed [8]. The fuzzy rule set IDS showed outstanding performance with respect to the existing IDS. Another important approach decentralized rule-based is used for the development of IDS is found in the study [9]. The proposed system is limited to detecting only routing-based attacks.

After the study of literature, it has been observed that the prediction of data packets or network traffic activity is a kind of problem that can be significantly solved by the concept of machine learning techniques. Therefore, authors attempt to design an IDS using *k*-means technique to detect if the network traffic is a type of attack or not. The proposed model used KDD Cup 1999 dataset for training and testing purposes. The author didn't use feature selection technique in the proposed work to select the prime attributes [10]. It has been also observed that PCA is used for selecting the prime attributes [11]. In this study, *k*-mean is used for the clustering to design IDS. The concept of machine learning for IDS development has been also explored in the studies [7, 12–16]. In these studies, SVM, Naive Bayesian, Decision Tree, Random Forest ANN, and neuro-fuzzy classifiers are used. The study reported in [7] shows the performance evaluation of IDS designed by classification techniques like ANN and SVM.

The study reported in [17] attempts to design an IDS using unsupervised learning which is capable of detecting unknown attacks. In this, Markov model is used, which learns the time-related changes. The proposed model was able to detect the changes and events with respect to time, so that any malicious activity can be recognized. It has been also found that the IDS had been designed for Big Data Environment [18]. In this work, the authors performed data preprocessing and normalization to figure out the improved IDS. Decision tree classifier is used and compared the performance of Naive Bayesian and kNN classifier based IDS. A similar type of work has been also done in the study [19], in which SVM multi-class classifier is used. For gaining higher accuracy, optimizations techniques like artificial bee colony, GA, and PCA are used [20].

It has been also found that the open-source software suite called "*Weka*" has been used to design IDS and evaluate the system [21]. In this work, artificial neural network classifier is used and observed the role of number of hidden layers. After the experiments, it has been concluded that the prediction rate varies with the number of hidden layers.

It has been also observed that the concept of deep learning is used for the development of IDS. In the study [22], authors have been developed an IDS using the

concept of deep learning. The proposed work applied LSTM architecture and RNN using the KDD Cup 1999 dataset. In this work, the size of hidden layers is varied and the outcome for accuracy and learning rate, author changing the size of the hidden layer. A similar type of concept has been also used in the study [23]. In this study, PCA-SVM is used and gives better results. This work is further extended with the help of deep learning model autoencoder. The work explained in the study [24] designed an IDS using RNN deep learning model noted as RNN-IDS.

In the present work, an IDS model is designed with the help of SVM and kNN classifier. Initially, a feature set of length 41 is used and passed to the SVM and kNN classifier. For SVM different kernel functions are used and optimum results are reported. Similarly, kNN classifier is used for the same feature vector length for different distance metrics. PCA-kNN classifier is used for different principal component and obtained results are reported.

3 Materials and Methods

The proposed model comprises of three sections such as (a) Network data packet collections, (b) intrusion detection engine, and (c) alert generation. The description of each section is given here.

3.1 Network Data Packet Collection

This section is used for the dataset collection on which tedious experiments have been conducted. The used dataset for the proposed IDS model is taken from the KDD-99 dataset [25], which is comprised of 4 GB archived raw samples. The network traffic of DARPA net is monitored through *tcpdump* tool for seven weeks. The complete KDD dataset consist of seven million of samples [26]. Each sample is represented in the form of 41 attributes also called features. The brief summary of KDD dataset is given in Table 1.

From Table 1, it has been observed that KDD-99 dataset is consisting of 4,898,431 samples having 972,781 samples of normal class and 3,925,650 samples of attack class. It is also mentioned that the attack class is a combination of four types of attacks like DOS, R2L, U2R, and probing. In this experiment, a total set of 32,640 samples are considered, in which 12,440 samples of normal and 20,200 samples of attack class are used. These samples are further bifurcated into training and testing set in a balanced manner. Thus, 16,320 samples (normal: 6220 and attack: 10,100) are used for training set and the same number of the set is used for the testing set.

