Intrusion Detection System Using Deep Learning

Prof. Smita Jawale
Computer Engineering department
VCET College
Vasai, India
smita.jawale@vcet.edu.in

Shruti Sankhe
Computer Engineering department
VCET College
Vasai, India
shrutisankhe4@gmail.com

Rutuja Parab
Computer Engineering department
VCET College
Vasai, India
Rutujarockstarparab555@gmail.com

Nehab Shaikh
Computer Engineering department
VCET College
Vasai, India
nehabkalim@gmail.com

***Abstract* — Nowadays, Intrusion Detection System (IDS) has become an essential layer in all the newest Ict entity due to an urge towards cyber safety in the day-to-day world. Reason including uncertainty in the finding of types of attacks and doubled the complexity of sophisticated cyber-attacks, IDS calls for the need of integration of Deep Neural Network (DNNs). In this paper, DNNs have been used to foretell the attacks on Network Intrusion Detection Entity (N-IDS). A DNN with 0.1 rate of learning is applied and is run for thousand number of epochs and KDDCup-'99' dataset has been used for training and benchmarking the network. For comparison purposes, the training is done on the same dataset with some other classical machine learning algorithmic program and DNN of layers ranging from 1 to 5. The results were compared and concluded that a DNN of 3 layers has superior performance over all the other classical machine learning algorithms.**

***Index Terms*—Intrusion detection, deep neural networks, machine learning, deep learning**

I. Introduction

In the modern world, the fast-paced technological advancements have encouraged every organization to adopt the integration of information and communication technology (ICT). Hence creating an environment where every action is routed through that system making the organization vulnerable if the security of the ICT system is compromised. Therefore, this call for a multilayered detection and protection scheme that can handle truly novel attacks on the system as well as able autonomously adapt to the new data. There are multiple systems that can be used for shielding such ICT systems from vulnerabilities, namely anomaly detection and IDSs. A demerit of anomaly-detection systems is the complexity involved in the process of defining rules. Each protocols being analyzed must be defined, implemented and tested for accuracy. Another pitfall relating to anomaly detection is that harmful activity that falls within usual usage pattern is not recognized. Therefore the need for an IDS that can adapt itself to the recent novel attacks and can be trained as well as deployed by using datasets of irregular distribution becomes indispensable. Intrusion Detect Systems (IDSs) are a range of cybersecurity based technology initially developed to detect vulnerabilities and exploits against a target host. The sole use of the IDS is to detect threats. Therefore it is located out-of-band on the infrastructure of the network and is not in the actual real-time communication passage between the sender and receiver of data. Instead, they solutions will often make use of a TAP or SPAN ports to analyze the inline traffic stream's copy and will try to predict the attack based on a previously trained algorithm, hence making the need of a human intervention trivial. The IDS has three methods for detecting attacks; Signature-based detection, Anomaly-based detection, and Hybrid-based detection. The signature-based detection is designed to detect known attacks by using signatures of those attacks. It is an effective method of detecting known attacks that are preloaded in the IDS database. Therefore, it is often considered to be much more accurate at identifying an intrusion attempt of known attack. However, new types of attack cannot be detected as its signature is not presented; the databases are frequently updated in

order to increase their effectiveness of detections. To overcome this problem Anomaly-based detection that compares current user activities against predefined profiles is used to detect abnormal behaviors that might be intrusions. Anomaly-based detection is effective against unknown attacks or zero-day attacks without any updates to the system. However, this method usually has high false positive rates. Hybrid-based detection is a combination of two or more methods of intrusion detection in order to overcome the disadvantages in the single method used and obtain the advantages of two or more methods that are used. Many researches proposed machine learning algorithm for intrusion detection to reduce false positive rates and produce accurate IDS. However, to deal with Big Data, the machine learning traditional techniques take a long time in learning and classifying data. The KddCup99 dataset are tested in this study. Towards the end, the sections are organized as followe: Section II reviews the work related to IDS, different deep neural networks and some discussions about KDDCup-'99' dataset that was published. Section III takes an in-depth look at Deep Neural Networks (DNN) and the applications of ReLU activation function. Section IV analyses the dataset used in this paper, explains the shortcoming of it and evaluates the final results. Section V concludes and states a plausible workflow into the future of this research work.

