Comparative Analysis of Neural Networks for Intrusion Detection System

Nidhi Kakde[1], Nirali Shah[2], Bhumi Tejani[3], Prof. Martina D'Souza[4]

[1-3]Dept. of Information Technology, Xavier Institute of Engineering, Mumbai, India
[4]Prof. Martina D'Souza Dept. of Information Technology, Xavier Institute of Engineering, Mumbai, India

---***---

Abstract - *With rapid expansion in the range of computer hardware, networking and operating systems along with continuously changing capabilities and creativity of the attackers and ever-changing nature of threat to the systems, security of data has become a very crucial issue. Due to the exponential growth of network traffic data and modern attacks requirements, traditional Network Intrusion Detection Systems (NIDS) encounter difficulties. While Machine Learning algorithms show promising results, the Neural Networks have now gained popularity and are widely used for many applications. This paper presents a comparative analysis of four Neural Network based algorithms namely- CNN, DNN, LSTM and MLP to highlight the efficiency of the Neural Networks in detecting the attacks. For training and evaluating the intrusion, NSL-KDD dataset is used. The result of this comparative analysis has been found to be efficient with high accuracy and has a promising scope for further research.*

Key Words: Information security, Intrusion detection system, Machine Learning Algorithms, Neural network algorithms, NSL-KDD dataset.

1.INTRODUCTION

We know that networking has become a part of our lives. Many institutions have heavy usage of technology accompanied by networks. As the technology is being expanded each day it brings along various threats into picture. The efficiency of the network was being affected and such threats included malicious programs. We very well know that Data is today's gold and transmission of confidential data through networks is a massive threat to the institution. This issue needed a solution and thus, cyber security is the main domain where we can find solutions to such issues. Cyber security plays an important role in protecting the system information by preventing, as well as detecting and responding to attacks.

There are different types of threats with various intensities and these are capable of attacking computer systems and networks. There can be big losses like deletion of one's entire system, penetration of unauthorized users in a system, alteration of files, even credit card and bank information can be stolen, etc. This is where the Need for an Intrusion Detection Systems (IDS) becomes important. Usually, penetration and threats are made by hackers to access one's computer system or a network they desire either to collect information or perform an attack. IDS is an integral component of an in-depth architecture that provides a complete computer network security defense. A security analyst is given a chance or an opportunity to react against the threats by an alarm raised by IDS whenever an intrusive event is detected. IDS is not able to prevent threats but through detection of threats, a good amount of information is collected which can be used in filling the loopholes of the system. The outputs of this system are mostly alerting.

Hence, in today's world where there is vast network usage as well as usage of computer systems, we aim to build an IDS with a good accuracy record to prevent leakage of data and keep network and computer systems safe [5].

2. INTRUSION DETECTION SYSTEM

Intrusion detection systems are used to detect anomalies like unauthorized access, misuse and attacks 3on information systems in preferably real-time with an aim to catch the attackers before they do some real damage to the network or host system. They monitor the network traffic for any suspicious activities and alerts when such activities are discovered. Basic

duties of Intrusions Detection Systems include maintaining a set of user's profile history, matching an audit record with an appropriate profile, updating profile whenever necessary and reporting anomalies when detected [1]. While these are primary functions of an IDS, some IDSs are also capable of taking measures when a malicious activity is detected like blocking traffic from suspicious IP addresses.

Intrusion Detection Systems work by looking for signatures of known attacks or by detecting deviation of a system behavior or an activity from normal. These deviations are pushed up in the stack and then examined at protocol and application layer [10]. While IDS detects attack, it usually does not take any action to prevent the intrusions, they just alert the system administrators of the possible security breach. So, we can call IDS a proactive tool instead of a reactive tool.

3. TAXONOMY

IDSs comes in different types and detect suspicious activities using different methods, including the following:

1.Host Based IDS: They are installed on client computers or devices in the network with direct access to the internet as well as the organization's internal network. They monitor and track the changes made to the files and directories in the system by comparing the snapshot of the existing system state with the previous state of the system and alerts the admin if any deviation or breach is detected. They are also able to identify malicious traffic from the host system itself. Host Based IDS are also defined as personal firewalls or agent-based software.