Table 1 Description of dataset

Type		No. of samples	Used samples	Dataset bifurcation	
				Training	Testing
Normal		972,781	12,440	6220	6220
Attack	DOS	3,925,650	20,200	10,100	10,100
	R2L				
	U2R				
	Probing				
Total		4,898,431	32,640	16,320	16,320

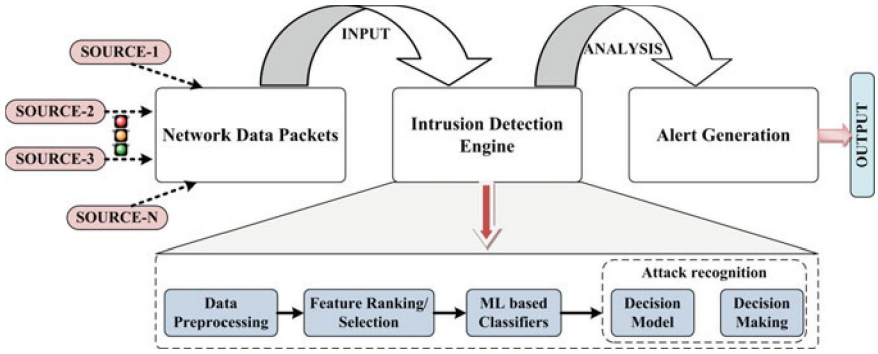

Fig. 2 Development of ML-based IDS

3.2 Intrusion Detection Engine

Intrusion detection engine is the heart of any IDS, which is used to generate an alarm signal for input samples on the basis of their decision rules. This section is consisting of a set of rules or machine learning algorithms that are widely used for the development of any decision-making system. In this work, machine learning based classification techniques are used to design IDS. The structure of machine learning based IDS is shown in Fig. 2.

3.3 Alert Generation

This section is used for the alert signal generation. After the analysis of activities at the detection engine, an alarm signal is generated for the particular activity. According to the generated alarm signal required action has been taken. Thus, the activity is detected as normal activity or an attack. If the activity is detected as normal, then

their request has been further processed otherwise their request has been blocked. So that the designed system shall provide the required system security and attain the desired goal.

4 Experiments and Result Analysis

4.1 Experimental Work Flow

The working and experimental workflow diagram is given in Fig. 3.

From the experimental workflow diagram, it has been observed that the input samples for IDS development are taken from KDD-99 dataset. The input samples are raw data, so there is a need for normalization. In this work, *minmax* approach is used for normalization. Then normalized input feature set is bifurcated into training set and testing set. Training set is used to train the classifiers and trained model is tested by the testing set. The performance parameters overall classification accuracy, individual class accuracy, and misclassification accuracy are used for the evaluation of the proposed system.

The extracted attributes are passed to classifiers. In this work, two classifiers support vector machine (SVM) and *k*-nearest neighbor (kNN) classifiers are used [27–29]. For SVM, different kernel functions are explored with the help of same

