IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

# NETWORK INTRUSION DETECTION SYSTEM USING DEEP NEURALNETWORKS

Dr.G.Vennira Selvi- Professor, [1]Goli Koteswarao, [2]Yangammagari Sai Kiran, [3]Shaik Anas, [4] Ande Karthik, [5] AFI P Mohammod

COMPUTER SCIENCE AND ENGINEERING(INTERNET OF THINGS),
PRESIDENCY UNIVERSITY, BENGALURU, INDIA

***Abstract:*** . In recent decades, rapid development in the world of technology and networks has been achie ved, also there is a spread of Internet services in all fields over the world. Piracy numbers have increased, also a lot of modern systems were penetrated, so the developing information security technologies to detect the new attack become an important requirement. One of the most important information security technologies is an Intrusion Detection System (IDS) that uses machine learning and deep learning techniques to detect anomalies in the network. The main idea of this paper is to use an advanced intrusion detection system with highnetwork performance to detect the unknown attack package, by using a deep neural network algorithm, also in this model, the attack detection is done by two ways (binary classification and multiclass classification). The proposed system has shown encouraging results in terms of the high accuracy (99.98% with multiclass classification and with binary classification).

***Index Terms*** **- Component, formatting, style, styling, insert.**

## 1. INTRODUCTION

After the rapid growth of technology and the spread of internet networks across the world, cyber-crime increased dramatically. According to the Internet Security Threat Report (ISTR), around 430 million new types of malwares were detected, 362 of which were Crypto-ransom ware in 2015 [1]. The estimatedcybercrime rates produced 1.5 trillion US$ in 2018. If there's one clear conclusion in 2019, it is that nocompany is safe from a cyber-attack, large or small. Cyber-attacks are more advanced, elusive, and targeted than ever before. [2]. Therefore, security techniques must be continuously developed.

A network intrusion detection system (NIDS) plays an important role in network security, where it detects the intrusions and communicates the appropriate authority. According to detection techniques, there are two kinds of IDS that we can classify: Anomaly detection and Misuse detection [3]:

**Anomaly Detection:** makes a normal activity database and any deviations from the normal activityare alerted that intrusion is possible in the network.

**Misuse Detection:** defines the attacks activity in the database and If there are the same kind of possibilities in the network then they are defined as attacks.

After increasing the cyber-crime types, the anomaly detection system becomes better than the misusedetection system for building a network intrusion detection system. An anomaly detection system is more suitable for detected unknown attacks.

Much artificial intelligence (AI) algorithms have been implemented for the misuse detection system and Anomaly detection system. Rule-based techniques, machine learning, and data mining techniques are direct techniques for executing these IDSs. Data mining is the first framework suggested for the development of an IDS. Data mining is the method of collecting knowledge from an extensive database.It helps to extract patterns from a knowledge base and use them to predict future intrusion in related datasets. [4]

There are drawbacks to most rule-based IDSs, though. They do not have the ability to detect new attacks that use new signatures because they don't have these signatures in their knowledge base, proposed deep learning techniques to get rid of these drawbacks. DNN is one of the most popular deeplearning techniques used.
DNNs contain an attractive essential intrusion detection function called learning by training, to deduce new information to provide a decision, this makes DNNs distinguishes from all the traditional programming techniques and make it an expert system. [4]

