Detecting anomaly-based network intrusions by using hybrid architectures of Convolutional Neural Networks

Abhishek Gokavarapu[1], DR. P.M. Manohar[2], B.S. Panda[3]

1 *Student, Dept of CSE, Raghu Engineering College, Visakhapatnam, A.P, India*
2 Associate Professor, Dept. of CSE, Raghu Engineering College (A), Visakhapatnam, Andhra Pradesh.
3 Professor, Dept. of CSE, Raghu Engineering College (A), Visakhapatnam, Andhra Pradesh

ABSTRACT

Software Defined Networking (SDN) is a promising technology for the future Internet. However, the Intrusion detection paradigm introduces new attack vectors that do not exist in the conventional distributed networks. This paper develops a hybrid Intrusion Detection System (IDS) by combining the Convolutional Neural Network (CNN) and Long Short-Term Memory Network (LSTM). The proposed model is capable of capturing the spatial and temporal features of the network traffic. Two regularization techniques i.e., L2 Regularization (L2Reg.) and dropout method are used to overcome with the overfitting problem. The proposed method improves the intrusion detection performance of zero-day attacks. The In KDD cup dataset — the most recent dataset for NSL-KDD networks is used to test and evaluate the performance of the proposed model. The results indicate that integrating the CNN with LSTM improves the intrusion detection performance and achieves an accuracy of 96.32%. The estimated accuracy is higher than the accuracy of each individual model. In addition, it is established that the regularization techniques improve the performance of the CNN algorithms in detecting new intrusions when compared to the standard CNN+LSTM. The findings of this study facilitate the development of robust IDS systems for SDN environment.

KEYWORDS: - *Deep Learning, Cyber security, Intrusion detection TensorFlow, Keras, Python, OpenCV.*

I. INTRODUCTION

The internet and society are deeply intertwined. The amount of data produced in response to this growing dependence is enormous. Information security is a big concern, so it's important to know how to spot attacks on networks, especially ones that haven't been seen before. An intrusion detection system, also known as an IDS, is capable of spotting both ongoing and past attacks. An intrusion detection system, also known as an IDS, is a device that sends out alerts whenever it detects suspicious activity in the network traffic. It is a piece of software that checks a system or network for malicious activity or policy violations. A security information and event management system are typically used to collect central reports of any malicious activity or violation. Although intrusion detection systems look for malicious activity on networks, they are also susceptible to false alarms. As a result, organizations must fine-tune their IDS products prior to installation. It entails properly configuring the intrusion detection systems to distinguish between malicious activity and normal network traffic.

1.1 Classification of Intrusion Detection System:

Intrusion detection system can be classified into five types.

i. Network Intrusion Detection System:

In order to examine traffic from all network devices, network intrusion detection systems, or NIDS, are set up at a designated location within the network. It matches the traffic that is passed on the subnets to the collection of known attacks by observing the passing traffic across the entire subnet. The alert can be sent to the administrator once an attack or abnormal behavior is discovered. Installing a NIDS on the subnet where firewalls are situated to determine whether someone is attempting to breach the firewall is one example of a NIDS.

ii. Host-Based Intrusion Detection System:

Host intrusion detection systems, or HIDS, are independent network hosts or devices. Only the device's incoming and outgoing packets are monitored by a HIDS, which will notify the administrator of any suspicious or malicious activity. It compares the current snapshot with the previous snapshot by taking a snapshot of the system files. An alert is sent to the administrator for investigation in the event that the files in the analytical system were altered or deleted. On mission-critical machines, which are not expected to change their layout, HIDS is used in an example.

iii. Protocol-Based Intrusion Detection System:

A protocol-based intrusion detection system, also known as a PIDS, consists of a system or agent that always resides at the front end of a server and is responsible for controlling and interpreting the protocol that is exchanged between a user's device and the server. By regularly accepting the relevant HTTP protocol and monitoring the HTTPS protocol stream, it is attempting to protect the web server. Since HTTPS is unencrypted and must be used in this interface before entering the web presentation layer, this system must use HTTPS.

