Mathematical Statistician and Engineering Applications
ISSN: 2094-0343
2326-9865

Network Intrusion Detection using Supervised Machine Learning Technique with Feature Selection

Dr. B. S. N. Murthy, Assoc Prof CSE:BVCE, murthy2007.b@gmail.com
Dr. K. Srinivas, Prof CSE:BVCE, ks.bvce@gmail.com
Mr. Shubhashish Jena, CSE,OUTR, BBSR, sjena1998@gmail.com
Angara V. L. Gopala Sandeep, CSE:BVCE
Male Swamy Naidu, CSE:BVCE
Masarapu Ravi, CSE:BVCE
Kanchustambham Sudheer, CSE:BVCE

Article Info
Page Number: 5242 - 5262
Publication Issue:
Vol 71 No. 4 (2022)

Article History
Article Received*: 25 March 2022*
Revised*: 30 April 2022*
Accepted*: 15 June 2022*
Publication*: 19 August 2022*

Abstract
A new supervised machine learning system is made to figure out whether network traffic is harmful or not. A combination of the supervised learning algorithm and the feature selection method has been used to find the best model based on how well it can detect. This study shows that Artificial Neural Network (ANN)-based machine learning with wrapper feature selection does a better job of classifying network traffic than the support vector machine (SVM) method. supervised machine learning techniques like SVM and ANN are used to classify network traffic from the NSL-KDD dataset in order to measure performance. Comparative studies show that the proposed model is better at detecting intrusions than other models that are already out there.

Keywords: network traffic, supervised machine learning, SVM, and NSL-KDD

1 INTRODUCTION

With more people using the internet and more people being able to access online content, cybercrime is also happening more and more [1-2]. The first step in stopping a security attack is to look for signs of intrusion. Studies pay a lot of attention to security solutions like Firewall, Intrusion Detection System (IDS), Unified Threat Modeling (UTM), and Intrusion Prevention System (IPS).

IDS can find attacks coming from many different systems and networks because it collects information and then looks at it for possible security holes [3]. The network-based IDS looks at the data packets that move through a network. It does this in two ways. Anomaly-based detection is still far behind signature-based detection, so it is still a major area of research [4-5]. The problem with anomaly-based intrusion detection is that it has to deal with new attacks for which there isn't enough information to figure out what's wrong. So, the system needs to be smart enough to tell which traffic is safe and which is dangerous or strange. Researchers have been looking into machine learning techniques for this purpose over the past few years [6]. IDS, on the other hand, is not the answer to all security problems. For example, IDS can't make up for weak mechanisms for identifying and authenticating users or for weak network protocols. With more people using the internet and more people being able to access online content, cybercrime is also happening more and more [1-2]. The first step in stopping a security attack is to look for signs of intrusion. Studies pay a lot of attention to security solutions like Firewall, Intrusion Detection System (IDS), Unified Threat Modeling (UTM), and Intrusion Prevention System (IPS). IDS can find attacks coming from many different systems and networks because it collects information and then looks at it for possible security holes [3]. The network-based IDS looks at the data packets that move through a network. It does this in two ways. Anomaly-based detection is still far behind signature-based detection, so it is still a major area of research [4-5]. The problem with anomaly-based intrusion detection is that it has to deal with new attacks for which there isn't enough information to figure out what's wrong. So, the system needs to be smart enough to tell which traffic is safe and which is dangerous or strange. Researchers have been looking into machine learning techniques for this purpose over the past few years [6]. IDS, on the other hand, is not the answer to all security problems. For example, IDS can't make up for weak mechanisms for identifying and authenticating users or for weak network protocols.

Intrusion detection was first looked into in 1980, and the first model of this kind was published in 1987 [7]. Even though businesses have spent a lot of money and done a lot of research in the last few decades, intrusion detection technology is still young and not very good [7]. Anomaly-based network IDS have not been as successful as signature-based network IDS. Signature-based network IDS have done well in the business world and are used by many technology-based companies around the world. Because of this, anomaly-based detection is a big area of research and development in the IDS field right now [8]. And key problems still need to be fixed before a large-

scale deployment of an anomaly-based intrusion detection system [8]. But there isn't much written about comparing how well intrusion detection works when supervised machine learning techniques are used [9]. Anomaly-based network IDS is a useful technology that can protect target systems and networks from malicious activities. Even though many anomaly-based network intrusion detection techniques have been written about in the past few years [8], security tools with anomaly detection capabilities are just starting to show up, and some important problems still need to be solved. Several techniques based on anomalies have been suggested, such as Linear Regression, Support Vector Machines (SVM), Genetic Algorithm, Gaussian mixture model, knearestneighbour algorithm, Naive Bayes classifier, and Decision Tree [3,5]. SVM is the most popular learning algorithm because it has been proven to work on many different types of problems [10]. One big problem with anomaly-based detection is that even though all of the proposed techniques can find new attacks, they all have a high rate of false alarms. The reason for this is how hard it is to make profiles of normal behaviour in real life by learning from training data sets [11]. Back propagation, which has been around since 1970 as the opposite of automatic differentiation [12], is often used to train Artificial Neural Networks (ANN) today.

The lack of a complete network-based data set [13] is one of the main problems with figuring out how well network IDS work. The KDD CUP 99 dataset [14] was used to test most of the proposed anomaly-based techniques found in the literature. In this paper, we used SVM and ANN, two machine learning techniques, on a popular benchmark dataset for network intrusion called NSLKDD [15].

1.1 MODEL OF SYSTEM

Fig.1 shows that the proposed system is made up of a feature selection algorithm and a learning algorithm. The job of the feature selection component is to find the most important features or attributes that link an instance to a certain group or class. The result from the feature selection component is used by the learning algorithm to gain the intelligence or knowledge it needs. The training dataset is used to teach the model and help it get smarter. Then, the learned intelligences are used on the testing dataset to see how well the model classified data it had never seen before.