## 2. RELATED WORKS

There are many research trends to a topic of Intrusion Detection System using deep learning and machine learning techniques,
some of these related works will be clarified as follow:
The work in [5], proposed the hybrid machine learning system (decision tree with support vector machine algorithms) to
raise the accuracy of the system. Decision Tree algorithm is used to classify theknown attack types. The support vector
machine (SVM) algorithm is used to classify the normal data. This model used NSL-KDD Dataset. The accuracy of this
system was 96.4 %. In [6] the researchers used a genetic algorithm (GA) with a support vector machine (SVM) to detect
the intrusion packets. SVM and GA are used with select features. To Classification and regression problems the
researchers use SVM. The Dataset of this paper was KDD Cup 1999, and the accuracy of detection was 97.3%. The
researchers in [7] established a network detection system by using Recurrent Neural Networks algorithmwith NSL-KDD
dataset the output of this paper is divided into binary classification with 83.28% accuracy, multiclass classification with
81.29% accuracy. The proposed system in [8] used a convolutional neural network to detect intrusion of the network.
Datasets were created by the KDD Cup1999 dataset, two dimensionalization were performed on test data for the CNN-
IDS model. The Detection Rate of this model was 97.7%. In [9] the researchers used Artificial Neural Network for
network intrusion detection system with KDD Cup 1999 dataset .in preprocessing this system used Principal Component
Analysis (PCA) to reduce the number of features, and min/max formula to normalized data. In artificial neural network
Architecture, this paper using Feed Forward Neural Network (FFNN) and Levenberg-Marquardt (LM) Backpropagation
and mean squared error as a loss function. The accuracy of this model was 97.97%.In [10] the proposed system was deep
neural networkwith NSL-KDD datasets .they proposed to use label-encoder and min-max normalization in preprocessing
and auto-encoders network for training the deep learning layers .This model building according to five categories .the
best accuracy detection from this categories was dos attack, reach to
97.7 %, and 89.8% for probe etc. In [11]the researchers proposed an artificial intelligence (AI) intrusion detection system using a deep neural network (DNN) was analyzed and checked with the KDD Cup 99 dataset. They used four hidden layer to build neural network, ReLU function as the activation functionin the forwarding and Adam optimizer in back-propagation training. The classification type was binaryclassification (normal or attack) and the accuracy was 99.08%. The proposed system in [12] was deepneural network (DNN) with DARPA 1999 dataset. With used ReLU as the non-linear activation functionin hidden layer, the output layer consists only of two neurons (Attack and Benign). Accuracy was 93%. In this paper, the proposed system to detect network intrusion packets by used Deep Neural Network (DNN) for its relevance and efficiency, this algorithm gives suitable weight for all features in the inputlayer and used these weights in decision making. So it is suitable for new signatures more than Rule-based. In this paper we use one-hot encoder technique and

z-score normalized in preprocessing data .indeep neural network training ,we used ReLU as the non-linear activation function , adam function as optimizer ,and cross entropy as loss function .the output classification methods used both binary andmulticlass

classification

**Table 1.** related works papers

| Researchers | Paper year | dataset | algorithms | accuracy |
|---|---|---|---|---|
| R. Wankhede and. C. Vikrant [5] | 2016 | NSL-KDD | Decision Tree (DT) and support vector machine (SVM) (hybrid classification ) | 96.4% |
| B. M. Aslahi-Shahri et.al [6] | 2015 | KDD CUP 1999 | genetic algorithm (GA) and support vector machine (SVM) (hybrid classification ) | 97.3%. |
| Chuan long yin et.al [7] | 2017 | NSL-KDD | Recurrent Neural Networks (RNN) in binary and multiclass classification | 83.28% 81.29% |
| Yuchen Liu et.al [8] | 2017 | KDD Cup 1999 | Convolutional neural network (CNN) | 97.7%. |
| Brijpal Singh et.al [9] | 2016 | | | |
| Sasanka Potluri et.al [10] | 2016 | KDD Cup 1999 | Artificial Neural Network (ANN) with PCA, NARX Neural Network | 97.97%. |
| | | NSL-KDD | Deep Neural Network(DNN) with auto-encoders network | 97.7 % |
| Jin Kim et.al [11] | 2017 | KDD Cup 1999 | Deep Neural Network (DNN) with ReLU activation function and Adam optimizer. In binary classification. | 99.08%. |
| Rahul Vigneswaran K et.al [12] | 2018 | DARPA 1999 | Deep Neural Network (DNN) with ReLU activation function. In binary classification Deep Neural Network (DNN). | 93% |
| the proposed system | 2020 | KDD Cup 1999 | Feed forward network with ReLU activation function and Adam optimizer.in binary and multiclass classification | 99.98% |

## 3. METHODOLOGY

The proposed intrusion detection system discover the attacks by using a deep neural network algorithmwith anomaly detection techniques without accessing information in the packet payload to avoid a breach of data privacy, this system establishes in several steps will be described below

