Int. j. inf. tecnol.
https://doi.org/10.1007/s41870-022-00925-w

ORIGINAL RESEARCH

A novel machine learning inspired algorithm to predict real-time network intrusions

Keshava Srinivas[1] · Narayanan Prasanth[1] · Rahul Trivedi[1] · Naman Bindra[1] ·
S. P. Raja[1]

Received: 9 November 2021 / Accepted: 2 April 2022

Abstract In today's digital world, most organizations are prone to cyberattacks. As a result, they face huge data and economic loss. Even under some circumstances, the organizations could lose their reputations and identity. A lot of research was conducted to address these cyber-attacks but still, it is a huge threat. Most of the algorithms address only the entry-level attacks and fail to replicate that performance to other attacks. Intrusion detection is one serious issue that can destabilize any kind of network. Especially, they are very difficult to contain in real-time systems. If these attacks are not detected in the early stages, they can create serious consequences for the network. The objective of the paper is to create a system that uses various machine learning algorithms to classify and predict network intrusions. Learning models are created and applied to large databases. Finally, these models were tested using various evaluation indicators for real-time data, and results are compared under various scenarios and use cases.

✉ Narayanan Prasanth
nnprcd@gmail.com

Keshava Srinivas
keshava.srinivas2017@vitstudent.ac.in

Rahul Trivedi
rahul.trivedu2016@vitstudent.ac.in

Naman Bindra
naman.bindra2017@vitstudent.ac.in

S. P. Raja
avemariaraja@gmail.com

[1] School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India

Keywords Intrusion detection · Deep neural networks · Adaboost · Random forest · Naïve Bayes · Logistic regression · Packet sniffer

1 Introduction

A computer or software program that monitors a network or system for malicious behavior or policy breaches is known as an intrusion detection system (IDS) [1]. Any breach is typically logged directly with the admin using the Security Information and Events Management (SIEM) [2] framework. The SIEM framework employs alert filtering algorithms to detect suspicious behavior from false alerts, including data from a range of sources. IDS forms vary from personal computers to large networks. The most common classifications are Network Intrusion Detection Systems (NIDS) and Host Intrusion Detection Systems (HIDS) [1]. A system that maintains track of critical operating system files is an example of HIDS, while a system that analyses incoming network traffic is an example of NIDS [3]. A detection strategy can also be used to identify IDS. Any IDS cargo is capable of responding to perceptible intrusions. An intrusion prevention system is a term used to describe a system that can respond. Custom methods, such as deploying honeypots to trace and describe hostile traffic, can be incorporated into intrusion detection systems to support specific functions.

As intrusion detection systems scan the network for potentially disruptive activities, false alarms can also occur. As a result, when organizations first install their IDS solutions, they need to fine-tune them [4]. It ensures that the intrusion detection system is properly configured to know what regular network traffic looks like versus malicious behavior. Intrusion detection mechanisms typically

Published online: 02 May 2022

oversee network packets accessing the device to look for associated suspicious activity and issue immediate warning alerts. IDS can also be categorized based on what kind of attack they are capable of predicting:

- A signature-based IDS—they successfully predict known attacks or attacks that closely resemble previously catastrophic attacks
- An anomaly-based IDS—they predict zero-day attacks, i.e., those attacks that are completely new and are not present in any previous database.

Machine learning is arguably one of the most powerful and powerful developments in the world today. More specifically, we still have a long way to go to reach our full potential. It is unlikely that he will continue to appear in the newspapers in the near future. Machine learning is a method of transforming information into knowledge. There is a lot of data from the last 50 years [5]. This volume of data is worthless until we study it and locate the models embedded in it. Machine learning methods are used to automatically identify useful underlying correlations in complex data that we would not otherwise detect. Habits and latent knowledge of the dilemma can be used to predict future outcomes and make all kinds of complex decisions. Many of us don't know that we are already working with machine learning every day.

Machine learning is a subfield of computing that seeks to enable machines to learn from data instead of explicitly programming it so that they can use PB-level data on today's Internet, but it still remains a complex task. Therefore, by using reliable classifier machine learning models, we can successfully detect any intrusion into our network and servers. There are various classification models like KNN, Decision Tree, Logistic Regression, support vector machine (SVM) [6], Deep Neural Network [7] that can be used for this purpose, each with its own pros and cons. The paper is categorized into following section: Sect. 2 discusses the related works and gaps identified, Sect. 3 discusses about the methodology used in this paper to detect intrusion, Sect. 4 discusses about the results and Sect. 5 concludes the paper.