2. Network Based IDS: They are positioned at strategic point or points in the network from where it can monitor and analyze the inbound and outbound traffic to and from all the devices connected to the network. System monitors the traffic passing through its own segment and the Network Interface card captures all network traffic that passes through the segment. A sensor determines if packet flow matches with any of the known signatures. Cloud-based intrusion detection systems are also available to protect data and systems in cloud deployments.

3. Signature Based IDS: They contain a database having attack signatures of all the known vulnerabilities. It monitors the network traffic and compares them against this database and raises alarm if the known attack is matched. Its working is very similar to antivirus software and can be placed on both network and host systems. In this case, the rate of false alarm rate is less and only known attacks can be detected.

4. Anomaly Based IDS: They monitor the network and system behavior with respect to an established forecast or a baseline based on say, bandwidth, ports, protocols, etc. to determine what is considered as normal network or host behavior. Machine learning is used to design these IDSs. If the pattern obtained is suspicious then, it is flagged down as an attack and the admins are alerted. In this case, the rate of false alarm is high and it can detect unknown attacks.

4. NEURAL NETWORK

Neural Networks represent deep Learning using Artificial Intelligence. According to Howard Rheingold, Neural Networks are a kind of technology defined by a network that has weights on it and these weights can be adjusted so that it can learn through trials [12]. Neural Networks is a system of hardware or software inspired from the working of the human brain and nervous system. The connections of the biological neurons are modeled as weights [14]. If we are to put it practically, Neural Networks are non-linear statistical data modeling tools which can be used to model complex relationships between inputs and outputs or to find patterns in data

Working of Neural Networks:
A neural network consists of a large number of processors that operate parallelly but are arranged as tiers. Raw input is received by the first tier (like raw info received by the optic nerve in humans). Each of the successive nerves then receives input from the tier before it and passes on its output to the tier after it. Final output is processed by the last tier. Tires are made up of small nodes. These nodes in consecutive tiers are highly interconnected. Each node has its own boundary of knowledge which includes rules that are already programmed and also the rules that it learns by itself. The nodes weigh the importance of the input it receives from the previous node and gives the highest weight to the inputs that it considered contributing the most towards the right output [12].

5. METHOD

Each of the existing system have their own disadvantages concerning false alarm rate and detection of unknown attacks.

To have a detailed analysis of Which algorithm works best on IDS or is the most efficient choice for intrusion detection system, we implemented four algorithms namely Convolution Neural Network (CNN), Deep Neural Network (DNN), Multi-Layer Perceptron (MLP) and Long short-term memory (LSTM), a kind of Regression Neural Network (RNN).

We used the following algorithms due to various reasons mentioned below:

Convolution Neural networks (CNN) is made up of Multiple layers and has the fast -processing ability. CNN can learn from various levels of features from a vast amount of data that is unlabeled. Therefore, the ways CNN can be used in a field of network intrusion detection are comprehensive. It learns through pattern recognition and thus the accuracy of detecting an attack is very good.
Deep Neural Network (DNN) uses Feed Forward Networks that only work in a one-way format. The data flows from input layer to output layer and does not go backward which results in having no memory of the past experiences.
Multi-Layer Perceptron (MLP) train on a set of input-output pairs and learn to model the correlation (or dependencies) between those inputs and outputs. MLP approximates any continuous function and can solve problems which are not linearly separable.
Long short-term memory (*LSTM*), is basically a modification of Regression Neural Network **(RNN).** LSTM cells has memory that can store previous timestep information and this is how it learns. It trains the model by using back-propagation. They not only work on the information you feed but also on the related information from the past and capable of learning long-term dependencies.[15] Thus, by using these algorithms on the IDS we can have a comparative analysis of the accuracies each of the model provides.

For training and evaluating the intrusion, NSL-KDD dataset is used and the accuracies are recorded accordingly.

The IDS is capable of detecting both known and unknown attacks and will categorize them into normal or threat with low rate of false alarms.

6. DATASET

NSL-KDD training dataset consists of 1,48,517 single connection vectors each of which contains 43 features and are label as either normal or an attack, with exactly one specific attack type.