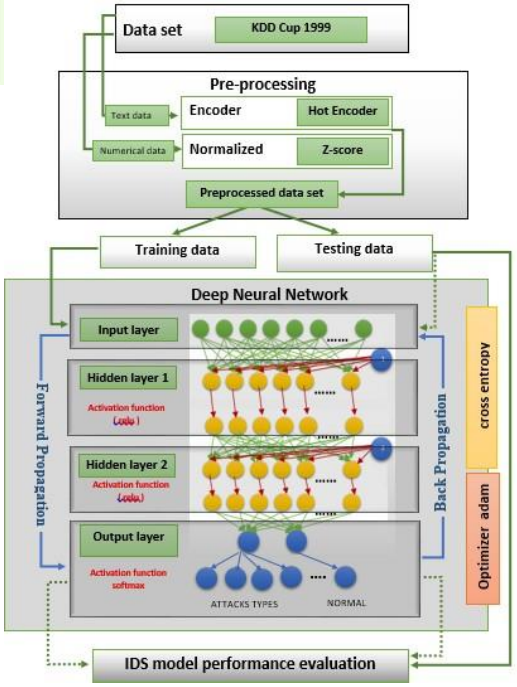



**Figure 1.** Block diagram of proposed DNN-NIDS

*3.1. Data Representation*

KDD CUP 1999 is an extremely large dataset that used with intrusion detection experiments, the complete dataset has approximately 4.5 million records. This dataset has 41 features that can be classified into three main Categories: TCP connections features, Content features, and Traffic features[13]. More details are shown in Table2:

**Table 2.** Features of KDDCUP 99 dataset

| ID | feature name | description | type |
|---|---|---|---|
| | | **TCP connections features** | |
| 1 | duration | length (number of seconds) of the connection | continuous |
| 2 | protocol_type | type of the protocol, e.g. tcp, udp, etc. | discrete |
| 3 | service | network service on the destination, e.g., http, telnet, etc. | discrete |
| 4 | src_bytes | number of data bytes from source to destination | continuous |
| 5 | dst_bytes | number of data bytes from destination to source | continuous |
| 6 | flag | normal or error status of the connection | discrete |
| 7 | land | 1 if connection is from/to the same host/port; 0 otherwise | discrete |
| 8 | wrong_fragment | number of ``wrong'' fragments | continuous |
| 9 | urgent | number of urgent packets | continuous |
| | | **Content features** | |
| 10 | hot | number of ``hot'' indicators | continuous |
| 11 | num_failed_logins | number of failed login attempts | continuous |
| 12 | logged in | 1 if successfully logged in; 0 otherwise | discrete |
| 13 | num_compromised | number of ``compromised'' conditions | continuous |
| 14 | root shell | 1 if root shell is obtained; 0 otherwise | discrete |
| 15 | su_attempted | 1 if ``su root'' command attempted; 0 otherwise | discrete |
| 16 | num_root | number of ``root'' accesses | continuous |
| 17 | num_file_creations | number of file creation operations | continuous |
| 18 | num_shells | number of shell prompts | continuous |
| 19 | num_access_files | number of operations on access control files | continuous |
| 20 | num_outbound_cmds | number of outbound commands in an ftp session | continuous |
| 21 | is_hot_login | 1 if the login belongs to the ``hot'' list; 0 otherwise | discrete |
| 22 | is_guest_login | 1 if the login is a ``guest'' login; 0 otherwise | discrete |
| | | **Traffic features** | |
| 23 | count | number of connections to the same host as the current connection in the past two seconds | continuous |
| 24 | dst_host_count | count of the connections having same dst host | continuous |
| 25 | serror_rate | % of connections that have ``SYN'' errors | continuous |
| 26 | rerror_rate | % of connections that have ``REJ'' errors | continuous |
| 27 | same_srv_rate | % of connections to the same service | continuous |
| 28 | diff_srv_rate | % of connections to different services | continuous |
| 29 | srv_count | number of connections to the same service as the current connection in the past two seconds | continuous |
| 30 | srv_serror_rate | % of connections that have ``SYN'' errors | continuous |
| 31 | srv_rerror_rate | % of connections that have ``REJ'' errors | continuous |
| 32 | srv_diff_host_rate | % of connections to different hosts | continuous |
| 33 | dst_host_srv_count | count of connections have same dst host and using same service | continuous |
| 34 | dst_host_same_srv_rate | % of connections have same dst port and using same service | continuous |
| 35 | dst_host_diff_srv_rate | % of different services and current host | continuous |
| 36 | dst_host_same_src_port_rate | % of connection to current host having same src port | continuous |
| 37 | dst_host_srv_diff_host_rate | % of connections to same service coming from diff. hosts | continuous |
| 38 | dst_host_serror_rate | %of connection to current host that have an S0 error | continuous |
| 39 | dst_host_srv_serror_rate | %of connection to current host and specified service that have an S0 error | continuous |
| 40 | dst_host_rerror_rate | % of connection to current host that have an RST error | continuous |
| 41 | dst_host_srv_ rerror_rate | %of connection to the current host and specified service that have an RST error | continuous |

