TASR Trends in Applied Sciences Research

OPEN ACCESS

A Proposed Intrusion Detection System Based on an Improved Random Forest Using a Double Feature Selection Method

[1]Zaed Mahdi and [2]Negar Majma

[1]Department of Computer Engineering, Islamic Azad University, Isfahan, Iran

[2]Department of Computer Engineering, Naghshejahan Higher Education Institute, Isfahan, Iran

ABSTRACT

Cyber-attacks today are a source of great concern due to the increase in the use of the Internet in many areas, which has allowed increasing intrusion on networks and attempts to damage systems and others. Therefore, to stay up with the evolution of cyber-attacks, intrusion detection systems must be constantly improved. Intrusion detection system is a technique that may be applied to track both known and unidentified breaches before one of them damages network hardware. One of the very important things that has a big role in the strength of the system is the selection of good features in training the system. In this research paper, intrusion detection systems are proposed based on reducing and selecting features through the use of a "double feature selection" with the random forest algorithm. Experiments were performed on a data set NSL-KDD (it dataset from the Canadian Institute for Cybersecurity). By evaluating the performance, a system accuracy of 0.9981, a training time of 3.47 sec and a detection time of 0.24 sec were obtained. The proposed work was compared with related work using the same algorithm and dataset. The system proved superior to many of the proposed systems in terms of accuracy of the system, recall, precision, the time spent in training the system and the time of detection.

KEYWORDS

Intrusion detection, random forest, double features selection, cyber-attacks, select K best

INTRODUCTION

Hacking and cyber-attacks have substantially increased in frequency as a result of the development of computer systems connected to the Internet. Therefore, network security, cyber security and intrusion detection systems are crucial. A cyber-attack is a harmful, malevolent and intentional attempt by an individual or group to gain access to data in a person's or organization's information system. The cyber security is now essential for defending the network against these threats. An example cyber-attacks (Denial-of-Service (DoS) attacks, SQL injection attack, Remote user (R2L) attacks, User to Root (U2R) attacks, Probe attacks and other attacks)[1]. An Intrusion Detection System (IDS) is a type of security technology that may stop illegal connections and thwart outside threats. On the Internet, an Intrusion Detection System (IDS) can provide confidentiality, integrity and usability[2]. Particularly beneficial has been machining learning-based IDS particularly because machine learning enhances intrusion detection

ISSN: 2151-7908 (Online)
ISSN: 1819-3579 (Print)
https://doi.org/10.3923/tasr.2024.51.60

Received: 27 Mar. 2023
Accepted: 26 Dec. 2023
Published: 06 Jan. 2024

methodology[3]. Data mining-based IDS locate user data and predict future results. Knowledge discovery, or DM, made possible by databases has drawn increased interest from the general public and the information technology sector[4]. Despite the large number of studies regarding intrusion detection systems and the development of these systems, there are some defects and among these defects is the training of the system. It is very necessary to choose the appropriate data set for training, as well as the method of data processing and to choose only the appropriate features for training, which helps in good training and short training time.

In this paper, a method for intrusion detection system based on machine learning was developed based on the selection of appropriate data sets and the selection of features with the highest variance after data preprocessing. Two phases will be used to select the features. The first stage is to eliminate the least important features by the variance threshold algorithm and the second stage is to select the features with the highest score by (select k best algorithm and fclassif method). Which helps to reduce training time and increase the accuracy of the proposed system. The proposed model will be built by the random forest algorithm because of its robustness on a large scale. In this paper, the NSL-KDD dataset, an upgraded version of the well-known KDD dataset, was employed. The proposed approach produced much greater accuracy than current systems while requiring significantly less training time.

RELATED WORKS

A machine learning-based approach for cyber security intrusion detection was proposed by Sarker *et al.*[5] (IntruDTree). It uses a ranking system based on the importance of security elements. The process included examining the security dataset, producing raw data, determining the value of each characteristic and ranking. The techniques made use of a feature selection is basic Bayes classifier. The model was tested in experiments utilizing cyber security datasets (NSL-KDD) and the resulting tree was produced.