II. LITERATURE REVIEW

2.1 Domain Explanation

This section presents an extensive study over the various intrusion detection classifier techniques and other techniques. A number of research papers regarding to intrusion detection are discussed below and are widely classified into i) papers related to Neural network ii) papers related to Support vector machine iii) papers related to K-means classifier iv) papers related to hybrid technique and v) paper related to other detection techniques. A IDS is developed by combining the two approaches in one system. The hybrid IDS is obtained by combining packet header anomaly detection (PHAD) and network traffic anomaly detection (NETAD) which are anomaly based IDSs with the misuse-based IDS Snort which is an open-source project.

1. Deep Neural Network (DNN)

While traditional machine learning algorithms are linear, deep neural networks are stacked in increasing hierarchy of complexity as well as abstraction. Each layer applies a nonlinear transformation onto its input and creates a statistical model as output from what it learns. In simple terms, the input layer is received by the input layer and passed onto the first hidden layer. These hidden layers perform mathematical computations on our inputs. One of the challenges in creating neural networks is deciding the hidden layers' count and the count of the neurons for each layer.

2. Neural Network Based Intrusion Detection

A brief review of two techniques related with neural network based intrusion detection is discussed in this section. In 2009 a lot of papers have been presented to represent the neural network based intrusion detection a comprehensive literature survey on machine learning based ID with KDDCup 99 dataset was conducted. After the challenge, most of the published results of KDDCup 99 have used several feature engineering methods for dimensionality reduction. While few studies employed custom-built datasets, majority used the same dataset for newly available machine learning classifiers. These published results are partially comparable to the results of the KDDCup 99 contest. In, the classification model consists of two-stages: i) P-rules stage to predict the presence of the class, and ii) N-rules stage to predict the absence of the class. This performed well in comparison with the aforementioned KDDCup 99 results except for the user-to-root ('U2R') category. In, the significance of feature relevance analysis was investigated for IDS with the most

widely used dataset, KDDCup 99. For each feature they were able to express.

3. HOST-BASEDINTRUSION DETECTION SYSTEMS (HIDS)

Host-based intrusion detection systems (HIDSs) are applications that function on databases gathered from individual computer systems. This vantage point lets an HIDS to analyze activities on the host it monitors at a elevated level of detail; it can repeatedly decide which processes and/or users are entailed in malicious activities. Furthermore, in contrast with NIDSs, HIDSs are privy to the outcome of an attempted attack since they can straight get into use and monitor the data files and scheme processes targeted by these attacks.

Alternatively, HIDSs can exhaust databases sources of two types, functioning Scheme audit trails, and Entity logs. functioning Entity audit trails are generally created at the innermost (kernel) level of the functioning system, and are hence more detailed and greater saved than Scheme logs. Scheme logs are much less obtuse and much smaller than audit trails, and are normally far easier to comprehend.

In, frequency distribution based feature engineering approach with machine learning algorithms was explored to handle the zero-day and stealth attacks in Windows OS. In, an ensemble approach for HIDS was proposed using language modeling to reduce the false alarm rates which is a drawback in classical methods.

2.2 Existing System

An intrusion detection system (IDS) is a device that monitors or systems for malicious activity or policy violations. Any intrusion activity or violation is typically reported either to an administrator or collected centrally using a (SIEM) system. A SIEM system combines outputs from multiple sources and uses techniques to distinguish malicious activity from false alarms. IDS types range in scope from single computers to large networks. The most common classifications are network intrusion detection systems (NIDS) and (HIDS). A system that monitors important operating system files is an example of an HIDS, while a system that analyses incoming network traffic is an example of an NIDS. It is also possible to classify IDS by detection approach. The most well-known variants are (recognizing bad patterns, such as) and (detecting deviations from a model of "good" traffic, which often relies on). Another common variant is reputation-based detection (recognizing the potential threat according to the reputation scores).

