Indonesian Journal of Electrical Engineering and Computer Science
Vol. 24, No. 2, November 2021, pp. 1100~1106
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v24.i2.pp1100-1106

Intrusion detection system based on bagging with support vector machine

Ali Khalid Hilool, Soukaena H. Hashem, Shatha H. Jafer
Department of Computer Science, University of Technology, Baghdad, Iraq

Article Info

Article history:

Received Jun 6, 2021
Revised Aug 27, 2021
Accepted Sep 2, 2021

Keywords:

Bagging
Computer worms
Ensemble learning
IDS
SVM

ABSTRACT

Due to their rapid spread, computer worms perform harmful tasks in networks, posing a security risk; however, existing worm detection algorithms continue to struggle to achieve good performance and the reasons for that are: First, a large amount of irrelevant data affects classification accuracy. Second, individual classifiers do not detect all types of worms effectively. Third, many systems are based on outdated data, making them unsuitable for new worm species. The goal of the study is to use data mining algorithms to detect worms in the network because they have a high ability to detect new types accurately. The proposal is based on the UNSW NB15 dataset and uses a support vector machine to train and test the ensemble bagging algorithm. To detect various types of worms efficiently, the contribution suggests combining correlation and Chi2 feature selection method called Chi2-Corr to select relevant features and using support vector machine (SVM) in the bagging algorithm. The system achieved accuracy reaching 0.998 with Chi2-Corr, and 0.989, 0.992 with correlation and chi-square separately.

Corresponding Author:

Ali Khalid Hilool
Department Computer Science
University Of Technology
Baghdad, Iraq
Email: cs.19.42@grad.uotechnology.edu.iq

1. INTRODUCTION

Worms are a hazard because they are self-contained and do not count on other software for propagation, allowing them to spread rapidly. A lot of methods have been used to detect computer worms and estimate their damage, including the use of encryption, firewalls, machine learning approaches, and a variety of other methods [1], [2]. Intrusion detection system (IDS) is one of the most reliable systems for detecting penetrations and attacks in network infrastructure [3]-[5]. Anomaly-based detections and misuse-based detections are two types of intrusion detection methods. Attacks are identified using anomaly-based detection approaches based on their behavior. When a departure from usual behavior is noticed in a connection, it is classed as an assault. Misuse-based detection, on the other hand, identifies an incursion by comparing it to preset signatures. As a result, understanding the characteristics of assaults is required in order to create a misuse-based detection system [6], [7]. IDS generates a large number of false alarms, which has prompted many researchers to seek a solution to distinguish alerts to less serious incidents and reduce false alarms, which are both false positive (FP) and false negative (FN). IDS based on data mining techniques can be used to improve IDS in actual time, eliminate normal activity from alarm data in order to focus on real attacks, and detect abnormal activity that reveals a real attack [8], [9]. IDS consists of two sorts of approaches: host-based (HIDS) and network-based (NIDS) [10]. HIDS is the most common type of IDS; its primary function is internal monitoring (of a computer or machine); however, several versions of HIDS have been developed that may be

used to monitor a network. HIDS determine whether a system has been hacked and issue appropriate warnings to administrators [11]. A network intrusion detection system (NIDS) is used to monitor and manage network traffic in order to protect a system against network-based attacks [12], [13]. Because of the consequences of increased security attacks today, network intrusion detection systems (NIDS) have become the most critical part of modern network technology. The intrusion detection system (IDS) generates a large number of alarms; however, algorithmic procedures are used to reduce false positives [14], [15].

Ensemble learning is a machine learning technique that entails teaching a group of bad learners (models) to solve a problem and then combining their results to get better results. The basic idea is that we can get more accurate and/or robust models by combining weak models in the right way [16]. The three types of ensemble approaches are as follows. Bagging is a method of combining homogeneous weak learners, training and testing them in parallel, and then combining them using average voting [17]. Boosting, which brings together a group of similar poor learners and trains and tests them in a systematic manner (each iteration depends on previous ones) [18]. Stacking is an ensemble method in which a new model learns the most efficient way to combine the predictions of multiple existing models [19].