2 Related work

Intrusion detection relies heavily on safety technology such as intelligent security devices, intrusion detection systems, intrusion prevention systems, and firewalls. Various intrusion detection systems are employed, but their effectiveness remains an issue. The effectiveness of intrusion detection is determined by its accuracy, which must be improved in order to reduce false alarms and increase the rate of detection. Recently, multilayer perceptron, support vector machine, and other techniques have been used to address performance issues. These approaches show drawbacks and aren't effective while use of large databases, for example data from the system and network. In the analysis of huge traffic data, IDS is made use of; hence, a robust classification technique is required to overcome the problem. In this article, this topic is taken into consideration. Well-known techniques of machine learning are applied, SVM [6], random forest, and extreme machine learning. Because of their classification ability, these approaches are well-known. The NSL-knowledge discovery and data mining dataset are used as a benchmark in the evaluation of intrusion detection systems. Extreme learning machine (ELM) [8] outperforms other approaches, according to the data.

In the last couple of decades, intrusion detection systems [1] have become significant. In IDS, many methods have been made use of, but machine learning (ML) algorithms are popular in recent literature. In addition, a variety of machine learning algorithms have been utilized, although some are better suited to analyzing large amounts of data for network and information system intrusion detection. Various machine learning algorithms, including SVM, random forest (RF), and ELM, are examined and compared to address this problem in this paper. In precision, accuracy, and memory, ELM outperforms other methods on complete data collections comprising 65,535 reports of events containing normal and invasive events. In comparison, in 1/2 of the data samples and in quarter of the data samples, the SVM showed better performance than other datasets. These results show that for a large dataset the ELM method acts excessively well when there are over 20 classifiers. However, SVM acts better for smaller datasets which again leads to the intuition that since the data is to be classified only as binary or not, logistic regression can be a better indicator for precision accuracy and recall [8].

In light of the long execution duration and poor execution performance of the Support Vector Machine in large-scale testing samples, the study proposed an online incremental and decremental learning approach based on a variable support vector machine (VSVM) [6]. Each sample consists of improved improvements in testing datasets to deeply understand the operational process and correlation algorithms for VSVM and the learning algorithm classifier requires to be modified. Next, the online growth quantity of the learning algorithm is entirely used by the incremental pre-calculated data and does not need retraining with the latest incremental testing datasets. Secondly, the inverse computation process of the incremental matrix substantially decreased the run time of the algorithm and is provided to validate the validity of the online learning algorithm. Finally, nine groups of datasets from the standard library were chosen for the pattern classification experiment. The results of the experiments show that the online learning algorithm used in the case ensures accurate classification rates and quick training. The online learning algorithm based on VSVM should address the issue of gradual process introduction, training meetings, and the necessity for large-scale data storage space, all of which contribute to slow training.

Chen et al. [9] proposed VSVM-based online exponential learning algorithm. The suggested algorithm has taken full use of incremental pre-calculated effects, does not need to re-learn the complete training datasets under the assumption, does not decrease the correct training rate and measure the correct rate, such that after calculating the inverse matrix, the reduced sum increases. In order to ensure the classification accuracy of the successful progress of training tempo, the experimental findings in the paper have shown that the algorithm increases with gradual performed training assembly, leading to sluggish training. The problem is solved using an online decremental learning algorithm based on the VSVM, which is defined by the inverse matrix decremental learning algorithm velocity by the inverse matrix. In conclusion, this research paper shows overwhelming results for this field of study. These results should not only be extolled and lauded but may also be used as a rudimentary aspect of further research [9].

Network security is becoming increasingly vital in our daily lives, not only for businesses but also for individuals. Multiple machine-learning algorithms have been advocated to boost the reliability of intrusion detection systems [1], which have been widely utilized to avoid compromising information. However, higher-quality data from preparation is a significant variable that could improve detection efficiency. It is well known that the most potent univariate classifier is the marginal density ratio. They suggested a powerful intrusion detection system [1] in this paper focused on a support vector machine (SVM) with enhanced functionality. They do this by transforming the logarithm marginal density ratios that compose the original characteristics into new and higher-quality altered characteristics that can considerably improve the detection capabilities of an SVM-based detection model. The NSL-KDD dataset is utilized to test the suggested methodology, and the analytical findings show that it delivers higher and more stable efficiency than current approaches in terms of accuracy, identification rate, false alarm rate, and training duration.

Many artificial intelligence (AI) [10] algorithms have been added to IDS's in order to boost the efficiency of intrusion detection. SVM is used of these techniques and have reasonably better performance. In addition, the efficiency of intrusion detection [1] is strongly dependent on the accuracy of data from preparation. They suggest an efficient IDS based on SVM with function augmentation in this article. This is used in this detection system to provide the SVM classifier with succinct, high-quality training data, which not only enhances the SVM's detection capability, but also decreases the training time needed. The results of the analytical experiment show that their suggested detection system can produce a stable result with high accuracy and a high detection rate, as well as a low false alarm rate and a rapid training speed. Their proposed approach has strong benefits and is highly efficient as opposed to other recently proposed solutions for intrusion detection problems. The various experimental and comparative studies shown in this article have confirmed the feasibility of their suggested structure. They found, however, just the binary case of intrusion detection problems in this article. Therefore, they intend to generalize our analysis to include distinct forms of attacks in future work [11].