iv. Application Protocol-Based Intrusion Detection System:

A system or agent known as an Application Protocol-based Intrusion Detection System (APIDS) typically resides within a group of servers. It monitors and interprets application-specific protocol communication to identify intrusions. This could, for instance, keep an eye on the SQL protocol that is specific to the middleware as it interacts with the database in the web server.

v. Hybrid Intrusion Detection System:

The hybrid intrusion detection system is created by combining two or more intrusion detection system approaches. Data from the host agent or system and information from the network are combined in the hybrid intrusion detection system to provide a comprehensive view of the network system. In comparison to the other intrusion detection system, the hybrid one is more efficient. Hybrid IDS can be seen in Prelude.

1.2 Detection methods of IDS

Two detection methods are there in intrusion detection system. Those are

a) Signature based IDS:

Attacks are identified by signature-based IDS based on specific patterns in network traffic, such as the number of bytes, ones, or zeros. Additionally, it does so based on the malware's well-known malicious instruction sequence. Signatures are the IDS's patterns that are found.

b) Anomaly based IDS:

A trustful activity model is created using machine learning in anomaly-based IDS, and anything that is compared to that model is deemed suspicious if it does not match the model. In comparison these models can be trained based on applications and hardware configurations

II. LITERATURE SURVEY

In previous studies, intrusion detection systems utilized a variety of approaches. An improved intrusion detection strategy, a three-layered RNN based on various features, was addressed by the authors in [3]. The model received features that are classified based on basic features, content features, time-based traffic features, and host-based traffic features as its inputs. As its output, the model either classifies attacks such as DoS, R2L, U2R, and probing or the normal class, which indicates that there are no attacks. They also talked about how they trained and tested the model using the KDD dataset and how the connections between the hidden layers, which are only partially connected, speed up the classification process. This paper makes use of misuse-based intrusion detection, which compares user actions to known attacker behaviors.

In [5], a new strategy using a Gated Recurrent United Recurrent Neural Network (GRU-RNN) to detect network intrusion was presented. The authors have attempted to provide a novel, low-feature-count solution for Network Intrusion Detection (NID) in Software Defined Networks (SDN). According to the authors, the use of GRU-RNN over conventional RNN or LSTM has the advantage of avoiding vanishing and exploding gradient issues. In addition, the authors have used the NSLKDD dataset to train and test their model and have implemented their system as an application on the SDN controller. With a detection rate of 89 percent for legitimate events and 90 percent for anomalous events, the proposed GRURNN outperformed other machine learning algorithms like Vanilla RNN, SVM, and DNN. In addition, the author asserted that the proposed solution's overall accuracy of 89 percent was the highest among the other modern algorithms discussed. A study in [6] that discusses and investigates network intrusion detection using Convolutional Neural Networks (CNN) was carried out using the CNN algorithm. A CNN-based behavior-based classifier learning model has been created for the study. Using statistical data, the CNN, TensorFlow, and SoftMax functions are used to extract behavioral features and classify threats. A benchmark set of data from sources like KDD Cup99's archive dataset of behavior features and NHSNC's suspicious network flow order are created during the extraction phase.

The gradient-descent optimization algorithm and a revised LeNet-5 model are utilized in the current study's model learning phase. By utilizing both the learning rate for all layers and the error derivatives of back propagation, these algorithms make it easier to fine-tune the model parameters. In a nutshell, these classifications aid in drawing the learning errors of multiple layer neural nets and reducing the weights of the neural network, both of which ultimately affect CNN's learning process.