The computer worm works to cause great damage to network systems, and systems that depend on classification and that are used to prevent it, suffer from several problems, as some of them use individual classifiers where new types are not discovered with high accuracy due to the limitations of individual classifiers. The data used in this field is often outdated and obsolete and suffers from the repetition of data and the presence of irrelevant data and wrong and distorted data. Therefore, all of this will affect the accuracy of the classification and will lead to a high false alarm rate. In order to overcome these limitations, we will first use the latest intrusion detection dataset (UNSW-NB15), which has fewer problems than its predecessors. And we make pre-processing it to get rid of the distorted data, then we propose to combine two methods of identifying features (Chi2-Corr) to determine only the features related to our problem, then we will use the ensemble methods that work on the principle of (union is strength) to overcome the problems of individual classification and give the highest accuracy of classification With the lowest false alarm rate.

The rest of the paper is organized as follows: In section 2, we describe the architecture of the worm detection system, which is based on ensemble bagging. Section 2.1 discusses the unsw-nb15 dataset, in sections 2.2 and 2.3 we discuss the preprocessing steps and feature selection methods which include using chi2 and correlation. In section 2.4, we'll go over how to construct a bagging classifier and how to train and test a model using an support vector machine (SVM) classifier. In section 3, we analyze our extensive tests for evaluating the proposed worm detection method. Section 4 wraps up by elaborating on the conclusion.

2. RESEARCH METHOD

To improve the detection of worms in networks, we propose an effective data mining model for worm detection that uses both anomaly and misuse detection techniques, where each case in a dataset is labeled as "attack" or "normal" (worms are one kind of attack), and a learning algorithm is trained over the class data. The structure of the proposed worm detection model is shown in Figure 1. Which is broken down into four distinct stages:

1) Dataset preprocessing: To prepare the data for the classification algorithm, we first add preprocessing steps to the initial datasets.
2) 2- Dimensionality reduction: To pick the most important features and reduce the dimensionality of the dataset, a feature selection strategy called (Chi2-Corr) based on chi-square and correlation features selection is used.
3) Classifier training: To improve the accuracy of worm detection, we use the bagging algorithm to construct classifiers.
4) To forecast the outcome of our model, we used classification (testing).

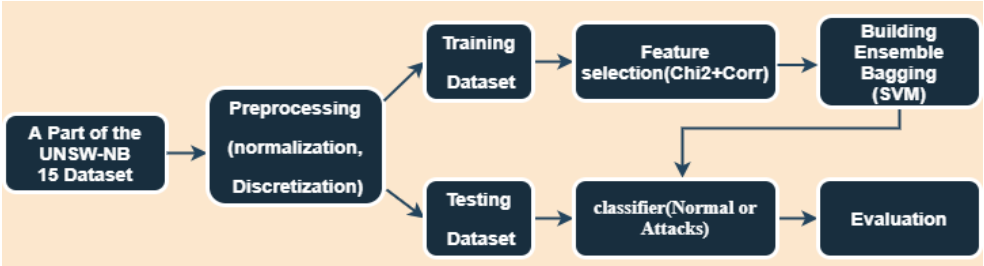

Figure 1. Flowchart of proposed worm detection

2.1. The unsw-nb15 dataset

The proposed model is trained using the UNSW-NB15 dataset. There are 2,540,044 instances in this dataset [20]. This data is split into four huge creating shared value (CSV) directories. There are detached training and testing sets. The training dataset consists of 175,341records, and the testing consists of 82,332 records. It has 45 columns, one for id and forty-four for features. The proposed model was trained using 5000 records from the aforementioned training and testing records, of which 154 records contain worms and the rest contain normal and other types of attacks. UNSW-NB15 dataset is consists of normal data and nine types of attacks (Backdoors, Dos attacks, Exploits attacks, Fuzzers attacks, Generic attacks, Reconnaissance attacks, Shellcode attacks, and Worms attacks) are all included in these training and testing datasets [21].