The dataset is the collection of network related information that was captured over a period of time. The data consists of a number of basic features: duration of the connection, protocol type, such as TCP, UDP or ICMP, service type, such as FTP, HTTP, Telnet, status flag, total bytes sent to destination host, total bytes sent to source host, whether source and destination addresses are the same or not, number of wrong fragments, number of urgent packets [16].

The simulated attacks fall in one of the following four categories:

1) Denial of Service Attack (DoS)
2) User to Root Attack (U2R
3) Remote to Local Attack (R2L
4) Probing Attack

The test data is not from the same probability distribution as the training data, and it includes specific attack types not in the training data which make the task more realistic. The training dataset is made up of 23 different attacks out of the 39 are present in the test dataset.

Features are grouped into four categories:

Basic Features: Basic features can be derived from packet headers without inspecting the payload. It includes the features such as protocol-type, service-type, duration, flag etc.

Content Features: Domain knowledge is used to access the payload of the original TCP packets. This includes features such as the number of failed login attempts.

Time-based Traffic Features: These features are designed to capture properties that mature over a 2 second temporal window. One example of such a feature would be the number of connections to the same host over the 2 second interval.

Host-based Traffic Features: Utilize a historical window estimated over the number of connections instead of time[1].

7. IMPLEMENTATION

Data Pre-processing:

We applied the NSL-KDD dataset which contains network connection features to evaluate the neural Network models. Thus, the data pre-processing is required to convert the raw data into a proper format to feed it to the NN model. For this, the first step is to map the 'normal ' and 'attack' values in the dataset with binary values that is '0' for normal and '1' for attack. This eliminates the categorical property of the label attribute. After this the rest of the categorical data in the dataset should be mapped to the numeric data at first and then the overall data should be normalized. As the dataset now consists of the three categorical features that is protocol type, flag and service, they are converted into numeric attributes using the one-hot encoder technique.

Then we applied the min-max normalization technique to the dataset to scale the original data to a fixed range of 0 to 1. This ensures the consistency of the data distribution avoiding the exploding gradients problem in the training phase.
Equation (1) represents the min-max normalization formula, where $Xscaled$, and X is the normalized, and original value, respectively. $\min(X)$, and $\max(X)$are the minimum and maximum values of the data.

$$X_{scaled} = \frac{X - \min(x)}{\max(x) - \min\,(x)}$$

Equation (1)

The above processing is done for both the train and the test sets followed by the splitting of the label attribute to finally get the sets required for developing the models. The final training and testing sets contain 83 attributes.

In this paper we mainly intend to highlight the efficiency of using the neural network algorithms in classification of the attacks. This implementation concerns the binary classification – attack or normal. Thus, for this purpose we have evaluated four neural network-based algorithms namely – Convolution Neural Network (CNN), Deep Neural Network (DNN), Multi-Layer Perceptron (MLP) and Long short-term memory (*LSTM*), a kind of Regression Neural Network (RNN).

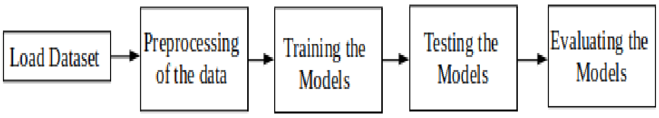

Fig-1: Flow of the process

The models are compiled with parameters - loss function: binary cross-entropy and optimizer: Adam. The overview of the structure of each model is explained below:

Implementation of the CNN model:

The model consists of 2 1D convolution and max pooling layers with activation function "ReLU" and kernel size = 3 along with a flatten layer and two Dense layers with activation function "sigmoid".

Implementation of LSTM model:

The model consists of 4 LSTM layers along with 1 Dense layer whose activation function is "sigmoid".

Implementation of MLP model:

Here, we use MLP Classifier with hidden_layer_sizes = (30,30,30). The activation function used is "ReLU".

Implementation of DNN model:

The model consists of 6 Dense layers, of which the first layer has activation "ReLU" and the last has activation function "sigmoid".

Model Evaluation Metrics:

For the evaluation of the performance of the model, we considered accuracy, precision, false alarm and F1-score metrics. These metrics use properties from the confusion matrix such as true positive (TP), false positive (FP), false negative (FN), and true negative (TN). TP is the number of attacks that are correctly classified as attacks, while FN is the attacks that are incorrectly classified. The number of incorrectly classified normal data is FN, and TN is correctly classified as normal data.