A unique SVM model integrating kernel principal component analysis (KPCA) [12], and genetic algorithm (GA) is developed for intrusion detection. A multi-layer SVM classifier is utilized in the proposed model to predict if the behaviour is an assault, and KPCA is utilized as an SVM preprocessor to minimize the dimension of feature vectors and shorten training time. To reduce the noise created by feature differences and increase SVM performance, an improved kernel function named N-radial basis function (RBF) is proposed by integrating the mean value and mean square difference values of feature attributes into the RBF kernel function. To maximize the penalty factor C, kernel parameters σ and the tube scale σ of SVM, GA is used. The experimental findings indicate that the proposed model achieves higher predictive precision, greater convergence speed and improved generalization compared to other detection algorithms.

A unique hybrid KPCA SVM with GAs model is provided in this paper for intrusion detection. In the N-KPCA-GA-SVM model, KPCA is utilized to capture critical aspects of intrusion detection data, and a multi-layer SVM classifier is utilized to determine whether or not the attack is an attack. The N-RBF kernel function is based on the Gaussian kernel function and is programmed to reduce training time and improve the efficiency of the SVM classification model. GA is used to select appropriate parameters for the SVM classifier [6], which prevents the SVM model from over-fitting or under-fitting due to incorrect parameter determination. The experimental results show that the proposed KPCA SVM model has better classification accuracies than randomly selected SVM classifiers with selected parameters, and that KPCA can achieve better generalization efficiency with the SVM classifier by feature extraction using principal component analysis (PCA) [12] than without feature extraction. Furthermore, the results of the experiments show that KPCA outperforms PCA in terms of intrusion detection. The reason for this is that KPCA can pursue higher order data from the initial inputs than PCA. By adopting the kernel methodology to generalize PCA to nonlinear, KPCA also takes into account higher order knowledge of the initial inputs. Additional principal components can be retrieved using KPCA, perhaps improving generalization findings. For future study, they plan to develop further algorithms that integrate kernel approaches with other classification methods for pattern analysis and online intrusion detection, as well as investigate additional optimization approaches for SVM parameters [13].

In machine learning, an ensemble classifier, which is a combination of classifiers, typically beats solo ones. Despite the fact that various ensemble approaches exist, determining the best ensemble configuration for a given dataset remains a difficult task. This research introduces a novel ensemble construction method that leverages particle swarm optimization (PSO) [14] weights to produce a set of classifiers with higher precision for intrusion detection. The local unimodal sampling (LUS) [14] technique is utilized as a meta-optimizer to obtain better PSO behavioral parameters. For our analytical research, we selected five random subsets from the well-known KDD99 dataset. Ensemble classifiers are created using the new techniques as well as the weighted majority algorithm (WMA) [14] strategy. Their findings suggest that the novel strategy can create sets that beat WMA in terms of classification precision. The purpose of this study is to develop ensemble-based classifiers that will improve the accuracy of intrusion detection. For this reason, they trained and tested 12 professionals before putting them together into an ensemble. We employed the PSO algorithm to weight each expert's opinion. Because the quality of the behavioral factors introduced by the customer into PSO strongly determines its efficacy, we used the LUS technique as a meta-optimizer for selecting high-quality parameters. We then used the improved PSO to create new weights for each specialty. For comparison, we developed an ensemble classifier with weights generated via WMA.

For consistency, the machine framework was divided into the following seven steps.

1. Kdd99 data is being pre-processed.
2. Data classification using six distinct SVM experts.
3. Data classification using k-NN with six different experts.
4. PSO-based data classification with ensemble classifier
5. Data classification with ensemble classifier centered on LUS augmentation of PSO.
6. WMA-based data classification with ensemble classifier.
7. Each strategy's results are compared.

The best outcomes we achieved were with the PSO. We have an overall accuracy gain of 0.756 percent relative to the highest base specialist's accuracy. They say that if they consider the scale of the entire test data, i.e. 311,029 observations, than for any 2351 observations, they could predict better classification results [14].

With the advancement of technology, intrusion detection has become a growing subject of research. The Intrusion Detection System (IDS) tries to distinguish between regular and abnormal user behavior and alerts them. IDS is a nonlinear and dynamic subject that deals with network traffic data. Several IDS methods have been proposed, with differing degrees of accuracy. This is why it's critical to have a reliable and accurate intrusion detection system. In this paper, they used a random forest classifier to create a model for an intrusion detection system. Random forest (RF) is an ensemble classifier that performs better than other conventional classifiers at detecting attacks. They conducted experiments on the NSL-KDD data set to assess the working of their model. Empirical findings suggest that the proposed model is accurate with low false performance. Feature selection is introduced to the data collection to reduce dimensionality and eliminate duplicated and irrelevant functions. They applied symmetrical attribute ambiguity that overcomes knowledge benefit issues. The recommended solution is tested using the data collection of the NSL KDD. In terms of accuracy and Matthews correlation coefficient (MCC) [15], they compared random forest modelling with the j48 classier. Their experimental outcome shows that their suggested approach improves the precision and MCC for four forms of attacks. In order to further enhance the performance of the classifier, they propose using evolutionary computation as a feature selection measure for potential work. Overall, the researchers in this paper showed accurate and meticulous results with unmatched potential for growth that can be accelerated to next levels with evolving technology. These results can be the basis for many other derivative studies regarding intrusion observation and exploration [15].

