Volume-11, Issue-7, July, 2022
International Journal ofEngineering Associates (IJEA)
Page Number: 01-06

DETECTING INFILTRATION AND INTRUSIVE PERFORMANCE IN WIRELESS NETWORKS USING DL & LSTM

Aastika Tiwari[1], Prof. Amit Namdev[2]
[1]M.Tech Scholar, [2]Assistant Professor
[1,2]Mittal Institute of Technology, Bhopal, M.P., INDIA
[1]aastika29@gmail.com, [2]amit.namdev1811@gmail.com

***Abstract*— Concerns about network access, integrity, and confidentiality are growing as computer networks become more widely used. When it comes to detecting unauthorised and malicious activity on a network, it's necessary to use a variety of intrusion detection systems (IDS). When someone deliberately violates a security policy, this is referred to as an intrusion. Consequently, intrusion detection systems monitor network traffic flowing thru computer systems in order to detect malicious actions as well as recognised dangers, generating alarms when they detect them. A deep learning-based intrusion detection system is the goal of this paper. Consequently, this effort aims to develop an intrusion detection and prevention system which uses Deep Learning to recognise and block attacks such as DOS, Probe, R2L, or U2R. One-dimensional CNN and Long Short-Term Memory (LSTM) are used to identify attacks on the KDD99 dataset, according to the authors (LSTM). The proposed model's effectiveness on binary and multiclass classifications is evaluated using the KDD99 datasets. An accuracy rate of 99 percent was recently achieved using the method we proposed. When compared to the current method, the proposed hybrid's accuracy improved by approximately 14 percent. This means that a small number of epochs can be used to achieve optimum accuracy.**

Keywords* - *DOS, Probe, R2L, U2R, One-dimensional, CNN and Long Short-Term Memory (LSTM).

1. INTRODUCTION

Computer attacks are becoming more diverse and more common as a result of the Internet's continued growth. Media coverage of ransomware attacks and zeroday exploits is on the rise because they're becoming so important. It is no longer sufficient to rely solely on antivirus software and firewalls to protect a company's network. IDS is one of the most critical layers, designed to protect its target from any possible threat by constantly monitoring the system (IDS). Anomaly detection as well as signature-based detection, furthermore known as "misuse detection," are the two main types of IDS currently in use. Signature-based detection is achieved by comparing IDS data to known attack patterns. Many security tools use this method, but it has one major flaw: it can only identify attacks that have already been recorded in a database. While anomaly detection builds a model of the game's typical behaviour prior to actually searching for anomalies in monitored data[1]. As a result, while this method can detect unknown attacks, it also generates a large number of false alarms.

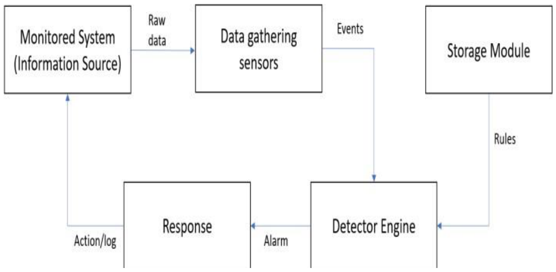


Fig. 1 : Intrusion Detection System architecture

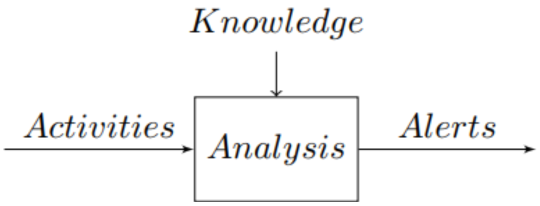


Fig. 2 : The Intrusion Detection System works.

A. Networking Attacks

An overview of the four primary types of networking assaults is given in this section. Each network attack can be categorized into one of these categories with ease. Denial of Service - DoS attacks, including Neptune, mail bomb, back, apache, UDP storm, smurf, and ping, are the first and most harmful types of assaults in which a hacker prevents a computer from responding to network requests because memory resources are overloaded.

Remote to User Attacks (R2U): In these attacks, a user sends packets to a system over the internet in order to discover its security holes and use privileges that a local user would normally have, such as xnsnoop, dictionary, guest, xlock phf, sendmail, and so forth, to gain access to the system.

User to Root Attacks (U2R):

These attacks on the system start with a standard user account and try to take advantage of system flaws like xterm and perl to gain super user privileges.

Probing - These attacks look for flaws or vulnerabilities in a machine or a networking device that can be later used to compromise the system. This technique is utilised by data mining tools like Portsweep, MScan, Nmap, Saint, and others.