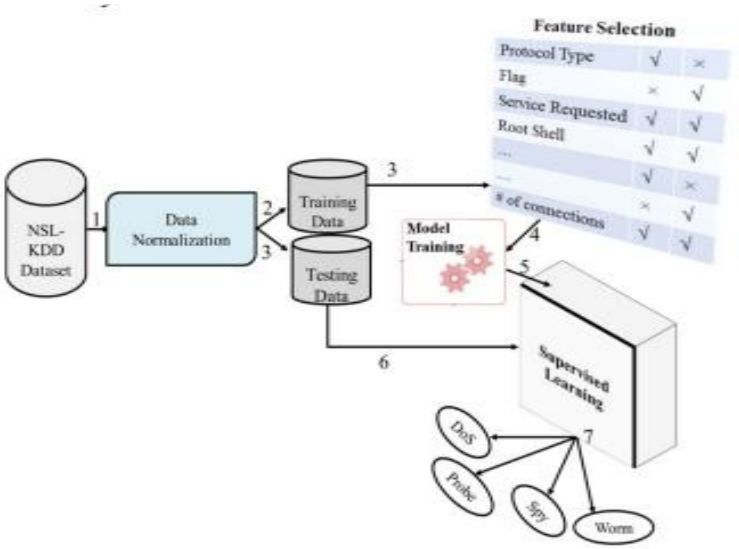


Fig 1: Proposed supervised machine learning classifier system

1.3MOTIVATION

The goal of this study is to find out how machine learning and deep learning algorithms can help predict plant and yield growth better and how the combination of these models works better than the ones that are already available, since there is no single paper that talks about the predictions made in this. Lastly, it's important to know and understand how these models can be different from each other in predicting data.

1.4 PROBLEM STATEMENT

The main problem with this is that it uses supervised machine learning and an environment-linked system to find network intrusions. So far, there hasn't been much progress in network intrusion detection using supervised machine learning with feature selection growth based on environmental variables.

1.5 OBJECTIVE

The main goals of this model were to: • Show what an Intrusion Detection System is and why it's important.

• Know how to use Snort IDS/IPS.

• To find out if someone got into a computer network without permission by looking for signs of bad behaviour in the network traffic.

1.6 SCOPE

This work's scope is

• Most intrusion detection systems use two main ways to find unauthorised people:
signature-based intrusion detection and anomaly-based intrusion detection

• Signature-based intrusion detection is meant to find possible threats by comparing network traffic and log data to known attack patterns.

• Anomaly-based intrusion detection is the opposite. It is made to find unknown attacks, like new malware, and adapt to them on the fly using machine learning. With machine learning, an intrusion detection system (IDS) can create trust models, which are baselines of trustworthy behaviour, and then compare new behaviour to verified trust models. When using an IDS based on "anomalies," there could be false alarms because normal network traffic that was not known before could be mistakenly seen as malicious activity.

3.1 ALGORITHMS:

3.1.1 Feature Selection

Feature selection is an important part of machine learning that helps to reduce the number of data dimensions. To find a reliable feature selection method, a lot of research has been done. Both the filter method and the wrapper method have been used to choose features. In the filter method, features are chosen based on how well they do in different statistical tests that look at how well they match up with the dependent variable or outcome variable. Wrapper method finds a subset of features by using the dependent variable to measure how useful a subset of features is. So, filter methods work with any machine learning algorithm, while the best feature subset in the wrapper method depends on the machine learning algorithm that was used to train the model. In the wrapper method, a subset evaluator uses all possible subsets and then uses a classification algorithm to convince classifiers from the features in each subset. The classifier looks at the subset of features that work best with the classification algorithm. The evaluator uses different search methods, such as depth-first search, random search, breadth-first search, and hybrid search, to find the subset. The filter method ranks all the features in the dataset by using an attribute evaluator and a ranker. Here, one low-ranking feature is left out at a time, and then the classification algorithm is used to see how well it can predict. The weights or ranks that ranker algorithms give are different from those that classification algorithms give. The wrapper method is good for testing machine learning, while the filter method is good for testing data mining because data mining has a lot of features.

A. Building Machine Intelligence

Learning models are made based on the best features found in the process of selecting features. An algorithm for machine learning is used to make the learning model. The chosen features are used to teach the algorithm how to work with the training dataset. In supervised machine learning, the class of each example in the training dataset is known. Depending on which machine learning algorithm is being used, the algorithm builds the learning model.

Support Vector Machine (SVM)

In SVM, the classifier is set by a separating hyper plane based on the type of problem and the available datasets. If the dataset has only one dimension, the hyper plane is a point. If the dataset has two dimensions, the hyper plane is a dividing line, as shown in Fig. 2. If the dataset has three dimensions, the hyper plane is a plane, and if it has more dimensions, it is a hyper plane. For a set of data that can be split into lines, the classifier or decision function will look like this:

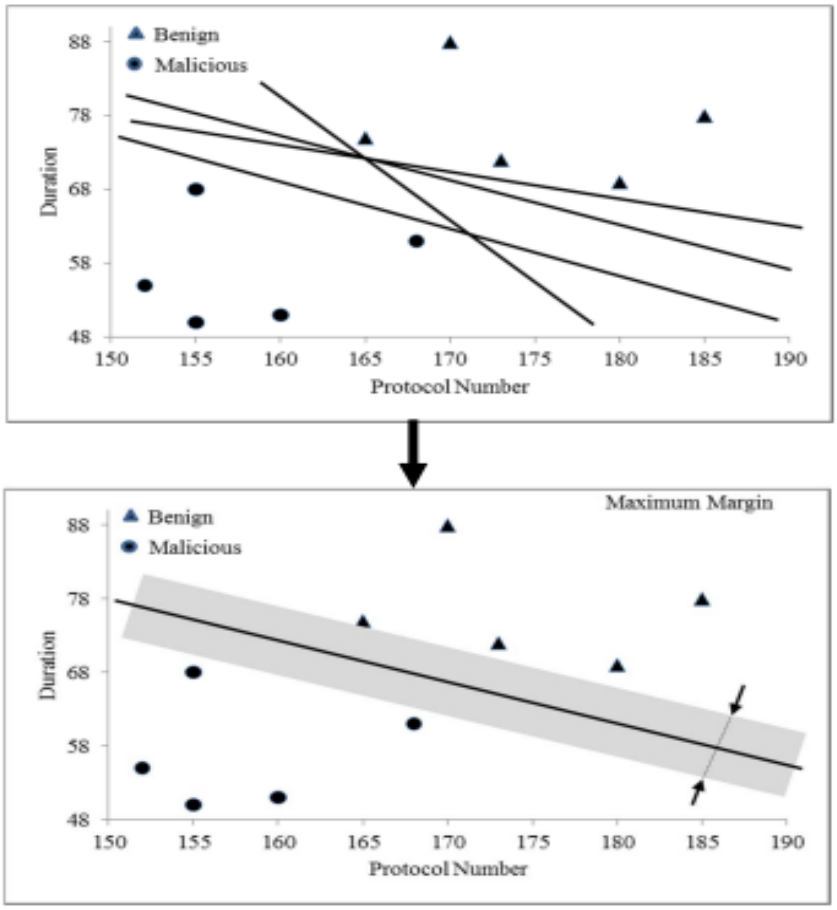


Fig 2: SVM classifier in two dimensional problem spaces

ax by c + += 0 (1)