Farnaaz and Jabbar[6] proposed an intrusion detection model based on the random forest algorithm. They categorized the attacks into four categories and applied 10 cross-checks to the classification. Excess features are also removed and relevant features are selected by filter method, wrapper method and embedded method. The performance of the model was evaluated using the NSL-KDD dataset. The authors obtained a system accuracy of 0.9967.

Kayode-Ajala[7], proposed various machine learning techniques for identifying network traffic anomalous. They employing these techniques on the NSL-KDD dataset and used Principal Component Analysis (PCA) for reduced dataset features to 20 principal components. The employed for evaluation was training and test accuracy, precision and recall. Logistic regression competitive results with a training accuracy of 86.97% and a test accuracy of 86.62%. The K-neighbor classifier competitive results with a training accuracy of 98.05% and a test accuracy of 97.94%. Their findings suggest the best algorithms for intrusion detection tasks is with K-neighbors classifier standing out as the most robust performer.

Gao *et al.*[8] concentrated their efforts on developing novel intrusion detection methods employing and PCA to choose feature selection, one of which is the application of machine learning techniques. Five voting classifiers are selected through comparative tests. Then, by adjusting the amount of samples, setting data weights, multi-layer detection and other combination techniques, the detection impact of each algorithm is enhanced. Finally, an adaptive voting mechanism using several class-weights is used to obtain the best detection results. The NSL-KDD preprocessing model was trained. To validate the technique, all of the algorithms are trained using cross validation with training data. The training accuracy of any algorithm depends on that algorithm's training accuracy. Decision tree, random forest, K-Nearest Neighbors (KNN) and Deep Neural Networks (DNN) all have assessment accuracy values of 99.63, 99.8, 99.59 and 98.4%, respectively.

Negandhi *et al.*[9] proposed random forests, a supervised learning approach, was employed for the maximum accuracy in order to identify and prevent network assaults. In order to limit the number of features, the traits chosen for the model based on Gini significance were used. It was proposed that the NSL-KDD dataset be used to train a model to categorize all of the assaults in the dataset; the inaccuracy of the result is 0.120% and the correctness of the result is 99.880%.

Using a recursive feature elimination to select features and use Deep Neural Networks (DNN) and Recurrent Neural Network (RNN) for classification suggested by Mohammed and Gbashi[10]. The accuracy rate is 94% and DNN was used for binary classification to classify attack or normal.

METHODOLOGY

Intrusion Detection System (IDS): A proactive intrusion technological detection called an Intrusion Detection System (IDS) finds and categorizes intrusions. Based on intrusive behavior, cyber-attacks and policy breaches at the network and host level infrastructure occur in real time[11]. It is a technique that may be applied to track both known and unidentified breaches before one of them damages network hardware. It is a crucial foundation for authentication used to safeguard the availability, confidentiality and integrity of data. An illustration of the intrusion detection model was shown in Fig. 1[12].

Feature selection: The feature selection method is used to identify and delete as much redundant and pointless information as is practical. This technique for dimensionality reduction may speed up and improve the performance of learning algorithms, improving classification accuracy[13].

Variance threshold technique: It is a basic technique for filtering features. Every feature whose variance falls below a certain threshold is eliminated. The zero variance characteristics are removed. This refers to characteristics whose default value is the same for all samples. It is assumed that features with a larger variation may contain more significant information, even if it should be highlighted that we are not taking into account the relationship between feature variables or the relationship between feature and objective variables. Low variance columns should probably be removed as they could divert some learning algorithms (particularly ones based on distance)[14]:

$$\text{Variance score (fi)} = p\ (1-p)^{14}$$

Where, p is the proportion of those that chose the feature value of 1.