There are 22 kinds of attacks in this dataset, which can be grouped into 4 major categories:

1. Denial of Service Attack (DoS): occurs when the attacker prevents legitimate users access to a system, by making system resource (computing or memory) too busy to handle legitimate requests e.g. syn flood

2. User-to-Root attack: the attacker has local access to a normal user on the system (may be obtained by a dictionary attack, sniffing passwords, or social engineering) and through some vulnerability attempts to obtain root access to the system to gain the capabilities of thesupervisor e.g., various

``buffer overflow" attacks.

3. Remote to Local Attack (R2L): occurs when the attacker sends packets from a remote machine over a network without having authorized access e.g. guessing a password;

4. Probing occurs when the attacker tries to gain information about the network to find some vulnerability. Via these operations, the attacker maps the topology of the network and discoversthe type of services operating on the network. e.g., port scanning. Figure 2 show attacks numbersin the full KDD CUP 1999 dataset

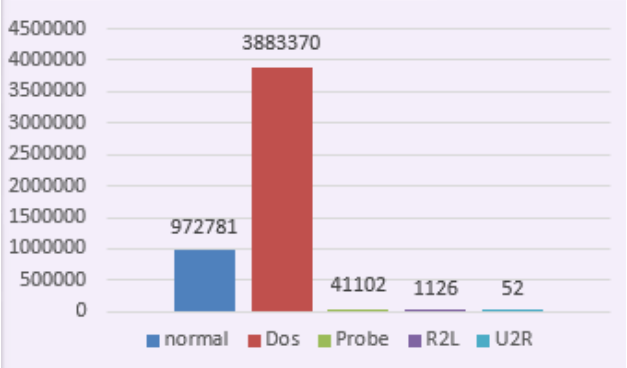



**Figure 2.** classification the class of full KDDCUP 99 dataset

*3.2. Preprocessing*

KDD CUP 99 dataset has not noise or missing value so it is a clear dataset, but it has numerical and Textvalues, the numerical value having large numbers, it will delay the training and complicate theprocessing. In addition, the text values cannot be processed in the operations of the deep neural networkalgorithm. Hence, the dataset must be preprocessed. The preprocessing in this model can be classified into two main steps: the normalization process and text mapping.

Z-score normalization stated in Equation 1 has been used for normalizing the numerical attributes to lessen its values and reduce the training processes (time and space).

$$Z = \frac{3-\mu}{\sigma} \quad (1)$$

Where Z refers to the Z-score normalization, x refers to the values, μ refer to the mean of the sampleand σ referred to the standard deviation of the sample.

In another way, Converting the text attributes to numerical values have been executed using One Hotencoder easy handling with math equations as Figure 3 represent the execution of protocol type attribute(column in KDDCUP 99 dataset ) on one hot encoder

| Protocol type |
|---|
| tcp |
| tcp |
| udp |
| icmp |
| tcp |
| icmp |

| Protocol type tcp | Protocol type udp | Protocol type icmp |
|---|---|---|
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 0 | 0 | 1 |

**Figure 3.** Explain one hot encoder on protocol type column

The number of features has been increased to 125 features rather than 41 features after executing onehot encoder on the dataset. In this paper, 75% of the data were assigned for the Training process, and 25% for the testing process.