Most modern IDSs are rules-based structures that are very difficult to encode rules and are unable to detect new intrusions. A hybrid detection framework is therefore proposed, which depends on the classification and clustering techniques of data mining. In abuse detection, the Random Forest classification method is used to automatically create intrusion patterns from a training dataset and then compare network connections to these intrusion patterns to detect network intrusions. The k-means clustering algorithm is used in anomaly detection to discover novel intrusions by clustering data from network links into one or more clusters. The anomaly section of the proposed hybrid system is reinforced by substituting a weighted k-means algorithm for the k-means algorithm, and it also employs a recommended strategy for selecting anomalous clusters by injecting proven assaults into unknown relationships. Their tactics are tested over the Knowledge Discovery and Data Mining (KDD'99) datasets.

The data-mining-based network intrusion detection systems are discussed in this article. Two data-mining methodologies are utilized in misuse, anomaly, and hybrid detection. To begin, the random forest algorithm is used as a data mining classification algorithm in a method of misuse detection to create intrusion patterns from a balanced testing dataset and to identify the captured network connexons as a result of the key types of intrusions using the constructed patterns. Their method is introduced in C#.NET using the original random forest implementation and evaluated by the KDD. The key drawback of the approach of detecting misuse is that it does not identify new intrusions for which it has not been educated before. Second, the k-means algorithm is used as a data-mining clustering technique to split the collected network connections into a predetermined number of clusters in an unsupervised anomaly detection system, and then discover the anomalous clusters based on their features. The "KMlocal" implementation of the k-means clustering technique is used to create our anomaly detection technique. After resolving concerns with category and multiple scale functionality, we put our method to the test on the KDD'99 datasets. The high false positive rate is the fundamental disadvantage of the anomaly detection methodology. Third, the random forest method is utilized in conjunction with the wkmeans method to create a hybrid system that addresses the shortcomings of both misuse and anomaly detection. The random forest algorithm is used to boost the detection rate of the anomaly detection component in the misuse detection component by using function significance values produced by the random forest algorithm. A supervised strategy is proposed to improve the detection of anomalous clusters by introducing known attacks into unknown data before clustering and evaluating the anomalous clusters using these known intrusions. On the KDD'99 datasets, our experiment is assessed. The findings suggest that identification rates and false positive rates are best accomplished by the hybrid system than the techniques [16].

Disruption can be a big challenge for computer and network networks, as one entry can cause large losses over a few seconds to avoid the intrusion, and a strong control mechanism is necessary. Existing access methods are effective, however, there are a lot of false alarms. The use of data like retrieval to tackle this issue, the most recent PCA method of subset selection in which the elements are first translated to their own space and the variables are selected independently, is one of the reasons for false alarms (e.g. segmentation. the selection of slow-moving genes). The PSO-based method was therefore suggested in the selection of the fabric used for this scientific research. Today, the attackers are creating polymeric malware to hack the device. The specialty of polymeric malware is that malware is able to constantly adjust its identifiable function to fool the detection technique and uses a signature-based process. This paper would establish a system that is capable of acquiring behavioral patterns that will conduct a static and dynamic analysis and a dissimilar machine learning (ML) [17] technique to detect the existence of malware. Various forms of malware are Adware, Spyware, Bug, Worm, Trojan, Rootkit, Backdoors, Key loggers, Ransomware, and Hijacker Browser. There are two types of malware discovery investigation procedures. These are the Static Approach and the Exploration Approach. The static approach used to examine the malware relies on the precharacterized label. The hierarchical framework is used to evaluate malware based on transition based on time and approach. The approach process, SVM, is used to recognize malware and class [18].

2.1 Gaps identified

In the existing work [8], authors aimed to compare 4 models for intrusion detection namely, SVM Linear, SVM RBF, random forest and extreme learning machine [19]. It was found that for full samples in the NSL-KDD dataset, extreme learning machine [19] is ahead in aspects and metrics compared to the others. Whereas, for half the dataset, SVM Linear is ahead. Despite attaining immaculate results in terms of the evaluation metrics, certain loopholes are found within the research. Some are:

- Instead of using the separate Test dataset within the NSL-KDD template, they have gone for a train-test split approach on the train dataset. This is concerning because, the train dataset is largely biased towards the benign values. Whereas, the test dataset is equal in terms of the normal data packets and the intrusion data packets. Hence this leads to an unrealistic premise and thus a biased conclusion.
- They have neglected the type and service attributes since they are categorical variables. However, these attributes reveal a great amount of information regarding the nature of the packet and thus will be of great help in the classification process.
- Since this is a binary classification problem it is evident that SVM Linear would outperform SVM RBF [20]. However, SVM in general would take huge amounts of time to classify since creating a hyper plane would be extremely time intensive.