2.2. Preprocessing

Because the UNSW-NB15 dataset contains both continuous and discrete features, it is necessary to convert the continuous attributes to discrete to ensure the system's efficiency and to deal with the issue of new values appearing in the test dataset that are not present in the training dataset. We used the Min-Max normalization process following discretization to improve the model's efficiency and effectiveness by placing attribute values between (0-1) [22]. We will use correlation feature selection and chi-square feature selection to exclude unused and redundant features from the dataset after discretization and normalization (See Algorithm 1).

Algorithm (1) min-max normalization
input : subset unsw-nb15 Datasets
Output: data values ranging from zero to one
For each column in the Dataset
extract the largest number in column
extract the smallest number in column
For each (X) rate in Feature extract
$$Value\_of(X) = \frac{Value(X)-Min}{Max-Min}$$
End For
End For

2.3. Feature selection

One of the most critical preprocessing steps in machine learning techniques is feature selection, which is used to remove unnecessary and redundant features from the dataset, enhance the model's efficiency by using the correct features and reduce the amount of time it takes to process the data [23], [24]. In this study, we used chi2 features selection and correlation features selection. See algorithm (2).

- Correlation feature selection

Correlation-based feature selection (CFS) ranks attributes using a heuristic assessment function based on correlations. The function compared attribute vector subsets that are connected to the class label but not to one another. The CFS method assumes that irrelevant features have a low correlation with the class and, as a result, should be neglect. Excess characteristics, on the other hand, should be looked at because they are frequently correlated with one or more of the other attributes. The criterion for evaluating a subset of n features is as follows:

$$M_s = \frac{N\overline{t_{cf}}}{\sqrt{N+N(N-1)\overline{t_{ff}}}} \quad (1)$$

MS signifies the evaluation of a subset of S that has N characteristics. $\overline{t_{cf}}$. is the average of the correlation between attributes and class labels. $\overline{t_{ff}}$. is the average correlation between two characteristics [25], [26].

- Chi-square feature selection

A statistical test is a Chi2 test. The Chi2 test examines the relationship between a class and a feature, allowing it to select features that are more relevant for a given dataset. As a result, features that aren't relevant for categorization can be removed from the feature space [27]. From the data of two features, we will get the observed count A and anticipated count E. The Chi-Square test is used to measure how far anticipated count E and observed count A differ.

$$X_c^2 = \frac{(A_i-E_i)^2}{E_i} \quad (2)$$

Where C is the degree of freedom, A denotes the observed value(s), and E denotes the expected value (s). We compare the value of $X_c^2$to the value of the chi2 table value where alpha =0.05 and delete the feature if it is less than the chi2 table value (independent); else, the feature is accepted.

```
Algorithm (2)  Chi-Corr feature selection
input : subset unsw-nb15 Datasets
Output: independent features with a strong connection to class, and features that are
class-dependent
Start
Step 1:  correlation CFS
           For each class column
           Extract  the correlation of class with all features
           Choose features that have a strong relationship with class.
            Remove the remainder
           End For
           For each feature in the subset you've chosen,
              Extract the correlation of feature with all features
                    Remove the remainder
           End For

Step 2: Chi square feature selection
       For each unsw-nb15Dataset feature
           seek for X_c^2with class. See (1)
           alpha=0.05
           from chi2 table find  X_c^2' where alpha=0.05 and matched it to X_c^2
            If X_c^2 < X_c^2' the feature is independent (droped)
            Else it is depend on class (not drop)
                 End For
End
```