### *3.3. Deep Neural Network Model*

In general, deep neural networks are considered one of the most important computational networks, which consists of multiple hidden layers with nodes and ways of interconnecting nodes. The deep neuralnetworks algorithm that builds the model of this paper can be described via three main steps. First: the topology of the model, which describes the number of layers and neurons for each layer with the connections between them. Second: the forward propagation with its perceptron classifier and activation function used by the artificial neurons. Third: the back propagation with loss function and optimizer.

#### *3.3.1. The Model Topology*

1. **Input layer:** it initializes data for the neural network purposes. The used system is based on 125 nodes as the input layer, which is represented by the features of the preprocessed dataset.
2. **Hidden layers**: It is the intermediate layer between the input and output layer and place where all the computation is done. The used system based on two hidden layers with (50 neural nodes)for the first hidden layer and (30 neural nodes) for the second hidden layer. This number selectedaccording to the training.
3. **Output layer**: it produces the result (normal or attack with mention to attack types).

All nodes in the input layer have full connect with all nodes in the next hidden layer, and so on in allthe rest layers. The connection of the nodes is considered as a connected graph as stated in Figure 4.

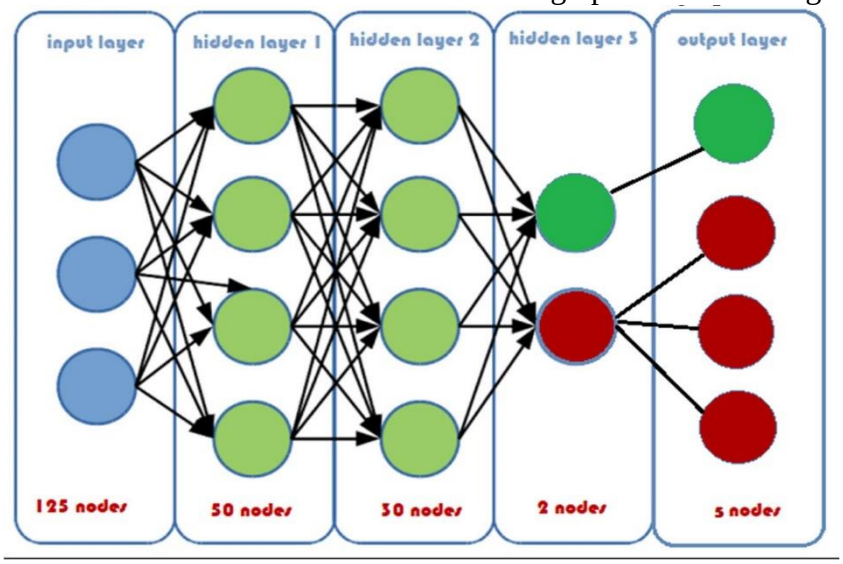



**Figure 4.** Proposed deep neural network topology

3.3.2. The Forward Propagation

Forwarding propagation aims to predict results (attack or normal) by using a perceptron classifier. Perceptron

is supervised learning, it has two types, Single-Layer Perceptron, and multi-layer perceptron. Artificial neural networks used a multi-layer perceptron, and the artificial neural network is the basis of deep neural networks. The main Equation of the perceptron is mentioned in Equation 2

$$y = \sum_{i=1}^{n} Xi\ Wi + b \quad (2)$$

Where n refers to the numbers of nodes in the layer, x refers to the values of these nodes (values ofthe dataset), W refers to Weights (the strength of the connection) and b refers to the Bias of these nodes.The results will be inserted into Activation Functions, in general, the theory of an activation function derives from the study of how a neuron acts inside the human brain, and the neuron becomes activated

above a certain level defined as the activation potential. This also puts the results within a limited range.Among the most common activation functions are Sigmoid, ReLU, softmax, and tanh, this model usedReLU activation functions (stated in Equation 3) in the hidden layers.

$$(x) = 0 \ \ for\ x \le 0 \quad (3)$$
$$= x\ for\ x > 0$$

Softmax activation functions stated in Equation 4 in the output layer.

$$\sigma(z)$$

$$\text{for } j = 1,2,3 \ldots \ldots, k \quad (4)$$

$$\frac{e^{zj}}{\sum_{k=1}^{k} e^{zk}}$$

Softmax activation functions transform an output prediction to a categorical probability variable.