III. Analysis

A. Datasets Description

The DARPA's program for ID evaluation of 1998 was managed and prepared by Lincoln Labs of MIT. The main objective of this is to analyze and conduct research in ID. A standardized dataset was prepared, which included various types of intrusions which imitated a military environment and was made publicly available. The KDD intrusion detection contest's dataset of 1999 was a well-refined version of this ReLu has turned out to be more efficient and have the A detailed report and major shortcomings of the provided syn- thetic data set such as KDDCup-'98' and KDDCup-'99" were discussed by The main condemnation was that they failed to validate their data set a simulation of real-world network traffic profile. Irrespective of all these criticisms, the dataset of KDDCup-'99' has been used as an effective dataset by many researchers for benchmarking the IDS algorithms over the years. In contrast to the critiques about the creation of the dataset, has revealed a detailed analysis of the contents, identified the non-uniformity and simulated the artifacts in the simulated network traffic data. The reasons behind why the machine learning classifiers have limited capability in identifying the attacks that belong to the content categories R2L, U2R in KDDCup-'99' datasets have been discussed by They have concluded that it is not possible to get acceptable detection rate using classical ML algorithms.

It failed in detecting dos and probing category but contrasting performing better than the detection of R2L and U2R.

3.1 Types Of Attacks

- **Denial-of-Service-Attack(DoS):** Intrusion where a per- son aims to make a host inaccessible to its actual purpose by briefly or sometimes permanently disrupting services by flooding the target machine with enormous amounts of requests and hence overloading the host.
- **User-to-Root-Attack (U2R):** A category of commonly used maneuver by the perpetrator start by trying to gain access to a user's pre-existing access and exploiting the holes to obtain root control.
- **Remote-to-Local-Attack (R2L**): The intrusion in which the attacker can send data packets to the target but has no user account on that machine itself, tries to exploit one vulnerability to obtain local access cloaking themselves as the existing user of the target machine.
- **Probing-Attack:** The type in which the perpetrator tries to gather information about the computers of the network and the ultimate aim for doing so is to get past the firewall and gaining root access.
- KDDCup-'99' set is classified into the following three groups: Basic features: Attributes obtained from a connection of TCP/IP comes from this group. implicitly delaying the 15 detection. Traffic features: Features computed w.r.t. a window of time is categorized under this group. This can be further subdivided into 2 groups:
- **"Same host" features:** The connections that has iden- tical end host as the connection under consideration for the continuously 2 seconds fall into this category and serves the purpose of calculating the statistics of protocol behaviour, etc.
- **"Same service" features:** The connections that are only having identical services to the present connection for the last two seconds fall under this category.
- **Content features:** Generally probing attacks and DoS attacks have at least some kind of frequent sequential intrusion patterns unlike R2L and U2R attacks. This is due to the reason that they involve multiple connections to a single set of a host(s) under short span of time while the other 2 intrusions are integrated into the packets of data partitions in which generally only one connection is involved. For the detection of these types of attacks, we need some unique features by which we will be able to search for some irregular behaviour. These are called content features.

3.2 Interface Requirement

A. Identifying network parameters

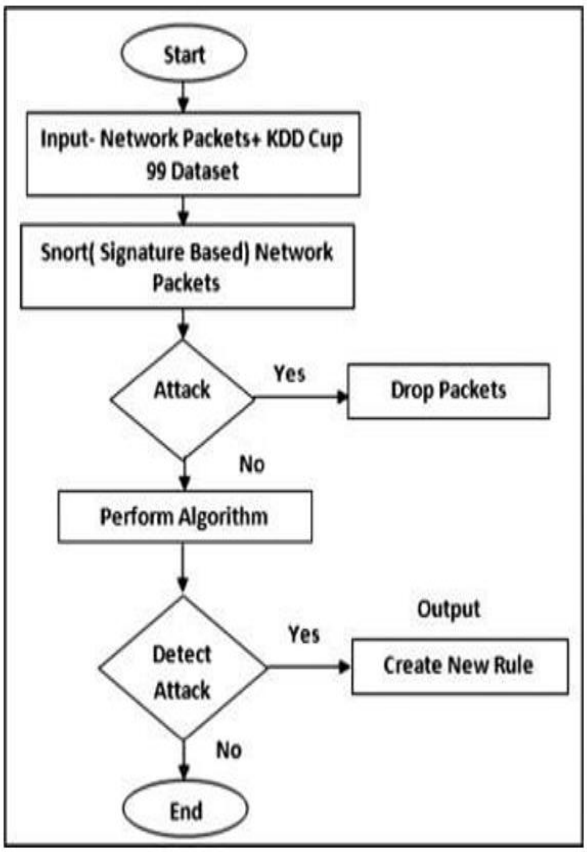
Hyper-tuning of parameters to figure out the optimum set of parameters to achieve the desired result is all by itself a separate field with plenty of future scope for research. In this paper, the learning is kept constant at 0.01 while the other parameters where optimized. The count of the neurons in a layer was experimented by changing it over the range of 2 to 1024. After that, the count was further increased to 1280 but didnt yield any appreciable increase in accuracy. Therefore the neuron count was tuned to 1024.