3 Methodology used

Looking at the last section and analyzing its gaps, we have decided to adopt the following variations in our model:

- Since SVM performs the best for smaller datasets and fails for larger ones because of its Gaussian function which leads to degradation. We propose the use of logistic regression for this since its results are not that affected by increase in training data. Also performance will certainly increase as logistic regression is much faster.
- This is also a good approach as we can see from the results that SVM (linear) preforms better than SVM (RBS). Therefore, logistic regression can be used as a place for SVM (linear), so one can't counter that logistic performs worse for nonlinear data, because the dataset is mostly linear as confirmed by SVM (linear).
- The service and protocol type attributes were neglected in the base paper. We would be keeping these attributes while still maintaining limited time complexity.

The machine learning part of the implementation makes up the chunk of the work and thus is broken into the following five as shown in Fig. 1.

3.1 Data extraction

(a) We have used analysed different data sets available for modelling network intrusion detection system

(b) These datasets are based on the literature survey that we have conducted. The various different papers have used wide range of datasets from various publications such as kaggle, google public datasets, UCI machine learning library and government sites.

(c) The different datasets were analysed based on the performance, robustness, quality, quantity, availability and volatility.

(d) We have selected NSL-KDD dataset due to the about mentioned factors.

3.2 Data preprocessing

(a) The next step includes data preprocessing, importing libraries that support implementation of that language, importing data sets, discovering missing data, coding of categorical data, separation of training data set and test set, scaling of functionality.

(b) Data preprocessing—map attack field to attack class

- NSL-KDD dataset consists of 42 attributes for every connection record comprising class label containing attack types. These attack types are categorized into four attack classes as described by
- Denial of Service (DoS): It is an attack in which the attacker directs a large number of traffic requests to the system to make the computing or memory resources too busy or too full to process legitimate requests and in the process deny legitimate users access to the machine.
- Probing attack (Probe): Poll computer networks to collect information used to disrupt the security controls.
- User to root attack (U2R): One type of vulnerability, the attacker first accesses the common user account on the system (obtained by crawling passwords, dictionary attacks, or social engineering) and can use certain vulnerabilities to gain root access to the system.
- Remote to local attack (R2L): When an attacker can send data packets to a machine through the network but does not have an account on the machine, certain vulnerabilities will be exploited to gain local access as the user of the machine.

(c) Convert categorical values into numerical columns using label encoder and one hot encoding techniques.

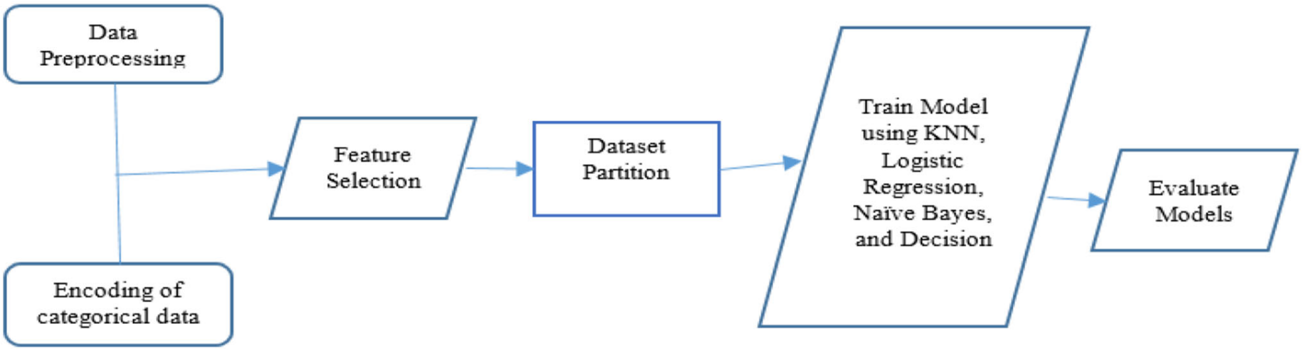
Data Preprocessing
Encoding of categorical data
Feature Selection
Dataset Partition
Train Model using KNN, Logistic Regression, Naïve Bayes, and Decision
Evaluate Models

Fig. 1 Flow diagram of the proposed approach

3.3 Feature extraction

(a) When we select a dataset, often times it comes with a lot of information that may or may not be required in the process or algorithm that we are use. Hence we need to choose the features that are needed by us and it is called as feature extraction.
(b) Drop attack attribute from test data and train data.
(c) Perform data analysis. Check quality of data—mean, count, standard deviation, min, max, 25%, 75%, 50% etc.
(d) 'num_outbound_cmds' attribute consists of all the zero values, which is then removed due to the redundancy.
(e) Attack class distribution in test and train dataset.
(f) Remove flag variable as previous research shows that the variable amounts to little to no results.