Equation (2), (3), (4), (5), and (6) are the mathematical definition of the performance metrics accuracy, precision, recall, false alarm, and F1-score, respectively.

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{(\text{True Positive} + \text{False Positive} + \text{True Negative} + \text{False Negative})}$$

Equation (2)

$$\text{Precision} = \frac{\text{True Positive}}{(\text{True Positive} + \text{False Positive})}$$

Equation (3)

$$\text{Recall} = \frac{\text{True Positive}}{(\text{True Positive} + \text{False Negative})}$$

Equation (4)

$$\text{False Alarm} = \frac{FP}{FP + TN}$$

Equation (5)

$$\text{F1-score} = \left(\frac{\text{Recall}^{-1} + \text{Precision}^{-1}}{2}\right)^{-1} = 2 * \frac{(\text{Precision} * \text{Recall})}{(\text{Precision} + \text{Recall})}$$

Equation (6)

8. RESULT

The Table-1 shows the comparison of the metrics of the evaluated models. The metrics are calculated on the basis of the test set.

Models	Accuracy (%)	Precision (%)	Recall (%)	False Alarm (%)	F1-Score (%)
MLP	85.88	92.78	81.52	7.21	86.78
CNN	85.88	96.72	72.31	3.27	82.99
LSTM	83.94	96.75	74.27	3.24	84.03
DNN	81.63	92.74	73.47	7.25	81.99

Table-1

9. CONCULSION

With increase in security threats to the Computer Network, it has become essential to develop an automatic network Intrusion Detection System solution to reduce these risks. The existing systems based on traditional methods and machine learning algorithms are not sufficiently effective for network intrusion detection problems. In this paper we have presented a comparative analysis of the evaluation metrics of several neural network algorithms to highlight their efficiency in detecting attacks. We trained models on KDDTrain+ dataset and evaluated performance on KDDTest+

dataset. The results show that the neural network based models perform excellently in the binary classification process in terms of accuracy, precision, recall, false alarm, and f1-score.

REFERENCES

[1] Devikrishna K S, Ramakrishna B B, "An Artificial Neural Network based Intrusion Detection System and Classification of Attacks" International Journal of Engineering Research and Application(IJERA), Vol. 3, Issue 4, Jul-Aug, pp. 1959-1964

[2] James Cannady, "Artificial Neural Networks for Misuse Detection", School of Computer Information Sciences, Nova Southeastern University, Fort Lauderdale, FL 33314, cannday@scis.nova.edu.

[3] L. P. Dias, J. J. F. Cerqueira, K. D. R. Assis, R. C. Almeida, "Using artificial neural network in intrusion detection systems to computer networks", IEEE Xplore, 9 Nov 2017

[4] Mohammad Masum, Hossain Shahriar, Hisham M., "A Transfer Learning with Deep Neural Network Approach for Network Intrusion Detection", Department of Information Technology, Department of Computer Science, Kennesaw State University, USA

[5] Karim Al-Saedi, Selva kumar Manickam, Sureswaran Ramadass, Wafaa Al-Salihy and Ammar ALmomani, "Research proposal An Intrusion Detection System Alert Reduction and Assessment Framework Based on Data Mining", Journal of Computer Science 9(4):421, April 2013

[6] https://www.javatpoint.com/artificial-neural-network

[7]https://www.sciencedirect.com/science/article/pii/S2405959518300493

[8] https://www.bmc.com/blogs/keras-neural-network-classification/

[9] https://keras.io/api/layers/core_layers/dense/

[10]https://searchsecurity.techtarget.com/definition/intrusion-detection-system

[11]https://www.fortinet.com/resources/cyberglossary/intrusion-detection-system

[12] https://www.digitalvidya.com/blog/types-of-neural-networks/

[13]https://www.mygreatlearning.com/blog/types-of-neural-networks/

[14] https://en.m.wikipedia.org/wiki/Neural_network

[15] https://medium.com/@sprhlabs/understanding-deep-learning-dnn-rnn-lstm-cnn-and-r-cnn-6602ed94dbff