In addition, the authors of paper [8] investigated and proposed a four-step neural network intrusion detection strategy that makes use of the concepts of the Deep Brief Network (DBN), the probabilistic neural network (PNN), and the particle swarm optimization (PSO) algorithm. They used the idea of deep learning as the first step because the raw data is transformed into low-dimensional data, which reflects the data's characteristics. A multilevel or deep neural network can accomplish this Additionally, the data normalization is ensured by the multi- stacked Restricted Boltzmann Machine (RBM) formation of these DBN. The PSO algorithm was used to optimize the data in the following step, which facilitated the network's learning performance by optimizing data's hidden layer nodes. Finally, data will be fed into a PNN for training and testing because it uses a Gaussian and network activation function-based local approximation network to produce optimized data. In addition, the authors of [9] employed a hybrid detection strategy that combines restricted Boltzmann machines (RBM) and RNN. To begin, they used a multi-layer RBM model with raw data inputs at the byte level to obtain network packet feature vectors. After that, the micro flow features are created by modelling a series of packets using the RNN model. Without feature engineering, the proposed model takes raw data as its input. Last but not least, a SoftMax classification is used to determine whether a micro-flow is malicious or not. They also conducted a series of experiments on the effects of the 1-5-layer RBM model, including the convergence rate and a variety of evaluation indicators, to determine the required number of layers of RBM features. Additionally, they have stated that the RBM model's convergence rate decreases as the number of layers increases, necessitating additional epochs for superior outcomes. As a result, they suggested employing a three-layered RBM model.

2.1 Summary of the literature review

From the literature survey the points to be noted are Recent studies proposed so many new techniques to detect and prevent the intrusions to protect network security.

The model received features that are classified based on basic features, content features, time-based traffic features, and host-based traffic features as its inputs. They also talked about how they trained and tested the model using the KDD dataset and how the connections between the hidden layers, which are only partially connected, speed up the classification process. The authors have attempted to provide a novel, low-feature-count solution for Network Intrusion Detection (NID) in Software Defined Networks (SDN).

By utilizing both the learning rate for all layers and the error derivatives of back propagation, these algorithms make it easier to fine-tune the model parameters. In a nutshell, these classifications aid in drawing the learning errors of multiple layer neural nets and reducing the weights of the neural network, both of which ultimately affect CNN's learning process.

III.PIPELINE FLOW

In the proposed work, by using Convolutional Neural Networks with the two RNN types of GRU and LSTM, we will attempt to provide a straightforward and effective technique to identify intrusions. Select the NSL-KDD dataset for testing and training in this experiment. The final predictions shown in both binary and multiclassification results.

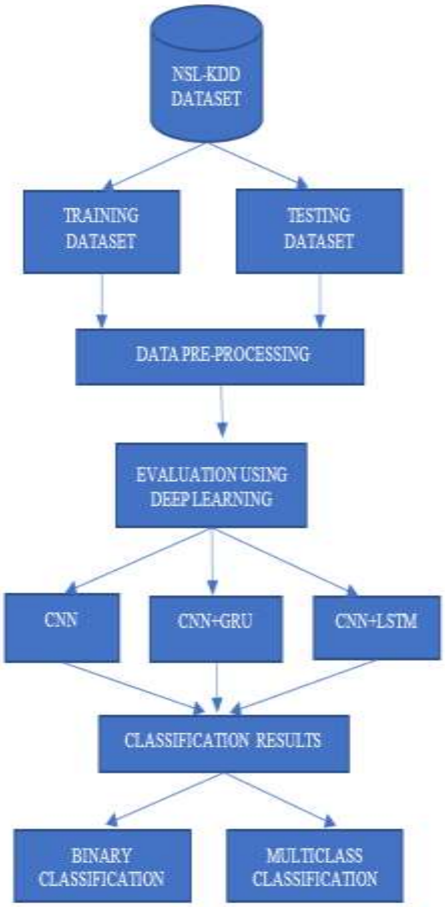


Fig 3.1 Proposed methodology diagram

The above diagram shows the methodology diagram for proposed methodology of network intrusion detection system. Firstly, the NSL-KDD dataset is taken and it have two parts of data, one is training dataset and another one is testing dataset.

3.2 Data Preprocessing

Data preprocessing, a component of data preparation, describes any type of processing performed on raw data to prepare it for another data processing procedure

3.3 Numericalization

There are 38 numerical features and three non-numerical features in the NSL-KDD dataset. We convert the non-numeric features into numeric values because the input values must be numerical. For instance, the feature "protocol type" can have three distinct types of attributes: "TCP," "UDP," ICMP," respectively. They are encoded using the binary vectors (0,0,1), (1,0,0), and the 41-dimensional feature map is transformed into a 122-dimensional one in this manner.