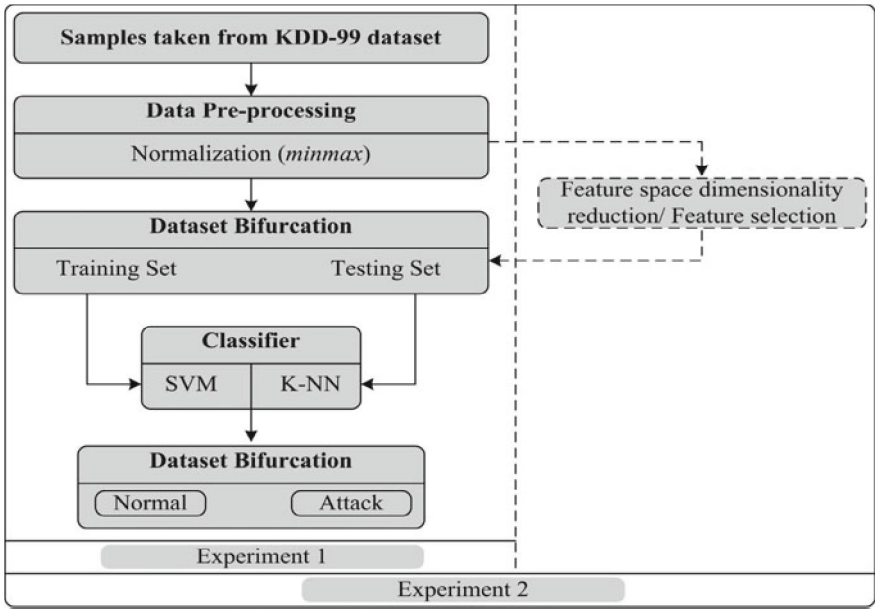

Fig. 3 Experimental workflow

Table 2 Brief description of experiments

Experiment No.	Description
Experiment No. 1	Development of an IDS using SVM classifier
	Development of an IDS using kNN classifier
Experiment No. 2	Development of an IDS using PCA-SVM classifier
	Development of an IDS using PCA-kNN classifier

training and testing dataset. Similarly, kNN classifiers have been explored with different distance measuring parameters. The performances of each case are reported in the results section.

It has been also noticed that some of the features are redundant and not relevant to the detection task. Therefore, principal component analysis (PCA) is used for dimensionality reduction [30]. In this work, the value of principal component (*pc*) is varied from 1 to 15 and the outcomes of each experiment are reported in the results section.

In this work, two exhaustive experiments have been performed. The brief description of the experiments is given in Table 2.

Experiment No. 1.1: In this experiment, a feature set of length 41 is used for development of an IDS using SVM classifier. For SVM classifier, different kernel functions are used in the experiments and obtained results are given in Table 3.
Experiment No. 1.2: In this experiment, a feature set of length 41 is used for the development of an IDS using kNN classifier. For kNN classifier, different distance measuring methods are evolved for the value of $k = 1, \ldots, 10$ in the experiment and obtained results are given in Table 4.

Table 3 Results of different kernel function for SVM classifier

Sr. no.	Kernel function	CM			Indiv. Acc. (%)	Accuracy (%)	Miss. Acc. (%)
			Attack	Normal			
1.	mlp	Attack	7512	2588	74.37	81.01	18.99
		Normal	510	5710	91.80		
2.	rbf	Attack	8049	2051	79.69	86.88	13.12
		Normal	90	6130	98.55		
3.	Linear	Attack	7742	2358	76.65	83.77	16.23
		Normal	290	5930	95.33		
4.	Quadratic	Attack	2598	7502	25.72	19.05	80.95
		Normal	5708	512	08.23		

Note mlp: multilayer perceptron, rbf: Gaussian Radial Basis Function, CM: confusion matrix, Indiv. Acc.: individual class classification accuracy, Miss. Acc.: misclassification accuracy

Table 4 Results of different distance methods using kNN classifier

Sr. No.	Value of k	Distance	CM			Indiv. Acc. (%)	Accuracy (%)	Miss. Acc. (%)
				Attack	Normal			
1.	$k = 1$	Cosine	Attack	8149	1951	80.68	87.24	12.76
			Normal	130	6090	97.90		
2.	$k = 1$	Euclidean	Attack	8049	2051	79.69	86.17	13.83
			Normal	115	6105	98.15		
3.	$k = 5$	Cityblock	Attack	7149	2951	70.78	79.89	20.11
			Normal	330	5890	94.69		
4.	$k = 1$	Correlation	Attack	7760	2340	76.80	83.75	16.25
			Normal	312	5908	95.33		

Experiment No. 2: In this experiment, PCA is used for dimensionality reduction. The reduced feature set is further bifurcated into training set and testing set. To get the optimum number of principle components, extensive work has been performed by varying the value of $p =$ '1, 2, …, 15'. Then the selected feature set is used for training the model. In this experiment, SVM with PCA (PCA-SVM) and kNN with PCA (PCA-kNN) are used to develop an IDS system.

Experiment No. 2.1: In this experiment, an IDS is designed using PCA-SVM classifier. The results of each experiment are given in Table 5.