Select K best method: This kind of univariate is used to choose characteristics. By using the "best K" strategy, the user may select the exact amount of characteristics they want to utilize for categorization in the future while disregarding any more features that were included in the first batch of data. By employing the fit transform approach, which is a variation of the methodology, we may change the model and convert the initial dataset's full set of characteristics into a new set of data that only contains the most important ones[15]:

$$\text{K best} = n_1\ (\bar{x}_1-\bar{x})^2+n_2\ (\bar{x}_2-\bar{x})^{2\ 16}$$

N_1 = No of class 1
N_2 = No of class 2
$\bar{x}$ = Mean all class
$\bar{x}_1$ = Mean class 1
$\bar{x}_2$ = Mean class 2

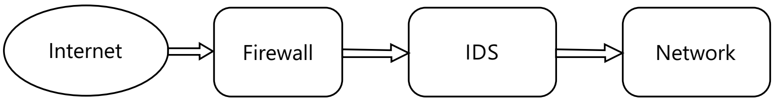

Fig. 1: IDS model

Random forest classify (RF): RF is a supervised machine learning approach since it has been used extensively because to its good performance[17]. A prediction is made using the RF, an ensemble approach that makes use of many CARTs. To replace the trees, a series of training sets is created (a bagging approach). To assess how well the RF model performs, the remaining one-third of the data also referred to as out-of-the-bag samples is used in an internal cross-validation approach. To train the trees, about two-thirds of the data is employed[18]:

$$\text{Entropy }(T) = \sum_{i=1}^{c} = -p_{i\,\log 2}\, p_i \quad [19]$$

$$\text{Entropy }(T, Z) = \sum_{c\in Z} p\,(c)\,\text{Entropy }(c) \quad [19]$$

$$\text{Gain }(T, Z) = \text{Entropy }(T)\text{-Entropy }(T, Z) \quad [19]$$

T is P_i the probability of classes ci in T, c is the kind of classes. Entropy (T, Z) entropy of feature select.

NSL-KDD dataset: After it was discovered that the KDD99 data set had several issues, a new version was created, the NSL-KDD data set, which does not have any of the previous issues with data redundancy and drastically reduces the number of cases that have an impact on the estimation of the findings. The KDDCup99 dataset and NSL-KDD share the same features[9]. The NSL-KDD dataset train consists of (41) features and the number of records is 125,972. With 67,343 normal traffic instances and 58,630 traffic attack instances.

General design of the proposed model: In this section, we provide details of the proposed model design, which consists of data pre-processing, feature selection and model design by the random forest algorithm. A description of the IDS proposed in this paper was shown in Fig. 2.

Preparing and pre-processing the dataset: The dataset utilized (NSL-KDD) reflects network traffic in real-time and includes sets from assaults. The major goal of utilizing a data set is to test the suggested approach in actual contexts in order to demonstrate how well the model performs. A crucial step in reducing false alarm rates and raising accuracy isdata set preparation and preprocessing. The pretreatment steps are as follows:

Step 1: Entering the data of the used NSL-KDD (KDD Train) into the system

Step 2: Check and delete duplicate rows. And remove the features redundant

Step 3: Treatment of missing values. Identify the missing values in the rows. Drop nan and null from all instances as well

Step 4: The data is split randomly into two parts (70% training and 30% testing)

Step 5: Using Sk learn variance threshold to exclude all features with low variance and features with the same value given that it just considers the qualities (x), as opposed to the anticipated outcomes, it may be utilized for unsupervised learning (y). The first step is to identify and eliminate features that are not crucial to the system. See algorithm 1.

Algorithm 1: Remove feature low variance threshold

Input: Training data and testing data.

Output: Features high variance.

```
Begin
    1. For get features from dataset
    2. Define variance threshold = 0
    3. Ordinal encoder assigns an integer value to each unique category value.
    4. Count the number of feature from zero to length dataset.
    5. Find the value variance to feature variance:

            Variance score (fi) = p (1-p)

    6. Drop feature where value < = 0
    7. End for
    8. End
```

Step 6: Use the standard scaler to resize the data frames and to make the characteristics more homogeneous:

$$\text{Scale} = \frac{(x - m)}{s} \quad (20)$$

M = Mean

SD = Standard deviation

Step 7: Standard deviation should be checked. If the data points are farther from the mean, the variance within the dataset will be higher, according to the calculation of each data point's departure from the mean. The data is more erratic the higher the standard deviation:

$$\text{SD} = \sqrt{\frac{\sum (X - U)^2}{N}} \quad (21)$$

X value in the dataset, U the mean value of the dataset and N is the total number of data points.