For a given data points (x,y), the above decision function will classify the point in one class if ax + by ≥ c or it will categorize if ax + by < c. The equation of a line y=ax+b can be rewritten as y−ax−b=0 that can be represent using two vectors as below

$$\mathbf{w}\begin{pmatrix}-b\\-a\\1\end{pmatrix} \text{ and } \mathbf{x}\begin{pmatrix}1\\x\\y\end{pmatrix} \quad (2)$$

Which says we can write the linear equation of a line using two vectors as below?

$$\mathbf{w}^T\mathbf{x} = (-b)\times(1)+(-a)\times x+1\times y, \text{ or}$$
$$\mathbf{w}^T\mathbf{x} = y-ax-b \quad (3)$$

The reason of using the hyper plane equation wTx instead of y=ax+b is because it is easier to work in more than two dimensions with this notation and the vector w will always be normal to the hyper plane. Once the hyper plan with maximum margin has been found, this hyper plane can be used to make predictions [11]. The hypothesis function h will be

$$h(x_i) = \begin{cases} +1; \ if \ \ \mathbf{w}.\mathbf{x}+b \geq 0 \\ -1; \ if \ \ \mathbf{w}.\mathbf{x}+b < 0 \end{cases} \quad (4)$$

A. Artificial Neural Network (ANN)

Artificial Neural Network is another tool used in machine learning. As it name suggests, ANN is a system inspired by human brain system and replicate the learning system of human brain. It consists of input and output layers with one or more hidden layers in most cases as shown in Fig 3. The ANN uses a technique called back propagation to adjust the outcome with the expected result or class.

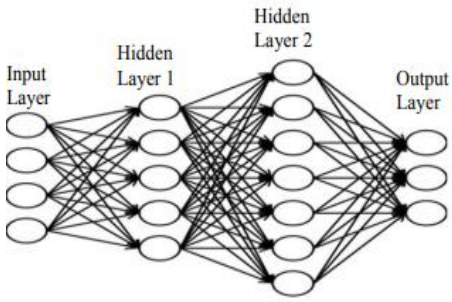


Fig 3: Artificial neural network showing the input, output and hidden layers

3.1.2 EXPERIMENTAL ANALYSIS OF THE SYSTEM

A. Feature Selection

The experiment carried out using Weka open source software suite popular for data mining and machine learning and consists of two parts. In the first part, we extracted most relevant features using different feature selection (FS) methods. In the wrapper method we used SVMclassification algorithm with cross-validation to avoid over fitting and under fitting problem. In the filter method a ranker algorithm is used to find the best result suitable for our proposed classifier. The training data we used from NSL-KDD dataset contains 25,191 labeled instances. Results of the feature selection experiment are shown in Table I.

TABLE I
RESULT OF FEATURE SELECTION

FS Technique	FS Type	Input Features	Output Features
Correlation Based	Wrapper	41	17
Chi-Square Based	Filter	41	35

Correlation-based feature selection found that 17 of the 41 features in the training dataset were the most important, while the Chi-Square algorithm kept 35 features that were more important to the final class. These 17 and 35 retained features were used to train the model with the training or seen dataset and to test the model with the unseen or testing dataset B. Using the training dataset, Weka's software suite builds four models using the features found in the "feature selection" part. For supervised machine learning to be used for classification, the model must first be trained using a training dataset. We used 25,191 labelled data points from 20% of the NSL-KDD dataset as training data. For each type of feature selection method, we used the SVM and ANN learning algorithm to train the model. So, we make four learning models, two of which use SVM and two of which use ANN. In the feature selection part, one of the two models built for each learning algorithm uses 17 features and the other uses 35 features. Then, 22,542 pieces of testing data from the NSL-KDD testing dataset were used to test these four trained models. The results are summed up in Table II, which is shown below:

TABLE II
RESULT OF CLASSIFICATION

Learning Type	Number of Features	Detection Accuracy
SVM	17	81.78%
SVM	35	82.34%
ANN	17	94.02%
ANN	35	83.68%

In Table III, we listed our results with recently published results in the literature. While comparing the performance of the proposed model with the others works, we picked works having hypothesis of comparable aspects related to learning algorithm and benchmarking datasets. But there are other aspects like attribute reduction, number of instances, the number layers and learning rates used. The detection success rate of the proposed model is also compared with other existing models in Table III as below

TABLE III
PERFORMANCE COMPARISON WITH EXISTING MODELS

Learning Type	Our Model Accuracy	Existing Model	Existing Model
SVM	82.34%	92.84% [16]	69.52% [17]
ANN	94.02%	81.2% [18]	77.23% [19]