B. Identifying network structures

Conventionally, increasing the count of the layers results in better results compared to increasing the neuron count in a layer. Therefore, the following network topologies were used in order to scrutinize and conclude the optimum network structure for our input data.

IV. Design

4.1 Activity Diagram Of IDS



V. Proposed Architecture

An overview of proposed DNNs architecture for all use cases is showcased in. This comprises of a hidden-layer connote of 5 and an output-layer. The input-layer consists of 41 neurons. The neurons in input-layer to hidden-layer and hidden to output-layer are associated completely. Back-propagation mechanism is utilized to direct the DNN networks. The proposed network is made of fully associated layers, bias layers and dropout layers to make the network more robust.

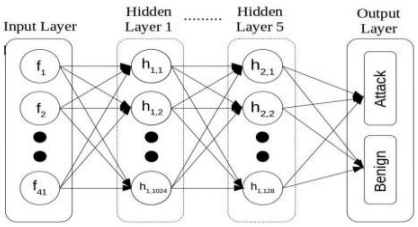


Fig 1 Proposed Architecture

5.1 Network Structure Information

TABLE I
NETWORK STRUCTURE INFORMATION

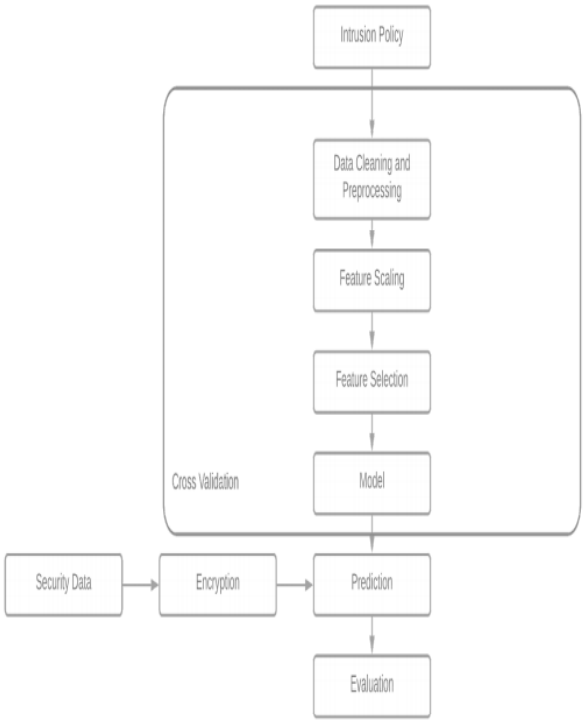
Layer (type)	Output Shape	Param
Dense-1 (Dense)	(NIL, 1024)	43008
Dropout-1 (Dropout)	(NIL, 1024)	0
Dense-2 (Dense)	(NIL, 768)	787200
Dropout-2 (Dropout)	(NIL, 768)	0
Dense-3 (Dense)	(NIL, 512)	393728
Dropout-3 (Dropout)	(NIL, 512)	0
Dense-4 (Dense)	(NIL, 256)	131328
Dropout-4 (Dropout)	(NIL, 256)	0
Dense-5 (Dense)	(NIL, 128)	32896
Dropout-5 (Dropout)	(NIL, 128)	0
Dense-6 (Dense)	(NIL, 1)	129
Activation-1 (Activation)	(NIL, 1)	0

Input and hidden layers: This layer consists of 41 neurons. These are then fed into the hidden layers. Hidden layers use ReLU as the non-linear activation function. Then weights are included to feed them send to the next hidden layer. The neuron connote in each hidden layer is cut steadily from the 1st to the output to make the outputs more correct and at the same time reducing the algorithmic cost.