3.4 Creating an intrusion detection system (IDS) to detect malicious activities

(a) Finalize data preprocessing for training.
(b) Train models on preprocessed data using scikitLearn.

1. KNN
2. Gaussian Naïve bayes
3. Decision trees
4. Logistic regression
5. Adaboost
6. Random forest
7. Deep neural network

(c) Create a packet sniffer that successfully de-encapsulates packets into their respective layer 2(frames), layer 3(packets), and layer 4 (segment) headers and payloads. Data is then extracted from this in the form of the NSL KDD datasets. Some attributes couldn't be extracted due to nature of attribute and packet, and thus a mean of the train data was placed in that case. From this, the test data was created and successful and real time predictions were made.

3.5 Analysis and testing

(a) Finally the evaluation of the model on test dataset with four major different methods as proposed.
(b) Cross validation mean Score—one of the most widely used technique to rate the models based on training the datasets on subsets of the available input data
(c) Model accuracy—accuracy is the percent/fraction of the cases the model got right
(d) Accuracy = number of correct predictions upon the Total number of predictions
(e) Confusion matrix—one of the most common criteria used to judge models
(f) A confusion matrix gives you an idea about how your machine classifier has performed.
(g) Classification report—to measure the quality of the predictions
(h) The report reports the main classification metrics of precision, recall and f1-score on each and every class.
(i) The metrics are judged by the true and false positives and negatives.

To detect the intrusion detection in real-time systems we used seven different kinds of machine learning algorithms. These algorithms are implemented to predict the intrusion and their results are compared for their performance. The algorithms used are:

1. K Nearest neighbour (KNN)
2. Gaussian Naïve Bayes
3. Decision trees
4. Logistic Regression
5. Adaboost
6. Random Forest
7. Deep Neural Networks

4 Results and discussion

NSL-KDD dataset has 42 properties for every association record including class mark containing attack types. The assault types are arranged into four assault classes as shown in Table 1, i.e. DoS, Probe, U2R and R2L. There are 4 types of feature sets in the dataset:

- Categorical attributes (2, 3, 4, 42).
- Binary attributes (7, 12, 14, 20, 21, 22).
- Discrete attributes (8, 9, 15, 23–41, 43).
- Continuous attributes (1, 5, 6, 10, 11, 13, 16, 17, 18, 19).

Each attribute gives some information to the model, thus helping in training the respective models. However, as shown in Fig. 2, some had a higher level of importance compared to others. Source bytes and destination bytes proved to be the highest in terms of importance.

Table 2 shows the various algorithms and their respective evaluation metrics which can be easily compared and chosen while predicting real-time network intrusions. Each evaluation metric provides a certain understanding on which algorithm to use in which exact scenario. In regards to the accuracy of the predictions, the Deep Neural

Table 1 KDD-NSL attack types

Classes	DoS	Probe	U2R	R2L
Sub classes	Apache2 Back Land Neptune Mailbomb Pod Processtable Smurf Teardrop Udpstorm worm	Ipsweep Mscan Nmap Portsweep Saint satan	Buffer_ouverflow Loadmodule Peri Ps Rootkit Sqlattack xterm	ftp_write guess_passwd httptunnel imap multihop named phf sendmail snmpgetattack spy snmpguess warezclient warezmaster xlock xsnoop