3.2 ARCHITECTURE/FRAMEWORK:

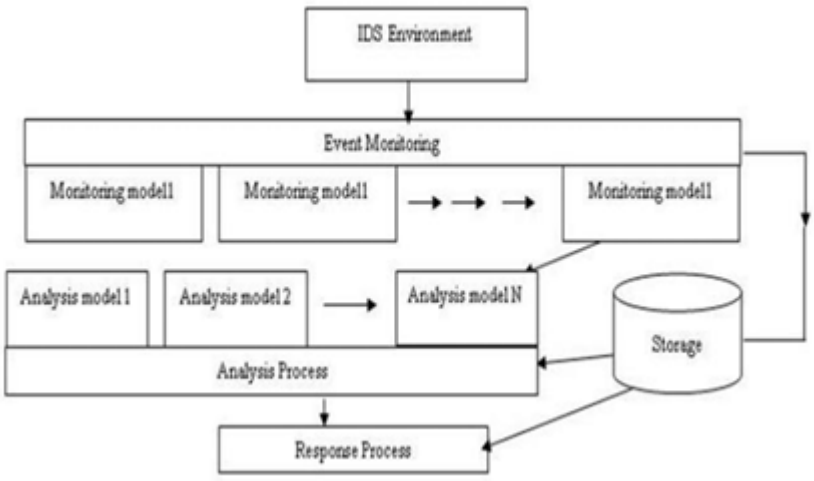


Fig.4 Common Intrusion Detection Framework Architecture

3.3 ALGORITHM AND PROCESS DESIGN:

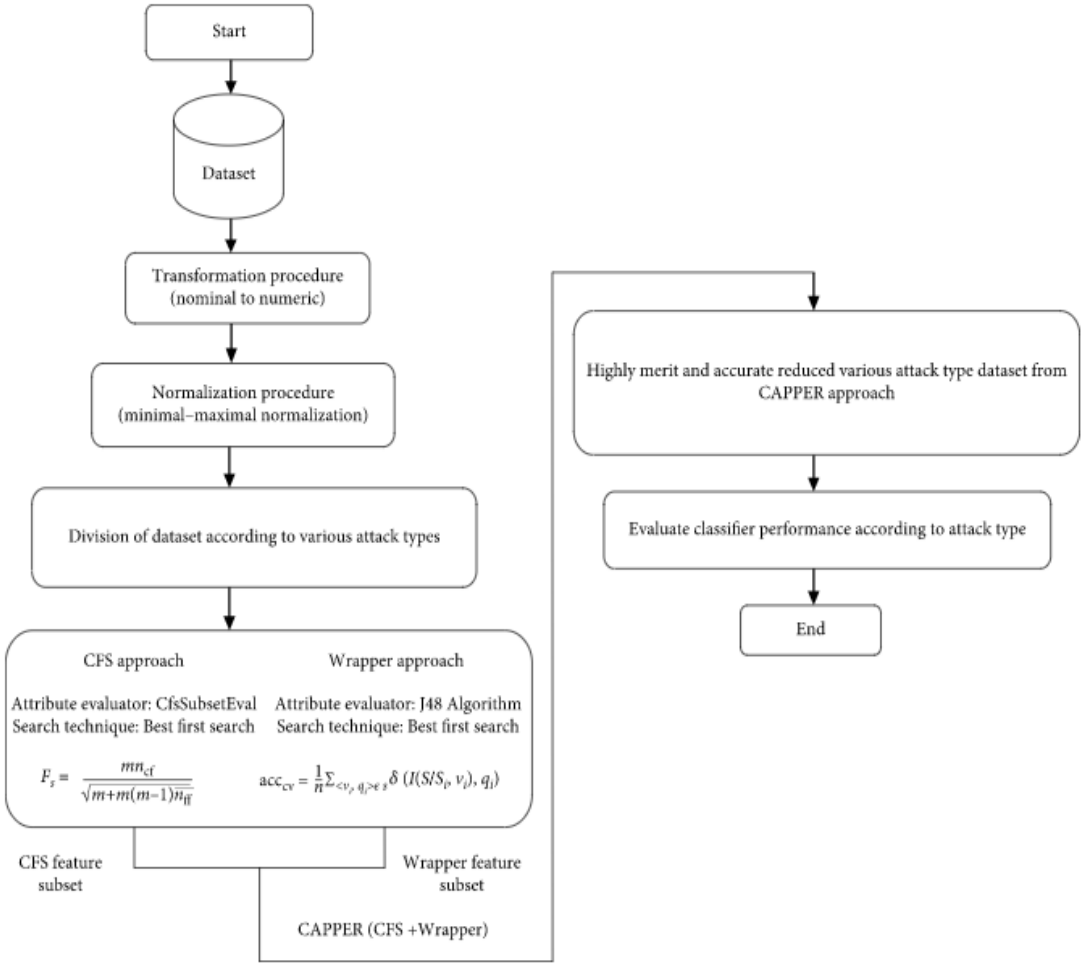


Fig.5. **PROCESS DESIGN**

4 IMPLEMENTATION AND OUTCOMES:

as expected, the performance of snort was found to be dependent on its support-ing hard-ware components (cpu, memory, nicetc). in the virtual scenarios, snort was found to be less accurate for all categories of background traffic. conversely, the performance of snort improved when run natively on its host machine by utilizing all of the available hardware resources. the statistics for percentages of dropped packets are shown in fig. 18. re-source constraints in the virtual machine have affected the overall performance of snort resulting in a high number of packets dropped and a reduction in alerts logged.

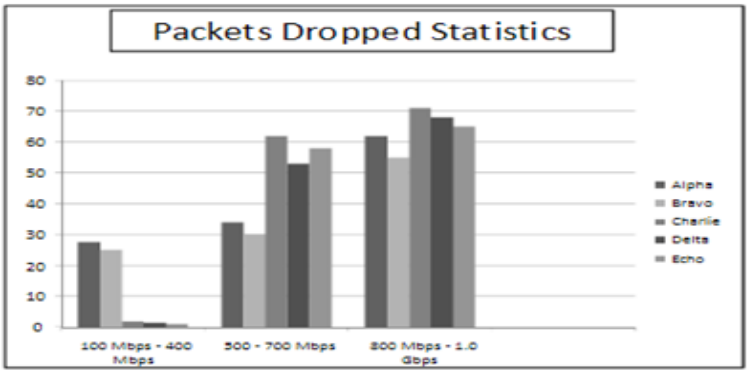


Fig.6. Packets Dropped

4.1 PERFORMANCE METRICS:

There are many different classification metrics for IDS, and some of them have more than one name. This picture shows the confusion matrix for a two-class classifier, which can be used to judge how well an IDS works. Each column in the matrix shows the instances that belong to a predicted class, and each row shows the instances that belong to an actual class.

IDS are usually judged by how well they do the following standard things:

• Rate of True Positives (TPR): It is found by dividing the number of attacks that were predicted right by the total number of attacks. If all intrusions are found, the TPR is 1, which is a very rare situation for an IDS. The Detection Rate (DR) or the Sensitivity is another name for TPR. The TPR can be written in numbers as

TPR=TPTP+FNTPR=TPTP+FN • False Positive Rate (FPR): It is the number of normal situations that were wrongly labelled as attacks compared to the total number of normal situations.

FPR=FPFP+TN

• False Negative Rate (FNR): A false negative is when a detector doesn't notice something out of the ordinary and labels it as normal. Mathematically, the FNR can be written as:

FNR=FNFN+TP

FNR=FNFN+TP

Rate of correct classification (CR) or accuracy: The CR measures how well the IDS can tell if traffic is acting normally or not. It is shown as the ratio of the number of cases where the prediction was right to the total number of cases:

Accuracy=TP+TNTP+TN+FP+FN

4.2 OUTCOME: The Artificial Neural Network (ANN)-based machine learning with wrapper feature selection does a better job of classifying network traffic than the support vector machine (SVM) method. supervised machine learning techniques like SVM and ANN are used to classify network traffic from the NSL-KDD dataset in order to measure performance. Comparative studies show that the proposed model is better at detecting intrusions than other models that are already out there.