Experiment No. 2.2: In this experiment, an IDS is designed using PCA-kNN classifier. The results of each experiment are given in Table 6.

Table 5 Results of different kernel function for PCA-SVM classifier

Sr. no.	Kernel function	p	CM			Indiv. Acc. (%)	Accuracy (%)	Miss. Acc. (%)
				Attack	Normal			
1.	mlp	15	Attack	8349	1751	82.66	88.71	11.29
			Normal	90	6130	98.55		
2.	rbf	5	Attack	8452	1648	83.68	**89.46**	10.54
			Normal	72	6148	98.84		
3.	Linear	9	Attack	8049	2051	79.69	85.41	14.59
			Normal	330	5890	94.69		
4.	Quadratic	5	Attack	8149	1951	80.68	87.24	12.76
			Normal	130	6090	97.90		

Bold indicates highest accuracy

Table 6 Results of different distance methods using kNN classifier

Sr. No.	Value of k	Distance	p	CM			Indiv. Acc. (%)	Accuracy (%)	Miss. Acc. (%)
					Attack	Normal			
1.	$k = 6$	Cosine	5	Attack	8540	1560	84.55	90.07	9.93
				Normal	60	6160	99.03		
2.	$k = 7$	Euclidean	15	Attack	8149	1951	80.68	87.24	12.76
				Normal	130	6090	97.90		
3.	$k = 3$	Cityblock	5	Attack	8100	2000	80.19	85.72	14.28
				Normal	330	5890	94.69		
4.	$k = 3$	Correlation	5	Attack	7149	2951	70.78	81.30	18.70
				Normal	100	6120	98.39		

4.2 Result Analysis

After the extensive experimentations, the following major observations have been pointed as:

1. There are two experiments that have been performed with the help of SVM and kNN classifiers. One experiment has been performed on the entire feature vector of length 41, and second experiment has been performed using PCA as feature vector dimensionality reduction and SVM, kNN as classifier so PCA-SVM and PCA-kNN are used.
2. From Table 3, it has been found that the maximum prediction rate or classification accuracy is 86.88% using SVM classifier with *rbf* function. The individual class classification accuracy for normal and attack is 98.55% and 79.69%, respectively. The same dataset has been tested on kNN classifier (results are reported in Table 4) and observed that the maximum prediction accuracy is 87.24% for the value of '$k = 1$'. In this exercise, the feature vector length is 41. Among these features, some features are redundant and not relevant for the classification task, so PCA is applied to reduce the dimensionality of feature vector. And the results are put in Tables 5 and 6.
3. From Table 5, it has been found that the highest prediction rate between normal and attacks is 89.46% for *rbf* kernel function using PCA-SVM classifier at the value of '$p = 5$'. The prediction rate for normal and attacks is 98.84% and 83.68%, respectively.
4. The same set of testing samples is passed through PCA-kNN classifier and obtained results are reported in Table 6. It has been observed that the maximum classification rate is 90.07% for '$p = 5$' and '$k = 6$'. The individual class classification accuracy is 84.55 and 99.03% for normal and attack detection, respectively.

4.3 Comparative Analysis

The developed IDS is compared with the study performed in [24]. It has been found that the accuracy reported for normal and attack discriminations is 83.28% but in the proposed system the achieved accuracy is 90.07%. The summary of results is given in Table 7.

The graphical representation of the obtained results is shown in Fig. 4.

Table 7 Comparative analysis of results obtained from different experiments

Sr. No.	Classifier	Accuracy	Nor_Acc.	Att_Acc.	Miss_Acc.
1	SVM	86.88	98.55	79.69	13.12
2	kNN	87.24	97.9	80.68	12.76
3	PCA-SVM	89.46	98.84	83.68	10.54
4	**PCA-kNN**	**90.07**	**99.03**	**84.55**	**9.93**

Note Nor_Acc.: individual class classification accuracy for normal, Att_Acc.: individual class classification accuracy for attack
Bold indicates highest accuracy