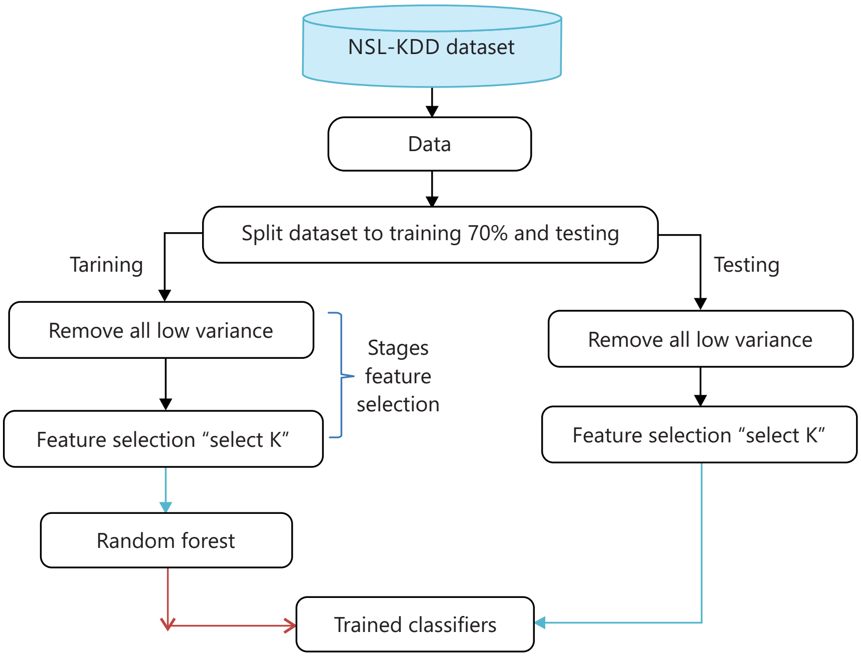

Fig. 2: Proposed model's general layout

Features selection: After some features were removed in the first stage, we'll start with the second stage of feature selection. The most crucial features are chosen using the feature selection approach. The second and most important stage is when the characteristics are determined and reduced. Univariate feature selection is employed in this stage. Using the classification (f classif) technique[16] and the feature selection algorithm (select K best). Look at algorithm 2.

Algorithm 2: Feature selection by select K best, f_classif

```
Input: The features with high variance for each type attack to training and testing.
Output: The features highest scores for each type attack to training and testing.
Begin
    1. Features with high variance
    2. Compute the score of each feature in dataset:

           $K\ best = n_1\ (\bar{x}_1-\bar{x})^2+n_2\ (\bar{x}_2-\bar{x})^2$ [16]

    3. Repeat until all end features.
    4. Select K feature f_classif (X_train, y_train) the high score features (three sets) and keep it for each type attack.
    5. End
```

Proposed model's training phase: The proposed approach employs a learning strategy to optimize classification via a random forest algorithm. After the important features have been identified through the previous stages. The training data is now entered into the random forest algorithm to classify and train the data as shown in the algorithm 3.

Algorithm 3: Random forest classifier

```
Input: The training dataset
Output: Classification of normal and abnormal network traffic
Begin
    1.  Choose number of trees in the forest N
    2.  For I to N
    3.  Choose at random one of the features A from the feature set
    4.  For J to A
    5.  Compute the gain information

            $Entropy\ (T) = \sum_{i=1}^{c} = -p_{i\ \log 2}\, p_i$

            $Entropy\ (T, Z) = \sum_{c \in Z} p\ (c)\ Entropy\ (c)$

            $Gain\ (T, Z) = Entropy\ (T) - Entropy\ (T, Z)$

    6.  Choose the node with the most information gain
    7.  Dividing the node into sub-nodes
    8.  Go to step 4 where j<> A
    9.  End for
    10. Go to step 1 where I<> N
    11. End for
    12. End
```

RESULTS

In this section, the experiment results are shown, the performance of the suggested intrusion detection model is assessed and a detailed explanation of how to apply the proposed work is provided. The first goal was to reduce and select the best features by two methods. The second goal is model building based on the random forest algorithm to reduce time training and time detection. The models were trained and tested using the (NSL-KDD) dataset for intrusion detection. The model proposed performance was assessed in terms of Acc, DR and precision, FAR, F-score, Recall and ER. The platform for the system workspace will be the following one: Windows 11 Professional, operating system (64 bit), Intel (R) Core (TM) i7-11650G CPU @2.80 GHz and 16 GB random access memory, Python 3.10.4 and utilized Jupyter Notebook (server: 6.4.11).