2.4. Training and testing

Following feature selection methods, we will divide the dataset into two parts: training and testing. Training contains 67 percent of the total number of records in the dataset, while testing contains 33 percent. The proposed model is trained and tested using the two parts. Then we will distribute the training part on three parallel SVM in ensemble bagging algorithm to make classification decisions. See algorithm (3).

```
Algorithm (3) Bagging SVM Ensemble Algorithm.
Input: A subset of UNSW-NB15 Dataset
Output: SVM Bagging Model
Begin
Steps:
Step 1: dividing dataset into three samples
Srep 2: Foreach Sample aply SVM algorithm
            -Initialize (Xi, Yj) for all training dataset points, where X is a data
    vector (x1…. , xn) and Y is a
             class vector.
            -Set the weight W vector.
               -Allotment points of (x, y) and elicitation the hyper plane separator.
               -Heck the hyper plane if it is provides the best separation, use it as a
classifier system for the
                classification of the unsw nb-15 testing dataset and switch to End;
otherwise, proceed to the next
                step.
              -Make the hyperplan bigger.
              -Set up the Lagrange multiplier. αi vector α1…αn.
              -Use the classification function.
              -Find the non-zero support vectors xi (support vectors are the points that
determine the rea of hyper
               plan).
              -Use the hyper plan that emerged after identifying support vectors as the
classifier model to classify
               the unsw nb-15 testing dataset.
          End For
          Make voting to return results
End
```

3. RESULTS AND DISCUSSION

The aim of this paper, as mentioned previously, is to develop a good-accuracy worm detection system. To remove irrelevant features and increase classification reliability, a model called Chi2-Corr that combines CFS and chi2 is used to evaluate a subset of the original features. Using the UNSW NB15 dataset, a bagging ensemble classifier is trained and tested during the classification stage using SVM as base estimator rather than decision tree which is the default estimator. The tests are conducted on a desktop PC with a 1.80 GHz Intel Core i3-3217U processor and 4GB of random access memory (RAM). The classification results of testing are either true positives (TP) (intrusion), true negatives (TN) (normal), FP

(misclassified as intrusion), (FN) (misclassified as normal), Unknown (new attacks). Table 1 and Table 2 shows the results of applying the bagging technique to classify the records in the testing dataset using SVM and diffusion tensor (DT) classifiers. In Table 1 When all features are selected and feature selection procedures are used, these findings reveal that the TP is bigger than the TN, FP, FN, and unknown. When employing feature selection methods in the SVM classifier, the FP rates drop from 15 to 10 when using CFS, 6 when using Chi2, and 0 when using Chi2-Corr.

Table 1. Classification results of bagging with SVM classifiers

Feature selection measure	TP	TN	FP	FN	Unknow
Chi2-Corr.	938	710	0	2	0
CFS	917	715	10	8	0
Chi2	913	725	6	6	0
ALL	902	721	15	12	0

Table 2. Classification results of bagging with DT classifiers

Feature selection measure	TP	TN	FP	FN	Unknow
Chi2-Corr.	920	713	10	7	0
CFS	900	701	25	24	0
Chi2	930	698	17	5	0
ALL	899	683	40	28	0

The detection rate (DR) is the ratio between the number of TP and the total number of intrusion records presented in the testing dataset. It has been computed using the following equation: DR = TP/(TP+FN+Unknown), and the false alarm rate (FAR) is the ratio between several "normal" records classified as attacks (FP) and the total number of "normal" records presented in the testing dataset. It has been computed using the following equation: FAR = FP/(TN+FP +Unknown), Selection of the best classification model can be done according to its classification accuracy, which is the ratio between the number of correctly classified patterns (TP, TN) and the total number of patterns of the testing dataset. The accuracy (Acc) of each classifier has been computed using Acc =(TP+TN)/(TP+FP+TN+FN+unknown), false discovery rate (FDR=FP/FP+TP), Precision=TP/(TP+FP), Specificity =TN/(TN+FP), F-measure=(2*Recall*Precision)/(Recall+Precision). Values for all mentioned metrics have been illustrated in Table 3 and Table 4.