3.4 Normalization

The dataset contains a number of features whose maximum and minimum values differ significantly. dst_bytes [0,1.3 109], src_bytes [0,1.3 109], and duration[0,58329] are examples of such features. After reducing the differences using the logarithmic scaling technique, we use the formula below to map them to the [0,1] range:
xi = (xi - Min) / (Max - Min) Algorithm for methodology of project:
Input Dataset: NSL-KDD dataset.
Number of features: 42 features.
Number of records: 125973 records for training and 22525 for testing.

Main Steps:

Step 1: Importing Essential libraries

Step 2: Making dictionary of labels

Step 3: Upload dataset, reading and processing data

Step 4: Normalization and Label encoding

Step 5: Model building (CNN, CNN+GRU, CNN+LSTM)

Step 6: Setting optimizer and fitting the model and finally model evaluation

3.5 Algorithm for convolutional neural networks (CNN) model

Step 1: Upload dataset

Step 2: Input layer: Input is given to the CNN layer for next processing.

Step 3: Convolutional layer: In order to calculate the output, the convolutional kernel or filter only moves in one direction, such as along the time axis.

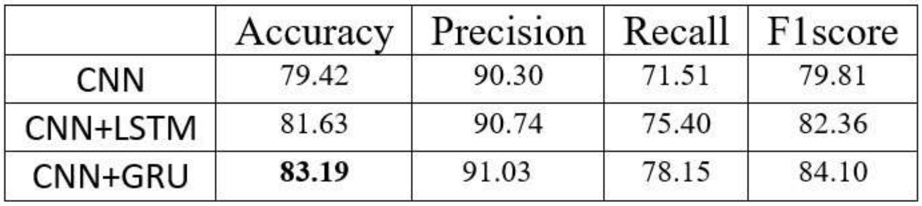
Step 4: Pooling layer: pooling refers to pooling all the samples of equal volume in a row or a column.

Step 5: dense layer: The neurons of the layer are connected to every neuron of its preceding layer

IV. RESULT ANALYSIS

Binary classification results:

In binary classification, 41-dimensional features have been mapped to 83-dimensional features. In the binary classification experiment, the CNN-IDS model has 122 input nodes and 2 output nodes.



	Accuracy	Precision	Recall	F1score
CNN	79.42	90.30	71.51	79.81
CNN+LSTM	81.63	90.74	75.40	82.36
CNN+GRU	**83.19**	91.03	78.15	84.10