*3.3.2. The Backpropagation*

Backpropagation is a popular way to train a deep neural network by Modification weights and bias [14].It contains loss function and Optimizers. The loss function (cost function) will reduce the value to reachthe best values for the Model parameter. There are various parameters for each model, the model configuration is represented in terms of the parameter values which are referred to as weight and bias inthe neural network. The model can be evaluated by cost function (loss function). Minimizing the loss function will be the guiding factor for each parameter to achieve the optimal value.

cross entropy Loss Functions have been applied in this work. The loss function needs to reach the optimal value of the model parameter (weight and bias). The optimizer is the way to get the best parameter value [14]. Among the most common Loss Functions are Batch gradient descent, RMSprop,Stochastic gradient descent, and Adam. the best optimizer was Adam according to testing several optimizers,

*3.4. EVALUATION METRICS*

To evaluate the DNN-IDS model, we implemented the NIDS according to two ways of classification:
1) binary classification (Normal and attack), 2) multi-class classification (Normal, DoS, R2L, U2R, andProbe). This paper use Accuracy, Precision, Recall, F-score, Specificity, and AUC to Measures the binary classification. And Average Accuracy, Precision, Recall, F-score to Measures multi-class classification.[15] All that according to the confusion matrix that shows in table3.

**Table 3.** confusion matrix

| | | Predicted | | |
|---|---|---|---|---|
| | | Attacks | Normal | total |
| actual | Attacks | TP | FN | TP+FN |
| Normal | | FP TP+FP | TN FN+TN | FP+TN |
| total | | | | |

The next section will display the results of the DNN-IDS model with these evaluation metrics.

**4.** EXPERIMENTS AND RESULTS

In this model, the most common intrusion detection evaluation metrics (confusion matrix) is used to evaluate the ANN-IDS model performance. This experiment is executed on MSI GF75 Thin 9SD laptopwhich has Intel Core i7-9750H CPU @ 2.60 GHz, 16 GB memory without using GPU. The dataset wassplitting in 75% training

(3 673 823 samplings) and 25 % testing (1 224 611 samplings), the model takes1008 sec (divided into 10 epochs) to build it according to training data.
This section displays the results of the DNN-IDS model of Binary classification and multi-class classification.

### *4.1. Binary classification (Normal and attack)*

The results of the first classification (Normal and attack) is shown in table4 as a confusion matrix fortesting data

**Table 4.** confusion matrix of Binary classification for testing data

| | | Predicted | | |
|---|---|---|---|---|
| | | Attacks | Normal | total |
| actual | Attacks | 1 177 312 | 207 | 1 177 519 |
| | Normal | 108 | 291 903 | 292 011 |
| total | | 1 177 420 | 292 110 | |

This table means that there are 291903 normal packets detected true as normal from 292011 normalpackets, and the error obtains on 108 packets, they detected as an attack, same case in attacks, there are1177519 packets in testing data was an attack, detected 1177312 as an attack, and the error obtained on207 packets. Figure 6 display the rustle of Accuracy, Precision, Recall, F-score, Specificity, and AUC for Binary classification.