Table 3 summarizes the outcomes from the UNSW-NB15 dataset, which includes the results of the ensemble Bagging with the SVM classifier. It is proposed that the ensemble classifier is not optimal enough in numerous criteria without feature selection. Performance, on the other hand, improves to the best feasible case when feature selection methods are applied. Without employing feature selection methods, our suggested system achieves an accuracy of 0.983, FAR of 0.020, DR of 0.986, Precision of 0.983, F-measure of 0.984, Specificity of 0.979, and false discovery rate (FDR) of 0.016. When employing feature selection methods, the results are optimized, and when using Chi2-Corr. the best case is reached with the maximum accuracy of 0.998, FAR of 0.0, and DR of 0.998, Precision of 1.0, F-measure of 0.998, Specificity of 1.0, FDR of 0.

Table 3. Metrics to evaluate ensemble bagging with SVM

Metrics	All	CFS	Chi2	Chi2-Corr
n. of.feature	44	33	33	27
Accuracy	0.983	0.989	0.992	0.998
DR	0.986	0.991	0.993	0.998
FAR	0.020	0.013	0.008	0.0
Precision	0.983	0.989	0.993	1.00
F-measure	0.984	0.989	0.993	0.998
Specificity	0.979	0.986	0.991	1.00
FDR	0.016	0.010	0.006	0.0

Table 4. metrics to evaluate ensemble Bagging with DT

Metrics	All	CFS	Chi2	Chi2-Corr
n. of .feature	44	33	33	27
Accuracy	0.958	0.97	0.986	0.989
DR	0.969	0.974	0.994	0.992
FAR	0.055	0.034	0.023	0.13
Precision	0.957	0.972	0.982	0.989
F-measure	0.962	0.972	0.987	0.99
Specificity	0.944	0.965	0.976	0.986
FDR	0.042	0.027	0.017	0.010

While using the DT classifier in conjunction with the ensemble Bagging algorithm Our proposed system has an accuracy of 0.958, FAR of 0.055, DR of 0.969, Precision of 0.957, F-measure of 0.962, Specificity of 0.944, and FDR of 0.042 without applying feature selection methods. The results are optimized when employing feature selection methods, therefore when using Chi2-Corr, the best case is attained with the greatest accuracy of 0.989, FAR of 0.013, DR of 0.992, Precision of 0.989, F-measure of 0.99, Specificity of 0.986, and FDR of 0.010. We compare our proposed system to some related work to better grasp the benefits of the suggested methodology. Table 5 shows the outcomes of the comparison.

The comparison includes the classification method, the selected dataset, feature selection approaches, the number of selected features, accuracy, FAR, and DR for intrusion detection, as shown in Table 5. When compared to C4.5, RF, our system has the best accuracy and detection rate, as well as the

lowest false alarm rate. When we compare our proposed system to the SVM, we can see that the ensemble method has advantages over the SVM, which is a single classifier with a huge variance. As a result, ensembles frequently reduce the variance component of contributing model prediction errors, resulting in a significant improvement in accuracy (from 0.85 to 0.998) as well as a reduction in the false alarm rate (FAR) (from 15.26 to 0.0). When SVM was used as the base estimator, our proposed method had the highest accuracy and detection rate when compared to other ensemble approaches like stacking, and boosting.

Table 5. Compression between the proposed system and the related work

Method	Dataset	Feature selection	n. of features	ACC
C4.5	CIC-IDS2017	CFS-BA	10	0.98
RF	CIC-IDS2017	CFS-BA	10	0.993
SVM	Train and Test UNSW-NB15	N/A	44	0.85
STACKING	UNSW-NB15	N/A	49	0.628
Boosting	UNSW-NB15	N/A	49	0.947
Proposed System	Subset of Train and Test UNSW-NB15	CFS-Chi2s	27	0.998

4. CONCLUSION

The suggested approach stresses the necessity of network intrusion detection systems (IDS) for detecting worm assaults, which are the most dangerous attacks in a network and have an impact on resource availability. Because of the normalizing and discretization operations, the suggested system is more efficient. The correlation and chi2 algorithms are offered as feature selection approaches to improve the accuracy of the proposed system and reduce the amount of time required. The accuracy of the Bagging classifier, which employs SVM and is assisted by Chi2-Corr, is better than utilizing all features or using Bagging Classifier with Corr or chi2 with 33 features, also the Chi2-Corr has a lower false alarm rate than CFS or Chi2. Using a decision tree classifier as a base estimator in Bagging (without our contribution) will result in a system that is less accurate, has less detection rate, and have a false alarm rate.