Table 4.1 Binary classification results

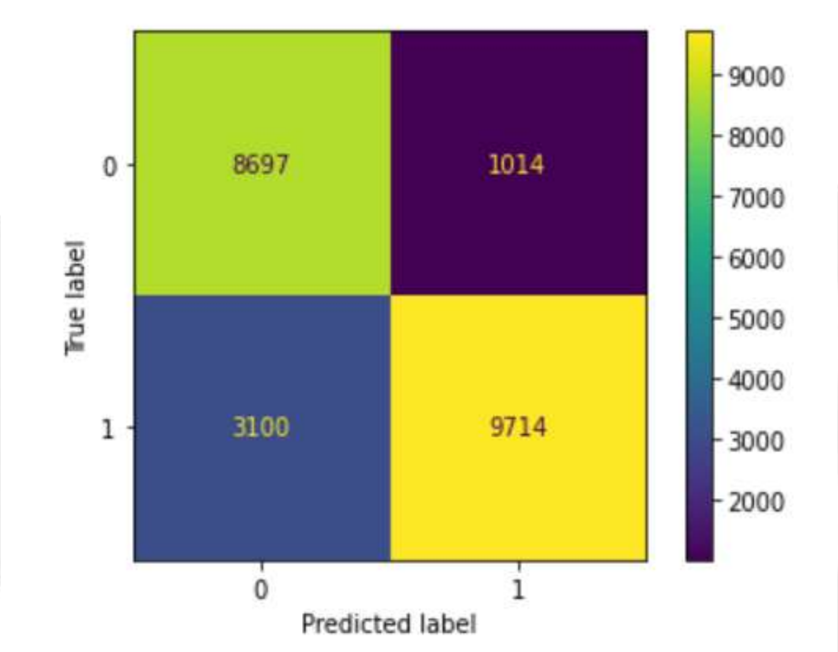


Chart 4.2 Confusion matrix of binary classification

The above diagram shows the confusion matrix of binary classification. 0 is denoted as normal activity and 1 is denoted as malicious activity(attack) by using the NSL-KDD dataset.

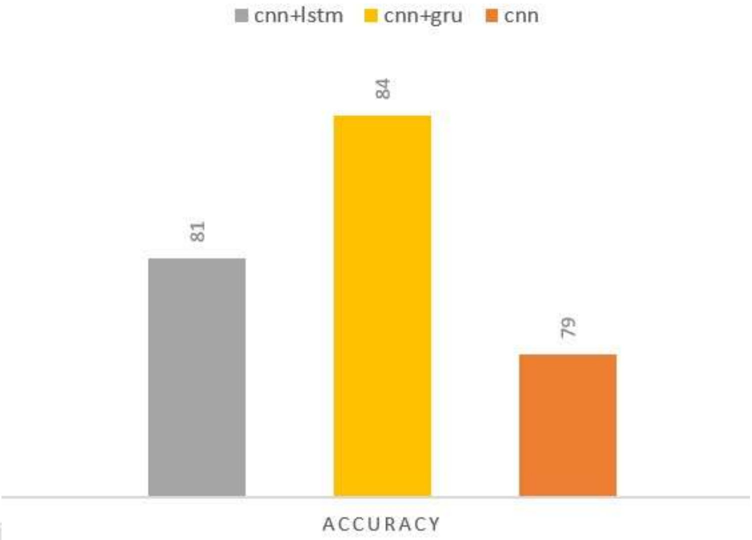
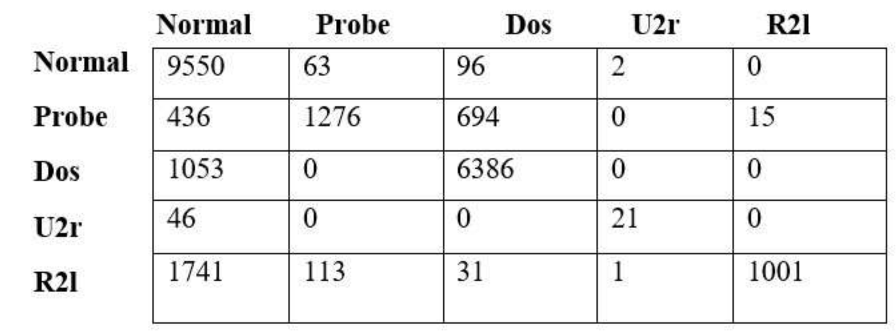


Chart 4.3 Binary classification results graph

Multi-Class Classification results:

In this Multi-Class classification, identifying the different multiclass attacks (normal, dos, probe, r2l ,u2r) accurately by using the CNN-IDS model.



	Normal	Probe	Dos	U2r	R2l
Normal	9550	63	96	2	0
Probe	436	1276	694	0	15
Dos	1053	0	6386	0	0
U2r	46	0	0	21	0
R2l	1741	113	31	1	1001

Table 4.4 Confusion matrix for multiclass classification

Above table shows that 9550 are detected as “Normal”, 1276 are detected as “Probe”, 6386 detected as “DOS”, 21 are detected as “U2R” and 1001 are detected as “R2L”. the accuracy of different multiclass attacks is shown below.