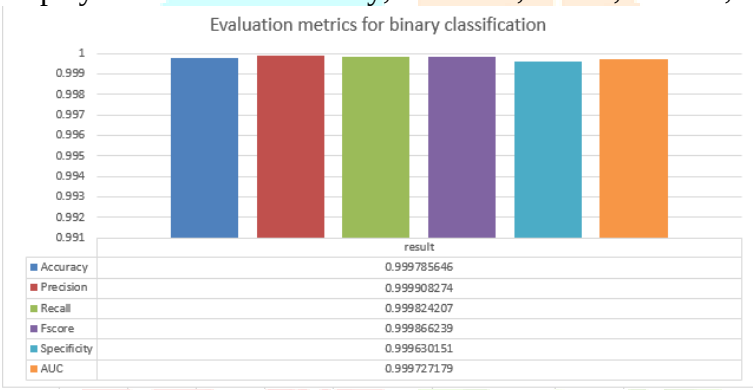



**Figure 5.** evaluation results for Binary classification (testing data)

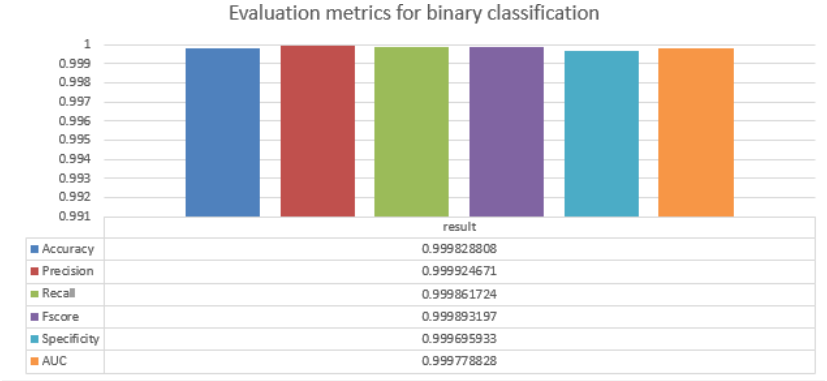
The confusion matrix of Binary classification (Normal and attack) for training data samples is shownin table 5

**Table 5.** confusion matrix of Binary classification for training data

| | | Predicted | | |
|---|---|---|---|---|
| | | Attacks | Normal | total |
| actual | Attacks | 2 747 751 | 380 | 2 748 131 |
| | Normal | 207 | 680 563 | 680 770 |
| total | | 2 747 958 | 680 943 | |

Figure 7 display the rustles of Accuracy, Precision, Recall, F-score, Specificity, and AUC for trainingdata

**Figure 6.** evaluation results for Binary classification (training data)



*4.2. Multi-class classification*

**In the second design** the attack is divided into 4 categories (DoS, R2L, U2R, and Probe), to be five- category with a normal class. In the testing part **the accuracy was 0.99979**, and the results of the confusion matrix for testing data shown in table 6

**Table 6.** confusion matrix of Multi-class classification for testing data

| | | Predicted | | | | | |
|---|---|---|---|---|---|---|---|
| | | DOS | Probe | R2L | U2R | normal | total |
| actual | DOS | 1 165 359 | 1 | 0 | 0 | 13 | 1 165 373 |
| | Probe | 5 | 12293 | 1 | 0 | 100 | 12 399 |
| | R2L | 1 | 0 | 269 | 0 | 79 | 349 |
| | U2R | 0 | 0 | 0 | 0 | 9 | 9 |
| | normal | 42 | 7 | 50 | 0 | 291 391 | 291 490 |
| | total | 1 165 407 | 12 301 | 320 | 0 | 291 592 | |

Table 6 means that there are 291391 normal packets detected true as normal from 291490 normal packets, and the error obtains on 42 packets detected as DOS attack,50 packets detected as Remote to Local Attack, and 7 as a probe. In Dos attacks, there are 1165359 packets detected true as a denial of service attack from 1165373 dos attack packets, the error obtained on 13 packets detected normal and 1packet detected as other types of attack, so according to this result, the accuracy of detected dos attacksis very high. And so in the rest of the results of the attacks.

Figure 8 display the rustles Precision, Recall, F-score per class dependent on testing data

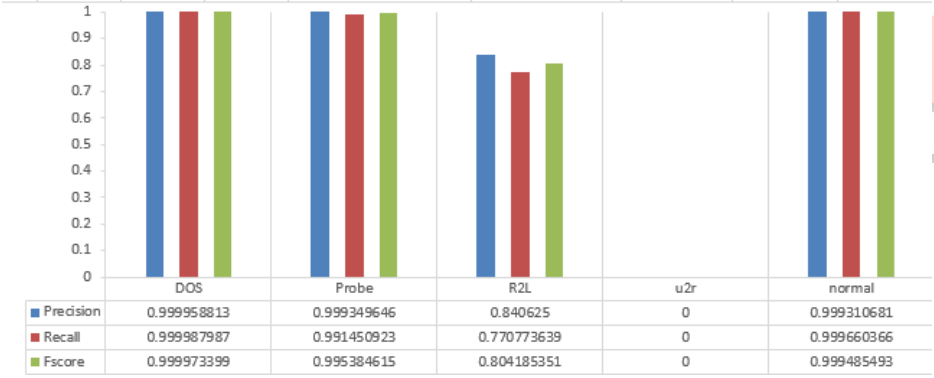



**Figure 7.** evaluation results for multiclass classification (testing data)

In the training part the accuracy was 0.999789, Table 7 describes the confusion matrix for multi- class classification that dependent on training data samples.