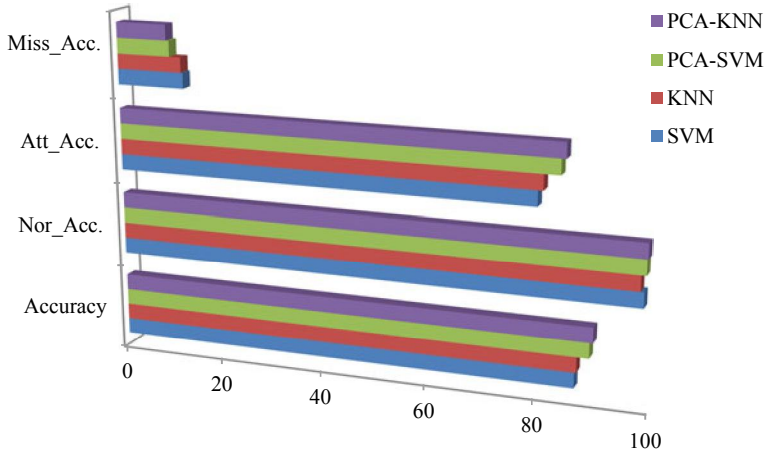

Fig. 4 Comparative analysis for outcome of tedious experiments

5 Conclusion

Intrusion detection system is a tool that monitors the network traffic and generates an alarm if any malicious activity occurred. The development of IDS is either signature-based or anomaly-based. The development of signature-based IDS is rule-based techniques, so this kind of tool can be also developed with the help of supervised machine learning based techniques. Thus, the authors have made an attempt to develop an IDS system using SVM and kNN classifiers. For this task, a benchmark dataset KDD-99 is used which comprised of 4,898,431 raw samples. Among the huge set of samples, a total set of 32,640 samples is considered. The used set is further bifurcated as training and testing set in balanced manner so that the training set consists of 16,320 samples and testing set consist of 16,320 samples.

After the experiments carried out for the development of IDS using supervised learning the obtained maximum accuracy for feature vector length of 41 is 87.24% with the help of kNN classifier. The PCA is used for the feature vector dimensionality

reduction and the obtained best-case accuracy is 90.07% using PCA-kNN classifier. Thus, it has been concluded that the PCA-kNN classifier performs better for IDS development. It is also worth mentioning that this work has been further extended to design IDS for a complete KDD-99 dataset using some advance machine learning techniques.