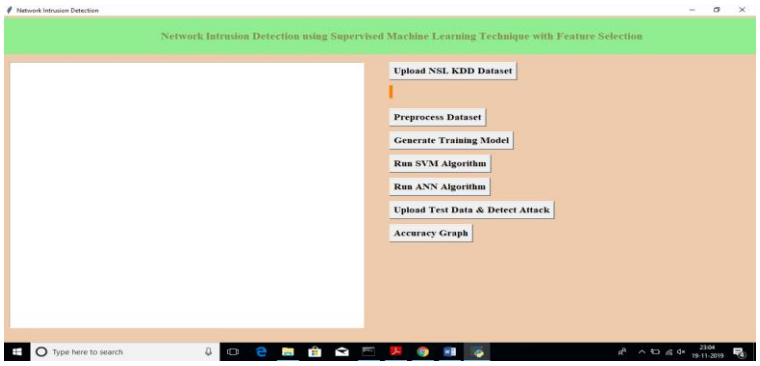
4.3 

Fig.7NSL KDD Dataset'

In above figure click on 'Upload NSL KDD Dataset' button and upload dataset

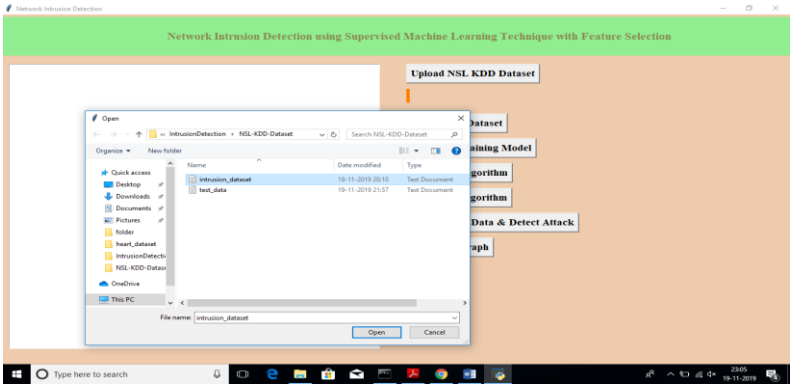


Fig.8. intrusion_dataset.txt'

In above figure I am uploading ‘intrusion_dataset.txt’ file, after uploading dataset will get below figure

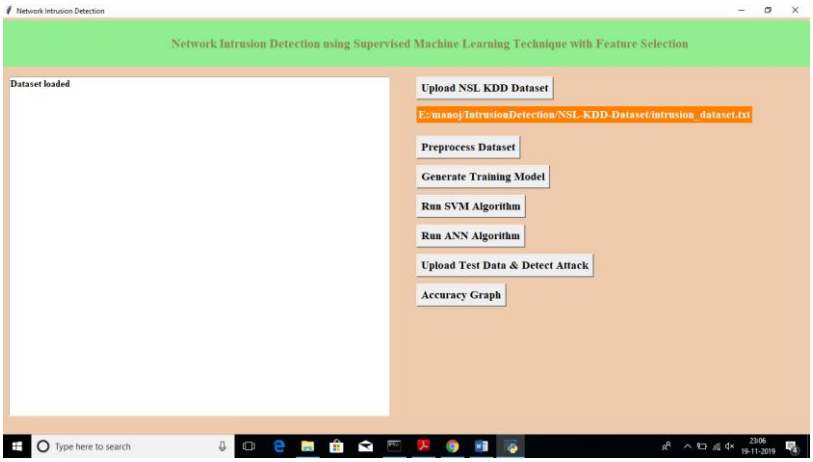


Fig.9. Pre-processing of Dataset’

Now click on ‘Pre-process Dataset’ button to clean dataset to remove string values from dataset and to convert attack names to numeric values

Fig.10. string values removed

After pre-processing all string values removed and convert string attack names to numeric values such as normal signature contains id 0 and anomaly attack contains signature id 1.

Now click on ‘Generate Training Model’ to split train and test data to generate model for prediction using SVM and ANN

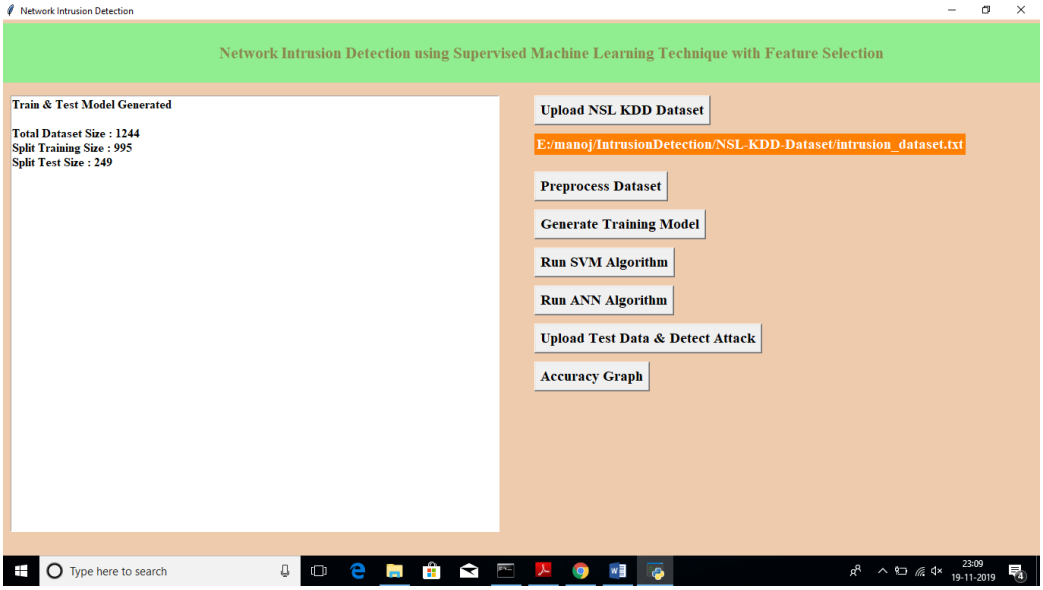


Fig.11. dataset details

In above figure we can see dataset contains total 1244 records and 995 used for training and 249 used for testing. Now click on 'Run SVM Algorithm' to generate SVM model and calculate its model accuracy