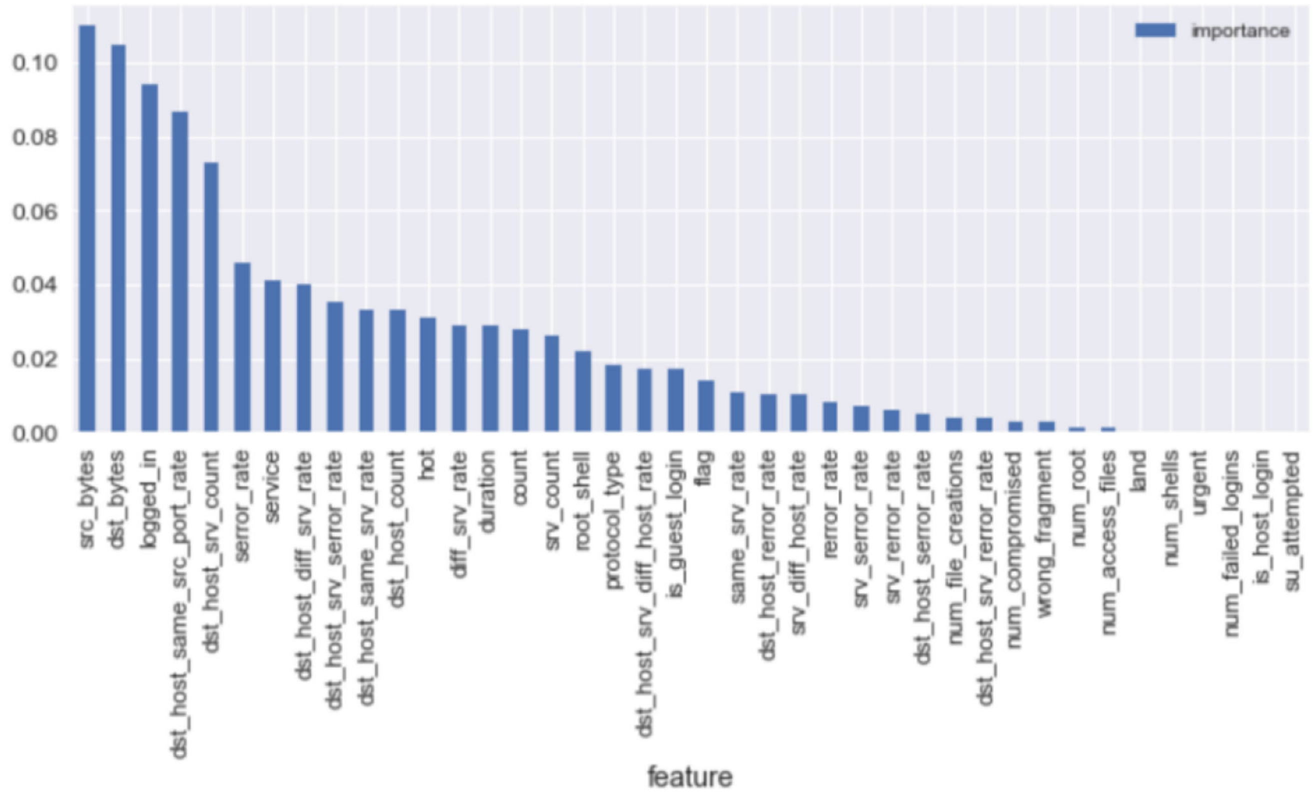

Fig. 2 Features and their importance

Network (DNN) model trumps the other models; however, is more time consuming than the others. Decision Trees are ahead of the other models in terms of the F1 score. The precision of DNN Layer 4 is the highest at 99.9, this almost makes the number of false positives nil, and hence can be used when the user doesn't want any packet to be incorrectly flagged as positive, i.e., when the user values the importance of any non-intrusion packet. In terms of recall, Naïve Bayes leads at 92.3 percentage and thus, the user can choose this model when the user cannot afford any intrusion packet that is when, the system is extremely sensitive and can't have any intrusions disrupting it. When it comes to lesser time complexity—logistic regression and KNN have the lead, whereas SVM has the highest with almost

Table 2 Evaluation metrics of various algorithms during intrusion detection process

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
Logistic regression	84.62	98.84	81.85	89.55
Naïve Bayes	92.94	98.83	92.32	95.47
Random forest	92.64	99.87	90.98	95.21
Decision tree	93.08	99.85	91.54	95.52
SVM	81.1	99.4	77.0	86.8
KNN	87.35	97.34	89.02	92.57
Adaboost	91.31	98.04	90.22	93.19
DNN layer 1	92.9	99.8	91.4	95.4
DNN layer 2	92.9	99.8	91.4	95.4
DNN layer 3	93.2	99.7	91.5	95.5
DNN layer 4	92.9	99.9	91.3	95.4
DNN layer 5	92.7	99.8	91.1	95.3

Table 3 Real time intrusion detection using Adaboost

IP Src	IP Dst	Ethernet Src	Ethernet Dst	Prediction
193.66.98.101	97.99.111.10	b'400040110f6b'	b'450000466e0f'	0
193.66.98.101	97.99.111.10	b'400040110f6b'	b'450000466e0f'	0
193.66.98.101	97.99.111.10	b'400040110f6b'	b'450000466e0f'	0
193.66.98.101	97.99.111.10	b'400040110e83'	b'450000466ef7'	0
193.66.98.101	97.99.111.10	b'400040110e83'	b'450000466ef7'	0
193.66.98.101	97.99.111.10	b'400040110e83'	b'450000466ef7'	0
193.66.98.101	97.99.111.10	b'400040110ce1'	b'450,000,466,099'	0
193.66.98.101	97.99.111.10	b'400040110ce1'	b'450,000,466,099'	0
193.66.98.101	97.99.111.10	b'400040110ce1'	b'450,000,466,099'	0

12 h to train and test the model making it impossible for real time detection.

Table 3 shows real time packet sniffing and intrusion detection being done using the Adaboost model. The ip and mac addresses are also displayed to provide more input to the user.