Regularization: To make the complete process effective and time-saving, Dropout (0.01). The operate of the dropout is to unplug the neurons randomly, production the exemplar more robust and secondly preventing it from over-fitting the training set.

Output layer and classification: The out layer consists only of two neurons Attack and Benign. Since the 1024 neurons from the preceding layer must be converted into just 2 neurons, a sigmoid activation operate is used. Due to the nature of the sigmoid function , I t reverts only two outputs, secondly favouring the binary classification that was meant in this paper.

VI. Implementation



Data Cleaning And Preprocessing:

The duplicates have already been removed as NSL KDD dataset is already standardised. The nan and Infinity values are replaced with zilch initially. Preprocessing operation is done on the dataset as the dataset contains numerical and non-numerical values. One-Hot Encoding is utilized for this operation. An integer matrix indicating the values of the categorical options is an input to the One Hot Encoder. This will convert all the categorical options to their corresponding binary options out of which one will be active at a time. The dataset is then divided into four parts as said by the attacks(U2R,Probe,DOS,R2L) which required to be classified.

Features Scaling:

Featuring scaling is performed to steer distinct of options which have big values as this will bear on the final result. Classic Scaler is utilized to compose this operation. In Classic Scaler the average for a option is calculated and then the intend is subtracted from the common charge of the option and the result is divided by the Standard Deviation. The standard deviation will be 1 after each feature is scaled.

Feature Selection And Model :

It's the process in which are immaterial and unnecessary options are eliminated with minimal databases loss. Subsets of the options are choosen which fully serves all the options in the dataset in terms of accuracy and other metrics. It's additionally possible that over there is a correlation between options when a big number of options are present. Option choice also supports to get rid of this problem. We have utilized Recursive option Elimination(RFE) to compose this operation. We plan the graph for the accuracy against the number of options and based upon that we choose the optimal number of options for each of the attacks. Here, we have created two models: Decision Trees, Random Forest. Both machine acquiring models are created for all 4 types of attack i.e U2R, R2L, DoS and This exemplar is utilized on the dataset containing any feature(123) and additionally separately for the features(13) choosen after option choice operation.

VII. Result And Evaluation

We calculate four metrics on the basis of which we assess the performance of our Intrusion Detection System. Given below is the confusion matrix which is a collection of four values: TP, TN, FP, FN.

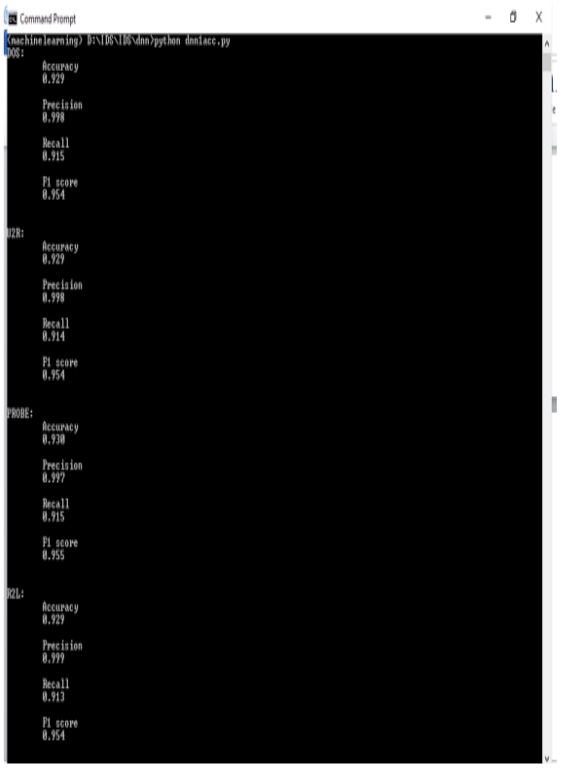
		ACTUAL VALUES	
PREDICTED VALUES		Actual: NO	Actual: YES
	Predicted: NO	True Negative	False Negative
	Predicted: YES	False Positive	True Positive

METRICS	FORMULA
Accuracy Rate	TP+TN/TP+FP+FN+TN
Recall(Detection Rate)	TP/TP+FN
Precision	TP/TP+FP
F-1 Score	2*(Recall * Precision) / (Recall + Precision)