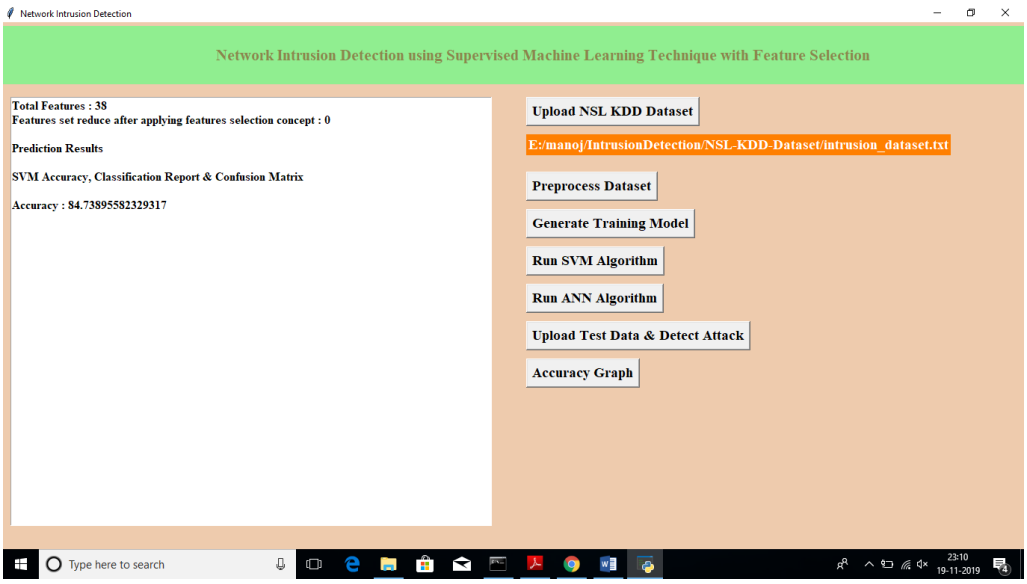


Fig.12. SVM we got 84.73% accuracy

In above figure we can see with SVM we got 84.73% accuracy, now click on 'Run ANN Algorithm' to calculate ANN accuracy

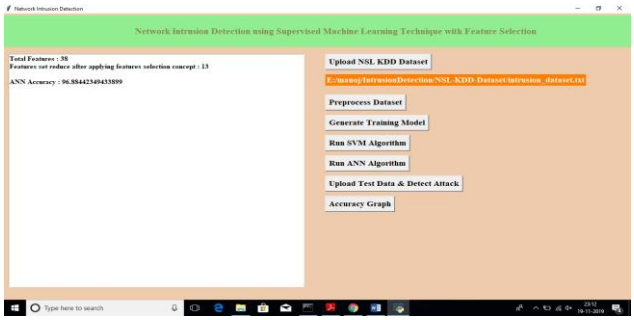


Fig.13. 96.88% accuracy

In above figure we got 96.88% accuracy, now we will click on 'Upload Test Data & Detect Attack' button to upload test data and to predict whether test data is normal or contains attack. All test data has no class either 0 or 1 and application will predict and give us result. See below some records from test data

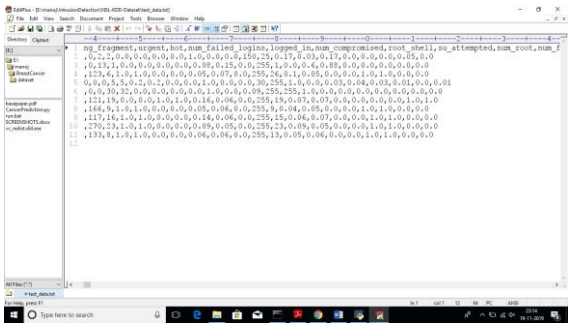


Fig.14. either '0' or '1'

In above test data we don't have either '0' or '1' and application will detect and give us result

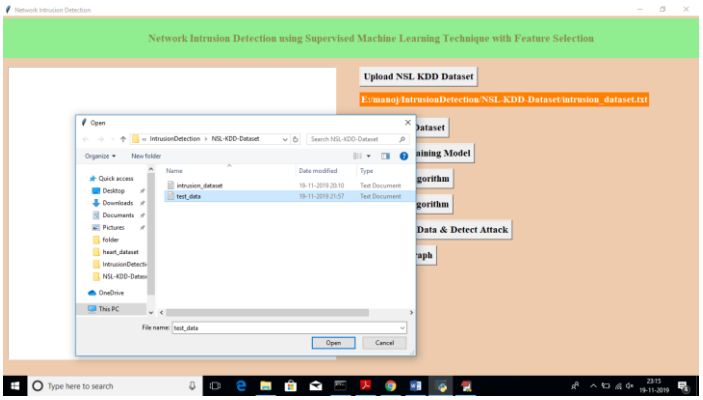


Fig.15. uploading 'test_data'

In above figure I am uploading ‘test_data’ file which contains test record, after prediction will get below results

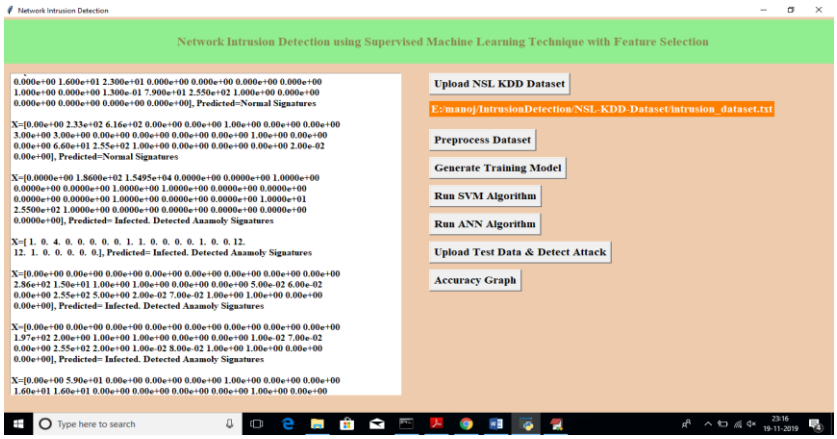


Fig.16. predicted results

In above figure for each test data we got predicted results as ‘Normal Signatures’ or ‘infected’ record for each test record. Now click on ‘Accuracy Graph’ button to see SVM and ANN accuracy comparison in graph format