A NSL-KDD train of the dataset's 125973 records, which contain samples of normal behavior and attacks, was employed in this study. By using cross-validation, the dataset is divided into training data and testing data in order to evaluate the proposed system. The number of selected samples training (88181) and the number of test samples (37792). The proposed model has been applied to detect normal and attack traffic with the use of a random forest algorithm. The performance of the system depends on the accuracy of intrusion detection, reducing false alarms, reducing training time and intrusion detection time. In the first performance evaluation, the system was implemented on the algorithm without inserting any method for selecting the features. The results were as shown in the confusion matrix. All the records that are in the normal class are 20,083. The 20,030 of them are correctly predicted in the normal class and 53 of them are incorrectly predicted in the attack class. There are also 17,709 records in the attack log. The 17,683 of them are correctly predicted in the attack class and 26 of them are incorrectly predicted in the normal class.

In the second performance evaluation, the first stage of feature reduction and selection (variance threshold) was applied. The 40 features were used. The results were as shown in the confusion matrix. All the records that are in the normal class are 20,027. The 19,914 of them are correctly predicted in the normal class and 113 of them are incorrectly predicted in the attack class. There are also 17,716 records in the attack log. The 17,663 of them are correctly predicted in the attack class and 35 of them are incorrectly predicted in the normal class.

In the final performance evaluation, a second stage of appropriate feature selection was applied by univariate feature selection (select K best, f_classif). The 28 features were used as shown in Table 1. The final results of the proposed model were shown in Table 2.

The confusion matrix produced by the implementation indicates that 49 ordinary occurrences were mistakenly identified as assaults. It accurately identified 20034 as a typical occurrence and detected it as such. Furthermore, assaults were accurately identified in 17689 incidents. Additionally, 20 attacks were mistakenly thought to be normal.

Table 1: Features final used in model proposed

No of features	Feature name	Feature name	Feature name	Feature name
28	Protocol type	Service	Flag	Src_bytes
	Dst bytes	Wrong fragment	Logged in	Num access files
	Is host login	Is guest login	Srv count	Serror rate
	Srv serror rate	Error rate	Srv rerror rate	Same srv rate
	Diff srv rate	Srv diff host rate	Dst host count	Dst host srv count
	Dst host same srv rate	Dst host diff srv rate	Dst host same src	Port rate
	Dst host srv diff host rate	Dst host serror rate	Dst host srv error rate	Dst host error rate
	Dst host srv error rate			

Table 2: Results of all the proposed model

Stages	Acc (%)	Recall	Precision	F-score	Time training (sec)	Time detection (sec)
Standard	0.9979	0.9973	0.9987	0.9980	3.90	0.27
First stage	0.9943	0.9915	0.9976	0.9946	3.70	0.25
Second and final stage	0.9981	0.9975	0.9990	0.9982	3.47	0.24

Table 3: Compare the proposed work with related works

Reference	Acc (%)	Recall	Precision	F-score	Time training (sec)	Time detection (sec)
Sarker *et al.*[5]	0.98	0.98	0.98	0.98	-	-
Farnaaz and Jabbar[6]	0.9967	-	-	-	-	-
Kayode-Ajala[7]	97.94	97.73	-	-	-	-
Gao *et al.*[8]	0.998	0.998	0.9979	0.9979	-	0.70
Mohammed and Gbashi[10]	94	-	-	-	-	-
Proposed model	0.9981	0.9975	0.9990	0.9982	3.47	0.24