VIII. Results

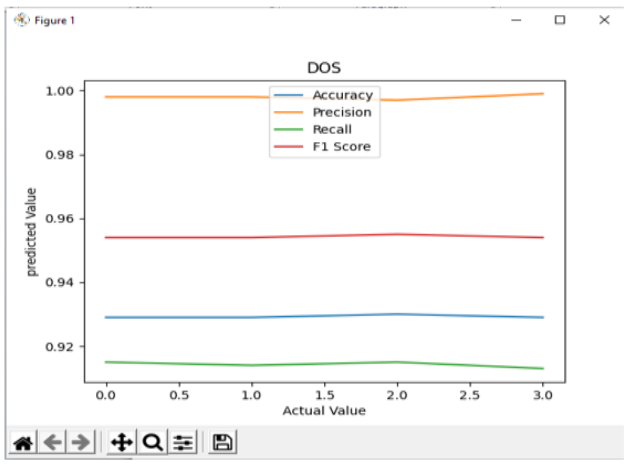
For the scope of this paper, the KDDCup-'99' dataset was fed into classical ML algorithmic program moreover DNNs of varying hidden layers. After the training is completed, all models were compared for f1-score, accuracy, recall and precision with the test dataset.

Cmd Output :

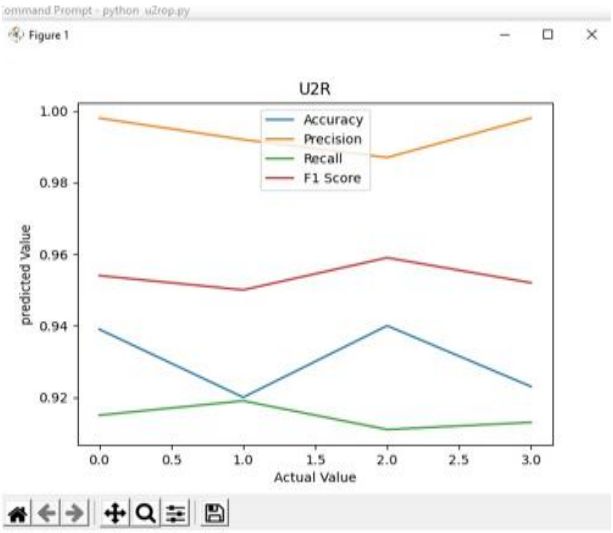


Histogram :

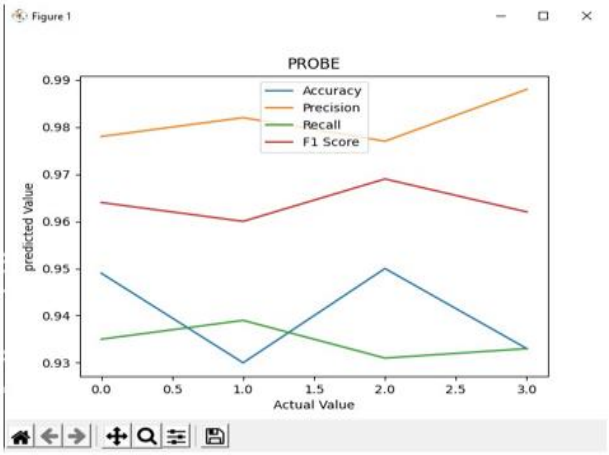
DOS Output:



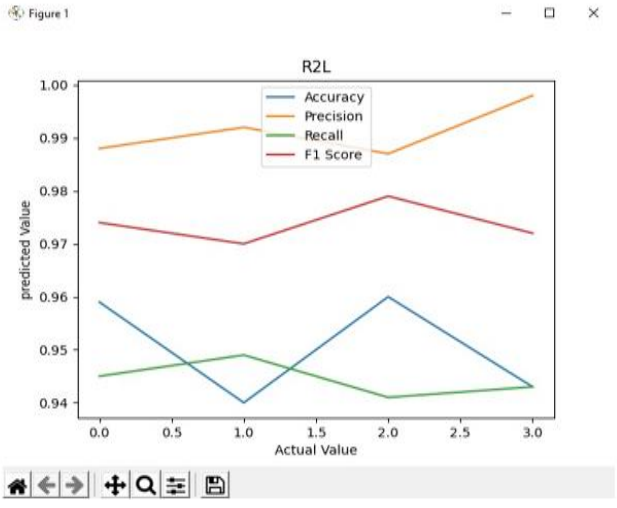
U2R Output:



PROBE Output:



R2L Output:



IX. Conclusion

This paper has elaborately recapitulated the usefulness of DNNs in IDS comprehensively. For the purpose of reference, other classical ML algorithms have been accounted and compared against the results of DNN. The publicly available KDDCup-'99' dataset has been primarily used as the bench- marking tool for the study, through which the superiority of the DNN over the other compared algorithms have been documented clearly. For further refinement of the algorithm, this paper takes into account of DNNs with different counts of hidden layers and it was concluded that a DNN with 3 layers has been proven to be effective and accurate of all.

From the empirical results of this paper, we may claim that deep acquiring methods are a promising direction towards cyber security tasks, but although the performance on artificial dataset is exceptional, application of the same on network traffic in the real-time which contains more complicated and current attack types is necessary. Additionally, studies regarding the flexibility of these DNNs in adversarial environments are required. The escalate in colossal variants of deep acquiring algorithmic program calls for an altogether evaluation of these algorithmic program in regard to its efficiency towards IDSs. This will be one of the directions towards IDS research can wander and for this reason will stay as a job of future. into account of DNNs with diverse counts of hidden layers and it was concluded that a DNN with 3 layers has been proven to be efficient and correct of all.

X. Reference

[1] S.Niksefat, P.Kaghazgaran and B.Sadeghiyan "Privacy issues in intrusion detection systems: A taxonomy, survey and future directions". Comput. Sci. Rev., 25, 69–78,2017.

[2] D. Archer, L. Chen, J. H. Cheon, R. Gilad-Bachrach, R. A. Hallman, Z. Huang, X. Jiang, R. Kumaresan, B. A. Malin, H. Sofia, Y. Song, and S. Wang, "Applications of homomorphic encryption," Homomorphic Encryption.org, Redmond WA, Tech. Rep., July 2017.

[3] B. Pfahringer. "Winning the KDD99 classification cup: Bagged boosting".
SIGKDD explorations newsletter, vol. 1, pp. 6566, 2000. DOI
http://dx.doi.org/10. 1145/846183.846200.

[4] Sarathiel Chaipa ; Mariki M Eloff ; Mariki M Eloff, "Towards the development of an Effective Intrusion Based Detection Model", 2017 Information Security for South Africa (ISSA), Johannesburg, South Africa, 16-17 Aug. 2017, Pretoria, IEEE.

[5] R. Kumar and D. Sharma, "Signature-Anomaly Based Intrusion Detection Algorithm," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, 2018, pp. 836-841, doi: 10.1109/ICECA.2018.8474781.

[6] I.Sharafaldin, A. H.Lashkari,, and A. A.Ghorbani "Toward generating a new intrusion detection dataset and intrusion traffic characterization" ICISSP, Funchal, Madeira-Portugal, 22–24 January 2018.

[7] R.Bost, R. A.Popa, S.Tu, and S.Goldwasser, "Machine learning classification over encrypted data", presented at the 2015 NDSS conference, CA, USA, Feb.8–11, 2015.

[8] H. Kayacik, A. Zincir-Heywood and M. Heywood. "Selecting features for intrusion detection: A feature relevance analysison KDD 99 intrusion detection datasets". In Proceedings of the third annual conference on privacy, security and trust (PST). 2005.

[9] C. Lee, S. Shin and J. Chung. "Network intrusion detection through genetic feature selection". In Seventh ACIS international conference on software engineering, artificial intelligence, networking, and parallel/distributed computing (SNPD), pp. 109114. IEEE Computer Society, 2006

[10] S. Chavan, K. Shah, N. Dave, S. Mukherjee, A. Abraham and S. Sanyal. "Adaptive neuro-fuzzy intrusion detection systems". In Proceedings of the international conference on information technology: Coding and computing (ITCC), vol. 1, pp. 7074. IEEE Computer Society, 2004. DOI http://dx.doi.org/ 10.1109/itcc.2004.1286428.