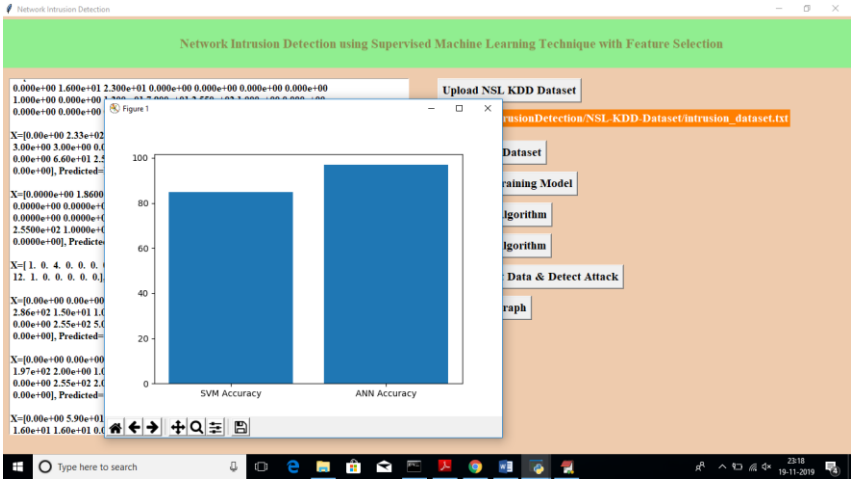


Fig.17. ANN got better accuracy

From the above graph, we can see that ANN is more accurate than SVM. The x-axis shows the name of the algorithm, and the y-axis shows how accurate that algorithm is.

Extension Outcomes:

In this project, the author used Traditional SVM algorithms, which are already in use, and an Artificial Neural Network, which was created for this project (ANN). SVM will be trained on a dataset without optimising its features, while ANN will filter the dataset with different numbers of input and hidden layers to find the most important features or to optimise the dataset's features. Because of this optimization of the dataset's features, ANN will make more accurate predictions.

After ANN was successful, a new version called Convolution2D neural network was made (CNN). CNN is better than ANN because it will use more input and hidden layers to filter datasets and get more optimised features.

Important facts about CNNs: 1) CNNs are based on the discovery that nerve cells in the visual cortex have orientation-selective local sensitivity.

2) They are a neural network with more than one layer.
3) They automatically pull out important features.
4) They are a feed-forward network that can use a dataset to find topological features.
5) They can see patterns in images made up of pixels without much preprocessing.
6) They are incredibly powerful because they can easily spot patterns that are very different from one another. e.g., attack NSL dataset.
7) CNNs are taught with a version of the back-propagation algorithm.
8) CNNs are based on the neuronal cells in the visual cortex, which makes CNNs and watches for specific features possible.
So, as an extension, we've tried out the CNN algorithm and found that it works better than CNN. We then used the CNN model on test data to predict both NORMAL and malicious signatures.

FIGURE SHOTS

Run the project in the same way as before, but I added a new algorithm called CNN. To see the results, double-click the "run.bat" file.

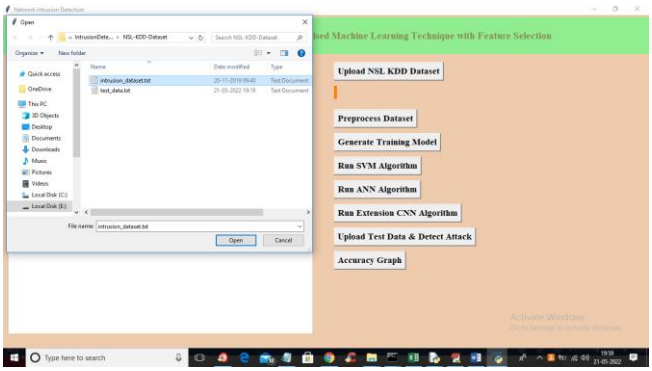


Fig.18. uploading dataset

In above figure uploading dataset and then click on all buttons one by one to get below output

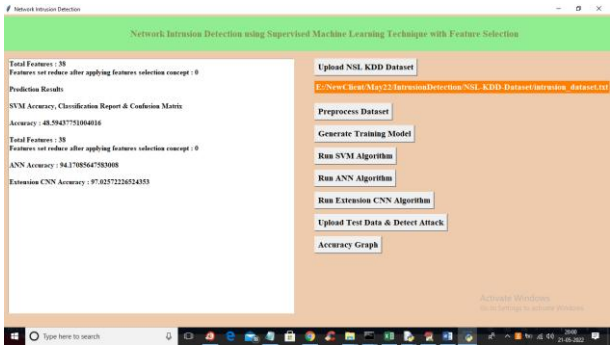


Fig.19. SVM we got 48% and with ANN 94% accuracy

In above figure with SVM we got 48% accuracy and with ANN we got 94% accuracy and with extension CNN we got 97% accuracy and now click on 'Upload Test Data & Detect Attack' button to upload test data and get below output

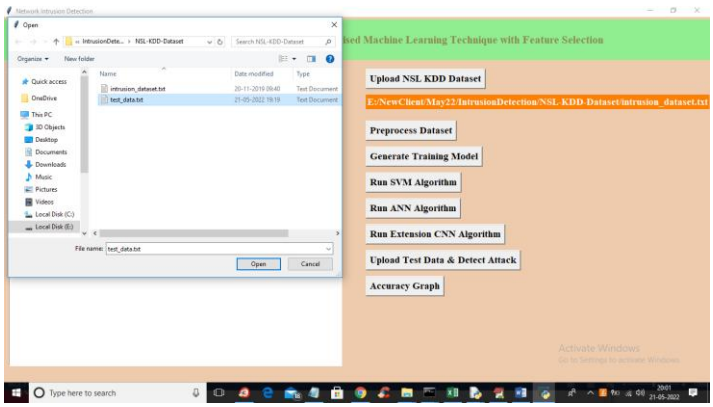


Fig.20. uploading test_data.txt

In above figure selecting and uploading test_data.txt file and then click on ‘Open’ button to get below output