**Table 7.** confusion matrix of multi-class classification for training data

| | | Predicted | | | | | |
|---|---|---|---|---|---|---|---|
| | | DOS | Probe | R2L | U2R | normal | total |
| actual | DOS | 2 717 973 | 2 | 0 | 0 | 22 | 2 717 997 |
| | Probe | 6 | 28 501 | 3 | 0 | 193 | 28 703 |
| | R2L | 0 | 8 | 574 | 0 | 195 | 777 |
| | U2R | 0 | 0 | 1 | 0 | 42 | 43 |
| | normal | 100 | 20 | 132 | 0 | 681 129 | 681 381 |
| | total | 2 718 079 | 28 531 | 710 | 0 | 681 581 | |

Figure 9 display the rustles Precision, Recall, F-score per class dependent on training data

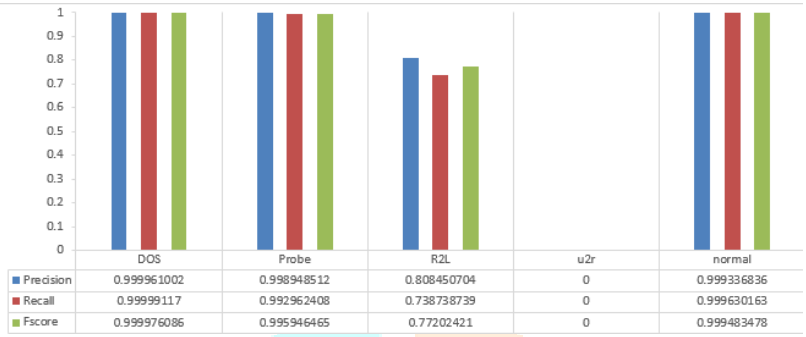



**Figure 8.** evaluation results for multiclass classification (training data)Figure 10 compares the accuracy of multi-class and binary classification


**Figure 9.** the accuracy of multi-class and binary classification

## 5. CONCLUSION

In this paper, we proposed two models (multi-class and binary classification) ., and proposed in these models to using deep learning techniques for detecting network attacks instead of using machine learning rules or signatures. Through this experimental research of multi-class classification, which hadbeen found in the KDD cup 99 datasets, we have shown that supervised learning models, which are DNN, are capable of detecting and classifying with high accuracy (99.98 %), and this detection executedon network packet analysis and connection parameters without packet payload information. Also, the accuracy of detected dos attacks was very high reach to 99.99%

## 6. REFERENCES

[1] Network Intrusion Detection Using Deep Neural Networks M.Ponkarthika1 and Dr.V.R.Saraswathy2 (Open Access Quarterly International Journal) Volume 2, Issue 2, Pages 665-673, April-June 2018
[2] Host Based Intrusion Detection System with Combined CNN/RNN ModelAshima Chawla(B), Brian Lee, Sheila Fallon, and Paul Jacob
[3] On the Effectiveness of Machine and Deep Learning for Cyber Security 2018 10th International Conference on Cyber Conflict

[4] Collective Anomaly Detection Based on Long Short-Term Memory Recurrent Neural Networks Lo¨ıc Bontemps, Van Loi Cao(B), James McDermott, and Nhien-An Le-Khac
[5] A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection Anna L. Buczak, Member, IEEE, and Erhan Guven, Member, IEEE
[6] Application of Neural Networks for Intrusion Detection in Tor Networks Taro Ishitaki∗, Donald Elmazi†, Yi Liu ∗, Tetsuya Oda ∗, Leonard Barolli‡ and Kazunori Uchida‡2015 29th International Conference on Advanced Information Networking and Applications Workshops
[7]ApplicationofDeepRecurrentNeuralNetworksforPredictionofUserBehaviorin To Networks 2017 31st International Conference on Advanced Information Networking and Applications Workshops