Accuracy (Acc): It is a scale used to measure the accuracy of the proposed model in classifying the elements as being attack or normal. It is computed by the following equation[22]:

$$\text{Accuracy (Acc)} = \frac{\text{TN} + \text{TP}}{\text{All (TP} + \text{TN} + \text{FP} + \text{FN)}}$$

Precision: It is the proportion of the classifier's accurate positive predictions. Using the following equation:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Recall: It is the proportion of accurate positives that the classifier has actually discovered:

$$\text{Recall or DR} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

F-score: Referred to the symmetric mean of precision and recall:

$$\text{F} = \frac{2 \times \text{Presision} \times \text{Recall}}{\text{Presision} + \text{Recall}}$$

By studying the results obtained from implementing the proposed methods for feature reduction and selection of features with the random forest algorithm on a dataset (NSL-KDD), it proved superior to the basic model and to the model with one stage of feature selection in terms of model accuracy, recall, precision, training time and detection time.

It was concluded from the presented results: (1) The method used in preprocessing and dividing attacks based on the type of attack in data analysis was good. (2) Using the binary selection method, it was able to select the best features because 19 features were used for one dataset (CIC-IDS2017) and 22 features were used for one dataset (NSL-KDD). This reduced the training time of the system without affecting its accuracy. (3) The proposed system performed well when selecting the best 19 out of 79 features for the CIC-IDS-2017 dataset and selecting the best 22 out of 41 features for NSL-KDD. (4) The use of group learning by (stack of estimators) with the best parameters for all algorithms in the development of the random forest algorithm gives a very high accuracy and detection rate, especially with the method of identifying only important features and dividing. Attacks according to the type of attack and very low detection time compared to previous works in network intrusion detection systems. (5) By studying the results obtained after testing the proposed model, it is possible to increase the accuracy of the model and reduce the false alarm rate, but there is a slight increase in the attack detection time. The results were also compared with related works[6-9,11] (Table 3), that also used the random forest algorithm and the same dataset and our proposed model proved its superiority in all measurements as well as in the time of training and detection. This proves the strength of the proposed model and shows the importance of reducing and selecting features in models. That plays an important role in the accuracy of the intrusion detection systems.

CONCLUSION

Many intrusion detection systems have been proposed with many methods and algorithms. Most systems are based on data mining. In this paper, a model based on two stages of features selection (variance threshold) and (select K best, fclassif) selection is proposed. The random forest algorithm was used for classification and the test was carried out on a dataset (NSl-KDD). The accuracy of (0.0981), recall (0.9975), precision (0.9990) and intrusion detection time (0.24 sec) were obtained. The results were compared with related work and proved the superiority of our proposed work. The proposed model was also compared according to the stages of work. In the future, research work on IDS development could be based on algorithm development in other ways and the application of the system to other modern and more realistic data sets to help increase the power of the intrusion detection model in predicting attacks.

SIGNIFICANCE STATEMENT

An intrusion detection system is a technique that may be used to detect known and unknown intrusions before one of them damages the network hardware. One of the most important things that plays a significant role in the strength of the system is the selection of good features in the training of the system. In this paper, intrusion detection systems based on reduction and feature selection using "dual feature selection" with random forest algorithm are proposed. Using this choice, the accuracy of the system was 0.998.