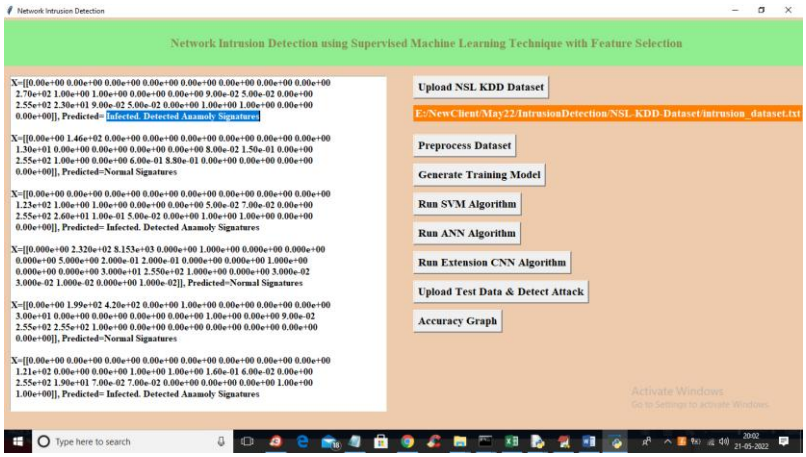


Fig.21. TEST data

In above figure square bracket contains TEST data and then after square bracket we can see predicted classes as INFECTED or NORMAL and now click on ‘Accuracy Graph’ button to get below output

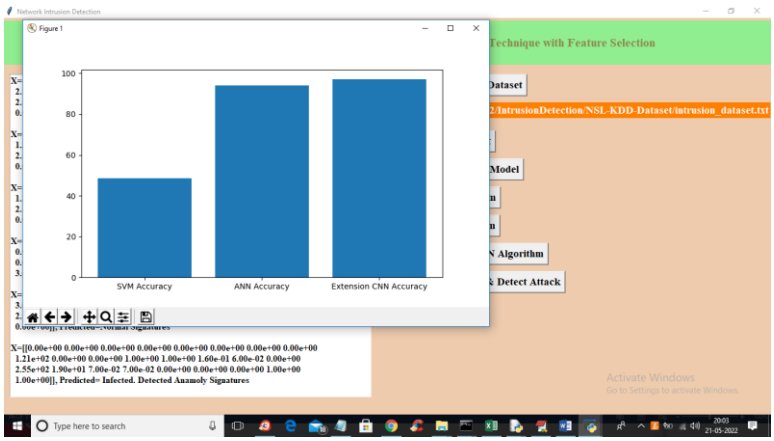


Fig.22. algorithm names vs accuracy

In above graph x-axis represents algorithm names and y-axis represents accuracy and in all algorithms extension got high accuracy

V. CONCLUSION

In this paper, we looked at different machine learning models. To find the best model, we used different machine learning algorithms and different ways to choose features. The analysis of the results shows that the model made with ANN and wrapper feature selection did a better job of correctly classifying network traffic than any other model, with a 94.02% detection rate. We think that these results will help researchers build a system that can find both known and new attacks. Today's intrusion detection systems can only find attacks that are already known to them. Finding new attacks or "zero-day" attacks is still a problem.

BIBLIOGRAPHY:

[1] H. Song, M. J. Lynch, and J. K. Cochran, "A macro-social exploratory analysis of the rate of interstate cyber-victimization," American Journal of Criminal Justice, vol. 41, no. 3, pp. 583–601, 2016.

[2]P. Alaei and F. Noorbehbahani, "Incremental anomaly-based intrusion detection system using limited labeled data," in Web Research (ICWR), 2017 3th International Conference on, 2017, pp. 178–184.

[3] M. Saber, S. Chadli, M. Emharraf, and I. El Farissi, "Modeling and implementation approach to evaluate the intrusion detection system," in International Conference on Networked Systems, 2015, pp. 513–517.

[4]M. Tavallaee, N. Stakhanova, and A. A. Ghorbani, "Toward credible evaluation of anomaly-based intrusion-detection methods," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 40, no. 5, pp. 516–524, 2010.

[5]A. S. Ashoor and S. Gore, "Importance of intrusion detection system (IDS)," International Journal of Scientific and Engineering Research, vol. 2, no. 1, pp. 1–4, 2011.

[6]M. Zamani and M. Movahedi, "Machine learning techniques for intrusion detection," arXiv preprint arXiv:1312.2177, 2013.

[7]N. Chakraborty, "Intrusion detection system and intrusion prevention system: A comparative study," International Journal of Computing and Business Research (IJCBR) ISSN (Online), pp. 2229–6166, 2013.

[8] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," computers & security, vol. 28, no. 1–2, pp. 18–28, 2009.

[9]M. C. Belavagi and B. Muniyal, "Performance evaluation of supervised machine learning algorithms for intrusion detection," Procedia Computer Science, vol. 89, pp. 117–123, 2016.

[10]J. Zheng, F. Shen, H. Fan, and J. Zhao, "An online incremental learning support vector machine for large-scale data," Neural Computing and Applications, vol. 22, no. 5, pp. 1023–1035, 2013.

[11]F. Gharibian and A. A. Ghorbani, "Comparative study of supervised machine learning techniques for intrusion detection," in Communication Networks and Services Research, 2007. CNSR'07. Fifth Annual Conference on, 2007, pp. 350–358.

[12]J. Schmidhuber, "Deep learning in neural networks: An overview," Neural networks, vol. 61, pp. 85–117, 2015.

[13]N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in Military Communications and Information Systems Conference (MilCIS), 2015, 2015, pp. 1–6.

[14] T. Janarthanan and S. Zargari, "Feature selection in UNSW-NB15 and KDDCUP'99 datasets," in Industrial Electronics (ISIE), 2017 IEEE 26th International Symposium on, 2017, pp. 1881–1886.

[15]L. Dhanabal and S. P. Shantharajah, "A study on NSL-KDD dataset for intrusion detection system based on classification algorithms," International Journal of Advanced Research in Computer and Communication Engineering, vol. 4, no. 6, pp. 446–452, 2015.

[16]A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," in Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS), 2016, pp. 21–26.

[17]M. Panda, A. Abraham, and M. R. Patra, "Discriminative multinomial naive bayes for network intrusion detection," in Information Assurance and Security (IAS), 2010 Sixth International Conference on, 2010, pp. 5–10.

[18]B. Ingre and A. Yadav, "Performance analysis of NSL-KDD dataset using ANN," in Signal Processing And Communication Engineering Systems (SPACES), 2015 International Conference on, 2015, pp. 92–96.

[19] L. M. Ibrahim, D. T. Basheer, and M. S. Mahmod, "A comparison study for intrusion database (Kdd99, Nsl-Kdd) based on self organization map (SOM) artificial neural network," Journal of Engineering Science and Technology, vol. 8, no. 1, pp. 107–119, 2013.