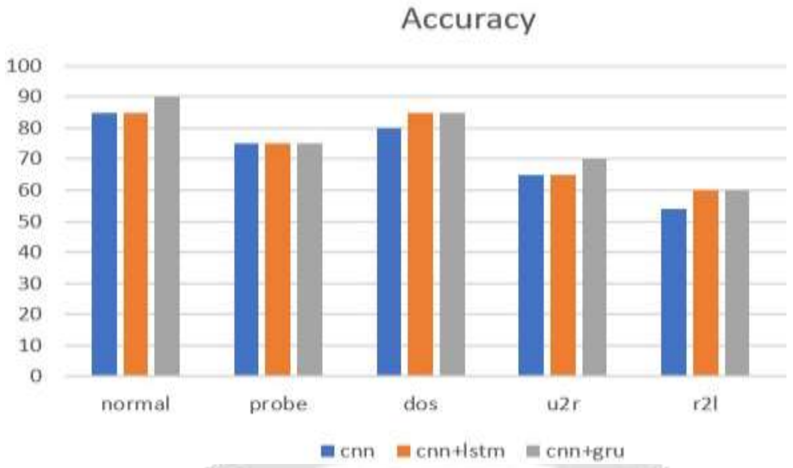


Chart 4.5 Multi Classification Results Graph

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed two models multiclass and binary classification, and we proposed in these models to using deep learning techniques for detecting networks attacks instead of using machine learning rules or signatures. Through this experimental research of classification which had been found in the NSL-KDD dataset, and we have shown that hybrid architecture of CNN (convolutional neural networks are combined with long- short term memory and gated recurrent unit) are capable of detecting and classifying with accuracy 83.1%. In multiclass classification the dos attacks are detecting with better accuracy than remaining attacks. In future work, better to implement by considering the large datasets because the hybrid architecture of deep learning models performs better on large amount of data. And also, by identifying relevant features inside the dataset we can increase the accuracy

VI. REFERENCES

[1] D. E. Denning, “An intrusion-detection model,” IEEE Transactions on Software Engineering, vol. SE-13, no. 2, pp. 222–232, Feb. 1987, ISSN: 0098-5589. DOI: 10.1109/TSE.1987.232894.

[2] M. K. Asif, T. A. Khan, T. A. Taj, U. Naeem, and S. Yakoob, “Network intrusion detection and its strategic importance,” in 2013 IEEE Business Engineering and Industrial Applications Colloquium (BEIAC), Apr. 2013, pp. 140–144. DOI: 10 . 1109 / BEIAC . 2013 . 6560100.

[3] M. Sheikhan, Z. Jadidi, and A. Farrokhi, “Intrusion detection using reduced-size rnn based on feature grouping,” Neural Computing and Applications, vol. 21, no. 6, pp. 1185–1190, 2010. DOI: 10.1007/s00521-010-0487- 0

[4]] Saporito, Gerry. "A Deeper Dive into the NSL-KDD Data Set" Medium, https://towardsdatascience.com/a-deeper-dive-into-the-nsl-kdd-data-set15c753364657. Accessed 17 September 2019

[5] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, “Deep recurrent neural network for intrusion detection in sdn-based networks,” in 2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft), Jun. 2018, pp. 202–206. DOI: 10. 1109/NETSOFT.2018.8460090.

[6] W. Lin, H. Lin, P. Wang, B. Wu, and J. Tsai, “Using convolutional neural networks to network intrusion detection for cyber threats,” in 2018 IEEE International Conference on Applied System Invention (ICASI), Apr. 2018, pp. 1107–1110. DOI: 10 . 1109 / ICASI . 2018 . 8394474.

[7] Jin Kim, Nara Shin, Seung Yeon Jo and Sang Hyun Kim, "Method of Intrusion Detection using Deep Neural Network," IEEE, 2017.

[8] G. Zhao, C. Zhang, and L. Zheng, “Intrusion detection using deep belief network and probabilistic neural network,” in 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), vol. 1, Jul. 2017, pp. 639–642. DOI: 10.1109/CSE-EUC.2017.119

[9] C. Li, J. Wang, and X. Ye, “Using a recurrent neural network and restricted boltzmann machines for malicious traffic detection,” NeuroQuantology, vol. 16, no. 5, 2018, ISSN: 1303-5150. [Online]. Available: www.neuroquantology.com/index.php/journal/article/ view/1391