REFERENCES

1. Chen, X. and W. Susilo, 2014. Guest editorial: Special issue on cyber security protections and applications. J. Internet Serv. Inf. Secur., 4: 1-3.
2. Liu, C., Z. Gu and J. Wang, 2021. A hybrid intrusion detection system based on scalable K-means+random forest and deep learning. IEEE Access, 9: 75729-75740.
3. Wang, M., K. Zheng, Y. Yang and X. Wang, 2020. An explainable machine learning framework for intrusion detection systems. IEEE Access, 8: 73127-73141.
4. Liu, J., X. Kong, X. Zhou, L. Wang and D. Zhang *et al*., 2019. Data mining and information retrieval in the 21st century: A bibliographic review. Comput. Sci. Rev., Vol. 34. 10.1016/j.cosrev.2019.100193.
5. Sarker, I.H., Y.B. Abushark, F. Alsolami and A.I. Khan, 2020. IntruDTree: A machine learning based cyber security intrusion detection model. Symmetry, Vol. 12. 10.3390/sym12050754.
6. Farnaaz, N. and M.A. Jabbar, 2016. Random forest modeling for network intrusion detection system. Procedia Comput. Sci., 89: 213-217.
7. Kayode-Ajala, O., 2021. Anomaly detection in network intrusion detection systems using machine learning and dimensionality reduction. Sage Sci. Rev. Appl. Mach. Learn., 4: 12-26.
8. Gao, X., C. Shan, C. Hu, Z. Niu and Z. Liu, 2019. An adaptive ensemble machine learning model for intrusion detection. IEEE Access, 7: 82512-82521.
9. Negandhi, P., Y. Trivedi and R. Mangrulkar, 2019. Intrusion Detection System Using Random Forest on the NSL-KDD Dataset. In: Emerging Research in Computing, Information, Communication and Applications, Shetty, N.R., L.M. Patnaik, H.C. Nagaraj, P.N. Hamsavath and N. Nalini (Eds.), Springer, Singapore, ISBN: 978-981-13-6000-8, pp: 519-531.
10. Mohammed, B. and E.K. Gbashi, 2021. Intrusion detection system for NSL-KDD dataset based on deep learning and recursive feature elimination. Eng. Technol. J., 39: 1069-1079.
11. Vinayakumar, R., M. Alazab, K.P. Soman, P. Poornachandran, A. Al-Nemrat and S. Venkatraman, 2019. Deep learning approach for intelligent intrusion detection system. IEEE Access, 7: 41525-41550.
12. Kaur, T., V. Malhotra and D. Singh, 2014. Comparison of network security tools-firewall, intrusion detection system and honeypot. Int. J. Enhanced Res. Sci. Technol. Eng., 3: 200-204.
13. Zhengtian, Z., R. Zhiyuan and D. Xiaoyan, 2023. Feature selection for binary classification based on class labeling, SOM, and hierarchical clustering. Meas. Control, 56: 1649-1669.

14. Wang, J., X. Li and H. Liang, 2022. A new exact p-value approach for testing variance homogeneity. Stat. Theory Relat. Fields, 6: 81-86.
15. Garcia-Chimeno, Y., B. Garcia-Zapirain, M. Gomez-Beldarrain, B. Fernandez-Ruanova and J.C. Garcia-Monco, 2017. Automatic migraine classification via feature selection committee and machine learning techniques over imaging and questionnaire data. BMC Med. Inf. Decis. Making, Vol. 17. 10.1186/s12911-017-0434-4.
16. Bavithra, K.B. and R.S. Kumar, 2019. High throughput K best MIMO detector using modified final selector based carry select adder. Microprocess. Microsyst., Vol. 71. 10.1016/j.micpro.2019.102847.
17. Musbah, H., H.H. Aly and T.A. Little, 2021. Energy management of hybrid energy system sources based on machine learning classification algorithms. Electr. Power Syst. Res., Vol. 199. 10.1016/j.epsr.2021.107436.
18. Belgiu, M. and L. Drăguţ, 2016. Random forest in remote sensing: A review of applications and future directions. ISPRS J. Photogramm. Remote Sens., 114: 24-31.
19. Shareef, S.M. and S.H. Hashim, 2018. Intrusion detection system based on data mining techniques to reduce false alarm rate. Eng. Technol. J., 36: 110-119.
20. Carlson, B.E., N. Østgaard, P. Kochkin, Ø. Grondahl and R. Nisi *et al*., 2015. Meter-scale spark X-ray spectrum statistics. J. Geophys. Res.: Atmos., 120: 191-202.
21. Macaskill, P., 2018. Standard deviation and standard error: Interpretation, usage and reporting. Med. J. Aust., 208: 63-64.
22. Sun, P., P. Liu, Q. Li, C. Liu, X. Lu, R. Hao and J. Chen, 2020. DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system. Secur. Commun. Networks, Vol. 2020. 10.1155/2020/8890306.