5 Conclusion

Daily activities of people and the proceedings of organizations are heavily dependent on how secure the internet network around them is. Data loss either personal or professional can have serious consequences on the life people. Securing sensitive information and hardwiring security is of the utmost importance, and this need isn't going to disappear in the near future where everything will shift to the cloud. Due to this need, Intrusion Detection systems have been given importance. However, the IDS available are either too computationally expensive for the end-user or not accurate enough in actively predicting real-time frames within the network. In this respect, our research has been able to overcome these shortcomings. With over 8 machine learning algorithms to choose from, each computationally feasible and accurate, the end-user has a variety of options to choose from depending on the context and needs at the moment. The evaluation metrics of each algorithm such as recall, precision and F1 score has been generated; thus, making it easy to decide which algorithm to go for while detecting real-time intrusions. This fulfils a use case which is actively been neglected for end user systems. As a future work, the algorithms can be tested under the constrained environment like Internet of Things. In these networks, the number of things gets connected increases exponentially and therefore they are more susceptible to attack by the intruders. So this could be very challenging and interesting work.

References

1. Liao H-J, Lin C-HR, Lin Y-C, Tung K-Y (2013) 'Intrusion detection system: a comprehensive review.' J Netw Comput Appl 36(1):16–24
2. Kotenko I, Chechulin A (2012) Common framework for attack modeling and security evaluation in SIEM systems. In: Proc. of 2012 IEEE International Conference on Green Computing and Communications, Conference on Internet of Things, and Conference on Cyber, Physical and Social Computing. Los Alamitos, California. IEEE Computer Society, 2012, pp 94–101

3. Kuha J, Mills C (2018) On group comparisons with logistic regression models. Sociol. Methods Res. 49(2):1–28. https://doi.org/10.1177/0049124117747306
4. Ahmad I (2015) Feature selection using particle swarm optimization in intrusion detection. Int J Distrib Sens Netw 2015:1–8. https://doi.org/10.1155/2015/806954
5. Aziz AA, Hanafi SE, Hassanien AE (2017) Comparison of classification techniques applied for network intrusion detection and classification. J Appl Logic 24(Part A):109–118. https://doi.org/10.1016/j.jal.2016.11.018
6. Pisner DA, Schnyer DM (2020) Support vector machine. In: Mechelli A, Vieira SBT-ML (eds) Chapter 6. Academic Press, Cambridge, pp 101–121, ISBN 978-0-12-815739-8
7. Miikkulainen R, Liang J, Meyerson E, Rawal A, Fink D, Francon O, Raju B, Navruzyan A, Duffy N, Hodjat B (2017) Evolving deep neural networks. arXiv preprint. https://arxiv.org/abs/1703.00548
8. Ahmad I et al (2018) Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection. IEEE Access 6:33789–33795
9. Chen Y et al (2019) A novel online incremental and decremental learning algorithm based on variable support vector machine. Clust Comput 22(3):7435–7445
10. Russell SJ, Norvig P (1995) Artificial intelligence: a modern approach. Prentice Hall, Englewood Cliffs
11. Jayakumar Kaliappan, Lokesh Kumar R, Thanapal P, Narayanan Prasanth, Luo Xianlu (2020) Network attack detection using Weighted Dempster-Shafer evidence theory. Int J Adv Sci Technol 29(5):3710–3720.
12. Abdi H, Williams LJ (2010) Principal component analysis. Wiley Interdiscip Rev Comput Stat 2:433–459
13. Kuang F, Weihong X, Zhang S (2014) A novel hybrid KPCA and SVM with GA model for intrusion detection. Appl Soft Comput 18:178–184. https://doi.org/10.1016/j.asoc.2014.01.028 (**ISSN 1568-4946**)
14. Aburomman AA, Reaz MB (2016) A novel SVM-KNN-PSO ensemble method for intrusion detection system. Appl Soft Comput 38:360–372. https://doi.org/10.1016/j.asoc.2015.10.011 (**ISSN 1568-4946**)
15. Teng S, Wu N, Zhu H, Teng L, Zhang W (2018) SVM-DT-based adaptive and collaborative intrusion detection. IEEE/CAA J Autom Sin 5(1):108–118. https://doi.org/10.1109/JAS.2017.7510730
16. Farnaaz N, Jabbar MA (2016) Random forest modeling for network intrusion detection system. Procedia Comput Sci 89:213–217. https://doi.org/10.1016/j.procs.2016.06.047 (**ISSN 1877-0509**)
17. Jordan MI, Mitchell T (2015) Machine learning: Trends, perspectives, and prospects. Science 349(6245):255–260
18. Ahmad, Amin FE (2014) Towards feature subset selection in intrusion detection. In: 2014 IEEE 7th Joint International Information Technology and Artificial Intelligence Conference, Chongqing, 2014, pp. 68–73, https://doi.org/10.1109/ITAIC.2014.7065007
19. Huang G-B, Zhu Q-Y, Siew C-K (2006) Extreme learning machine: theory and applications. Neurocomputing 70(1–3):489–501
20. Chithik R, Mohamed S, Munir M, Rabbani A (2017) Combined analysis of support vector machine and principle component analysis for IDS. https://doi.org/10.1109/CESYS.2016.7889868