References

1. Endorf, C., Schultz, E., Mellander, J.: Intrusion Detection & Prevention. McGraw-Hill, Osborne Media (2004). ISBN: 0072229543
2. Anderson, J.P.: Computer security threat monitoring and surveillance. In: James, P. (eds) Technical Report. Anderson Company (1980)
3. Denning, D.E.: An intrusion-detection model. IEEE Trans. Softw. Eng. **2**, 222–232 (1987)
4. Verwoerd, T., Hunt, R.: Intrusion detection techniques and approaches. Comput. Commun. **25**(15), 1356–1365 (2002)
5. Khan, S., Loo, J., Din, U.Z.: Framework for intrusion detection in IEEE 802.11 wireless mesh networks. Int. Arab J. Inf. Technol. 7(4), 435–440 (2017)
6. Abraham, A., Grosan, C., Martin-Vide, C.: Evolutionary design of intrusion detection programs. IJ Netw. Secur. **4**(3), 328–339 (2007)
7. Tiwari, A., Ojha, S.K.: Design and analysis of intrusion detection system via neural Network, SVM, and neuro-fuzzy. In: Emerging Technologies in Data Mining and Information Security, pp. 49–63. Springer, Singapore (2019)
8. Abraham, A., Jain, R., Thomas, J., Han, S.Y.: D-SCIDS: distributed soft computing intrusion detection system. J. Netw. Comput. Appl. **30**(1), 81–98 (2007)
9. Roman, R., Zhou, J., Lopez, J.: Applying intrusion detection systems to wireless sensor networks. In: IEEE Consumer Communications & Networking Conference (CCNC 2006) (2006)
10. Karataş, F., Korkmaz, S.A.: Big data: controlling fraud by using machine learning libraries on spark. Int. J. Appl. Math. Electron. Comput. **6**(1), 1–5 (2018)
11. Peng, K., Leung, V.C., Huang, Q.: Clustering approach based on mini batch K-means for intrusion detection system over big data. IEEE Access (2018)
12. Anuar, N.B., Sallehudin, H., Gani, A., Zakaria, O.: Identifying false alarm for network intrusion detection system using hybrid data mining and decision tree. Malaysian J. Comput. Sci. **21**(2), 101–115 (2008)
13. Golovko, V., Kochurko, P.: Intrusion recognition using neural networks. In: 2005 IEEE Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, pp. 108–111. IEEE (2005)
14. Tian, S., Yu, J., Yin, C.: Anomaly detection using support vector machines. In: International Symposium on Neural Networks, pp. 592–597. Springer, Berlin (2004)
15. Chen, W.H., Hsu, S.H., Shen, H.P.: Application of SVM and ANN for intrusion detection. Comput. Oper. Res. **32**(10), 2617–2634 (2005)
16. Belouch, M., El Hadaj, S., Idhammad, M.: Performance evaluation of intrusion detection based on machine learning using Apache Spark. Proc. Comput. Sci. **1**(127), 1–6 (2018)
17. Li, Y., Parker, L.E.: Intruder detection using a wireless sensor network with an intelligent mobile robot response. In: IEEE Southeast Con 2008, pp. 37–42. IEEE
18. Peng, K., Leung, V., Zheng, L., Wang, S., Huang, C., Lin, T.: Intrusion detection system based on decision tree over big data in fog environment. Wirel. Commun. Mobile Comput (2018)
19. Ye, K.: Key feature recognition algorithm of network intrusion signal based on neural network and support vector machine. Symmetry **11**(3), 380 (2019)

20. Kalaivani, S., Vikram, A., Gopinath, G.: An effective swarm optimization based intrusion detection classifier system for cloud computing. In: 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), pp. 185–188. IEEE (2019)
21. Taher, K.A., Jisan, B.M., Rahman, M.M.: Network intrusion detection using supervised machine learning technique with feature selection. In: 2019 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), pp. 643–646. IEEE (2019)
22. Kim, J., Kim, J., Thu, H.L., Kim, H.: Long short term memory recurrent neural network classifier for intrusion detection. In: 2016 International Conference on Platform Technology and Service (PlatCon), pp. 1–5. IEEE (2016)
23. Al-Qatf, M., Lasheng, Y., Al-Habib, M., Al-Sabahi, K.: Deep learning approach combining sparse autoencoder with SVM for network intrusion detection. IEEE Access **12**(6), 52843–52856 (2018)
24. Yin, C., Zhu, Y., Fei, J., He, X.: A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access **12**(5), 21954–21961 (2017)
25. Bay, S.D., Kibler, D.F., Pazzani, M.J., Smyth, P.: The UCI KDD archive of large data sets for data mining research and experimentation. SIGKDD Explor. **2**(2), 81–85 (2000)
26. Cup, K.D.: Dataset, p. 72. Available at the following website http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html (1999)
27. Kumar, I., Virmani, J., Bhadauria, H.S., Panda, M.K.: Classification of breast density patterns using PNN, NFC, and SVM classifiers. In: Soft Computing Based Medical Image Analysis, pp. 223–243. Academic Press (2018)
28. Kumar, I., Bhadauria, H.S., Virmani, J.: Wavelet packet texture descriptors based four-class BIRADS breast tissue density classification. Proc. Comput. Sci. **1**(70), 76–84 (2015)
29. Kumar, I., Bhadauria, H.S., Virmani, J., Thakur, S.: A hybrid hierarchical framework for classification of breast density using digitized film screen mammograms. Multimedia Tools Appl. **76**(18), 18789–18813 (2017)
30. Kumar, I., Virmani, J., Bhadauria, H.S., Thakur, S.: A breast tissue characterization framework using PCA and weighted score fusion of neural network classifiers. Classification Tech. Med. Image Anal. Comput. Aided Diag. **12**, 129 (2019)