REFERENCES

[1] Y. Yao, Q. Fu, W. Yang, Y. Wang, and C. Sheng, "An Epidemic Model of Computer Worms with Time Delay and Variable Infection Rate," *Security and Communication Networks* 2018, vol. 2018, doi: 10.1155/2018/9756982.

[2] S. H. Hashem and I. A. Abdulmunem, "A proposal to detect computer worms (malicious codes) using data mining classification algorithms," *Engineering and Technology Journal*, vol. 31, no. 2, 2013.

[3] A. D. Cesare *et al.*, "Combination of flow cytometry and molecular analysis to monitor the effect of UVC/H2O2 vs UVC/H2O2/Cu-IDS processes on pathogens and antibiotic resistant genes in secondary wastewater effluents," *Water Research*, vol. 184, p. 116194, 2020, doi: 10.1016/j.watres.2020.116194.

[4] S. H. Hashem, "Enhance network intrusion detection system by exploiting br algorithm as an optimal feature selection," *Handbook of Research on Threat Detection and Countermeasures in Network Security*, IGI Global, 2015, doi: 10.4018/978-1-4666-6583-5.ch002.

[5] T. B. Seong, V. Ponnusamy, N. Z. Jhanjhi, R. Annur, and M. N. Talib, "A comparative analysis on traditional wired datasets and the need for wireless datasets for IoT wireless intrusion detection," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 22, no. 2, May 2021, pp. 1165-1176, doi: 10.11591/ijeecs.v22.i2.pp1165-1176.

[6] B. N. Kumar, M. S. V. Sivarama Bhadri Raju, and B Vishnu Vardhan, "A novel approach for selective feature mechanism for two-phase intrusion detection system," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 1, pp. 101-112, April 2019, doi: 10.11591/ijeecs.v14.i1.pp101-112.

[7] H. Suhaim *et al.*, "Genetic algorithm for intrusion detection system in computer network," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 3, pp. 1670-1676, September 2020, doi: 10.11591/ijeecs.v19.i3.pp1670-1676.

[8] S. H. Hashim and S. Sharef, "Intrusion detection system based on data mining techniques to reduce false alarm rate," *Engineering and Technology Journal*, vol. 36, no. 2, 2018, doi: 10.30684/etj.36.2B.3.

[9] S. H. Hashim and S. Sharef, "Proposed Hybrid Classifier to Improve Network Intrusion Detection System using Data Mining Techniques," *Engineering and Technology Journal*, vol. 38, no. 1B, 2020, doi: 10.30684/etj.v38i1B.149.

[10] N. N. Tran, R. Sarker, and Jiankun Hu, "An approach for host-based intrusion detection system design using convolutional neural network," *International conference on mobile networks and management*, Springer, vol. 235, 2017, doi: 10.1007/978-3-319-90775-8_10.

[11] V. Sstla, V. K. K. Kolli, L. K. Voggu, R. Bhavanam, and S. Vallabhasoyula, "Predictive Model for Network Intrusion Detection System Using Deep Learning," *Revue d'Intelligence Artificielle Journal*, vol. 34, no. 3, pp. 323-330, June 2020, doi: 10.18280/ria.340310.

[12] R. Samrin and D. Vasumathi, "Review on anomaly based network intrusion detection system," 2017 *International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEEECCOT)*, 2017, pp. 141-147, doi: 10.1109/ICEECCOT.2017.8284655.
[13] S. H. Hashem Mahmood and A. Hafza, "Network intrusion detection system (NIDS) in cloud environment based on hidden Naïve Bayes multiclass classifier," *Al-Mustansiriyah Journal of Science*, vol. 28, no. 2, 2017, doi: 10.23851/mjs.v28i2.508.
[14] S. H. Hashem, "Efficiency of Svm and Pca to enhance intrusion detection system," *Journal of Asian Scientific Research*, vol. 3, no. 4 p. 381, 2013, doi:
[15] S. K Majeed, S. H. Hashem, and I. K. Gbashi, "Propose hmnids hybrid multilevel network intrusion detection system," *International Journal of Computer Science Issues (IJCSI)*, vol. 10, no. 5, 2013, doi:
[16] X. Dong, Z. Yu, CAO2. Wenming, S. H. I Yifan, and M. A. Qianli, "A survey on ensemble learning," *Frontiers of Computer Science*, vol. 14, no. 2, pp. 241-258, 2020, doi: 10.1007/s11704-019-8208-z.
[17] T. Goksu, and D. Birant, "Enhanced bagging (eBagging): A novel approach for ensemble learning," *The International Arab Journal of Information Technology*, vol. 17, no. 4, July 2020, doi: 10.34028/iajit/17/4/10.
[18] L. Yiheng and W. Chen, "A Comparative Performance Assessment of Ensemble Learning for Credit Scoring," *Mathematics*, vol. 8, no. 10, p. 1756, doi: 10.3390/math8101756.
[19] S. K. Singh, K. K. Bejagam, Y. An, and S. A. Deshmukh, "Machine-learning based stacked ensemble model for accurate analysis of molecular dynamics simulations," *The Journal of Physical Chemistry A*, vol. 123, no. 24, pp. 5190-5198, 2019, doi: 10.1021/acs.jpca.9b03420.
[20] C. Sarika, and N. Kesswani, "Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 datasets using deep learning in IoT," *Elsevier Journal*, vol. 167, pp. 1561-1573, 2020, doi: 10.1016/j.procs.2020.03.367.
[21] M. K. Sydney and Y. Sun, "Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset," *Journal of Big Data*, vol. 7, no. 1 pp. 1-20, 2020, doi: 10.1186/s40537-020-00379-6.
[22] A. Batool, A. Ghamdi, S. Kamel, and M. Khayyat, "Evaluation of Artificial Neural Networks Performance Using Various Normalization Methods for Water Demand Forecasting," *2021 National Computing Colleges Conference (NCCC)*, IEEE, 2021, pp. 1-6, doi: 10.1109/NCCC49330.2021.9428856.
[23] R. R. Zebari, A. M. Abdulazeez, D. Q. Zeebaree, D. A. Zebari, and J. N. Saeed "A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction," *Journal of Applied Science and Technology Trends*, vol. 1, no. 2, pp. 56-70, 2020, doi: 10.38094/jastt1224.
[24] N. M. N. Mathivanan, N. A. Md. Ghani, and R. M. Janor, "A comparative study on dimensionality reduction between principal component analysis and k-means clustering Genetic algorithm for intrusion detection system in computer network," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 16, no. 2, pp. 752-758, November 2019, doi: 10.11591/ijeecs.v16.i2.pp752-758.
[25] Wosiak, Agnieszka, and D. Zakrzewska, "Integrating correlation-based feature selection and clustering for improved cardiovascular disease diagnosis," *Complexity*, 2018, doi: https: 10.1155/2018/2520706.
[26] R. Bobby *et al.*, "DUBStepR: correlation-based feature selection for clustering single-cell RNA sequencing Data," *bioRxiv* (2021), 2020-10, doi: 10.1101/2020.10.07.330563.
[27] L. Ali, C. Zhu, N. A. Golilarz, A. Javeed, M. Zhou and Y. Liu, "Reliable Parkinson's Disease Detection by Analyzing Handwritten Drawings: Construction of an Unbiased Cascaded Learning System Based on Feature Selection and Adaptive Boosting Model," in *IEEE Access*, vol. 7, pp. 116480-116489, 2019, doi: 10.1109/ACCESS.2019.2932037.