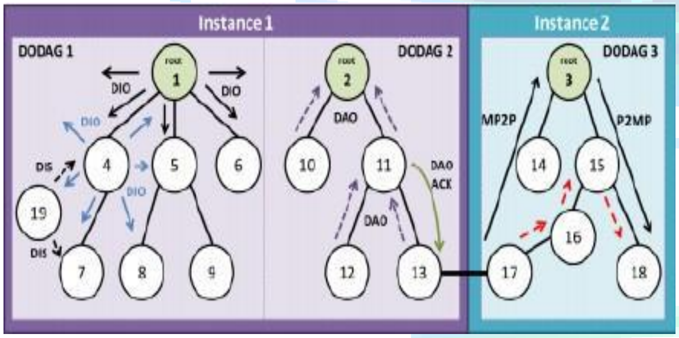


Fig. 3: Example of a RPL network with two instances and three DODAGs

.

II. LITERATURE REVIEW

K. Muthamil Sudar et al. SDN (software-defined network) is a neural network that can be used to build, design, and test hardware components in a simulated environment. Network connections can have their settings changed on-the-fly. It's impossible to change dynamically in a traditional network because the connection is fixed. DDoS attacks can still cripple SDN, despite its advantages. The internet is in danger as a result of the DDoS attack. The machine learning technique can be utilised to thwart a DDoS attack. There are a number of systems working together to attack a single server at the same time, known as a distributed denial of service (DDoS). In SDN, the devices in the infrastructure layer are controlled by software via the control layer, which is connected to both the application and the infrastructure layers. Detecting malicious traffic using Decision Tree as well as Support Vector Machine (SVM) is what we propose in this paper. The results of our research show that the Decision Tree as well as Support Vector Machine (SVM) algorithm provides greater accuracy and detection rate[24].

Pratik Gite et al. Building an idss into a WSN is essential for its security (IDS). In this article, we'll discuss a variety of WSN security issues. It suggests a method for detecting malicious nodes in a wireless sensor network (WSN). Base Station machine learning (ML) is used to identify four types of attacks: black hole, wormhole, grey hole and distributed denial of service (DDoS) assaults (BS). Each node's data is constantly analysed by the proposed ML algorithm. In order to prevent an attacker, BS identifies the network's harmful behaviour and sends warnings to its neighbours. To begin with, the various attacks are analysed and their characteristics are derived throughout terms of network parameters. Using this data, a machine learning algorithm can be built on top of it. An attack is then effectively and accurately classified as coming from an attacker node in BS. Simulated secure WSNs were created using the NS2 simulator. The results of the experiments showed that the proposed attack detection method has excellent accuracy. This level of accuracy has the potential to improve network efficiency, both in terms of power consumption and packet delivery (PDR)[25].

III. PROBOLEM FORMULATION

Following are the various research gaps

1. The techniques which are proposed to improve security of the data routing can have high latency. A technique needs to be proposed which should be light weighted so as to improve security of the network.
2. The routing technique which is already proposed can establish path from source to destination but version number attack is still possible which affects network performance.
3. A novel method needs to be proposed which helps in not only the detection but also isolation of malicious nodes from the network in the least amount of time and increased accuracy.

IV. PROPOSED MODIFIED

Because of the importance of networks in modern life, cyber security has emerged as an important research area. a system for detecting intrusions (IDS)[57] , An essential method for ensuring that all of the network's software and hardware are functioning properly. There are still challenges with current intrusion detection systems, even after decades of research on how to increase detection accuracy while decreasing false alarm rates and identifying unexpected threats A number of academics have devoted themselves to designing IDSs based on Deep Learning methodologies to address the aforementioned concerns. In the field of machine learning, "deep learning" is a branch with exceptional performance. The KDD99 dataset, which is freely available, was used in this study. EDA is then used to visualise the data. Data preprocessing techniques were used in order to check for null values, remove duplicates, values that changed to scalar, and finally extract the data features from the input dataset data preprocessing techniques have been completed F1-scores, accuracy and precision are measured in terms of the proposed hybrid model that includes 1DCNNs and an LSTM neural network. The proposed model's accuracy can be seen in the results section below.

Proposed Methodology Step:

Step 1: Dataset name KDD99
Collect the dataset, this dataset contains intrusion website information.
Step 2: Performing EDA on the dataset and get to know that it can be done as binary classification and multi-class classification.
Step 3: Processing
 Dropping Null values.
Removing duplicate Values
Changing To scalar values
Feature Extraction
Step 4: Plotting graphs and done final processing on the data for the training.
Step 5: Creating a Hybrid Deep Learning model and fitting the data to it, let it train. After completion, use the model for testing.
Step 6: Evaluation of the model, testing the model on the test set and measuring the performance in terms of precision, recall and F1-Score. The Hybrid Deep learning model performed very well

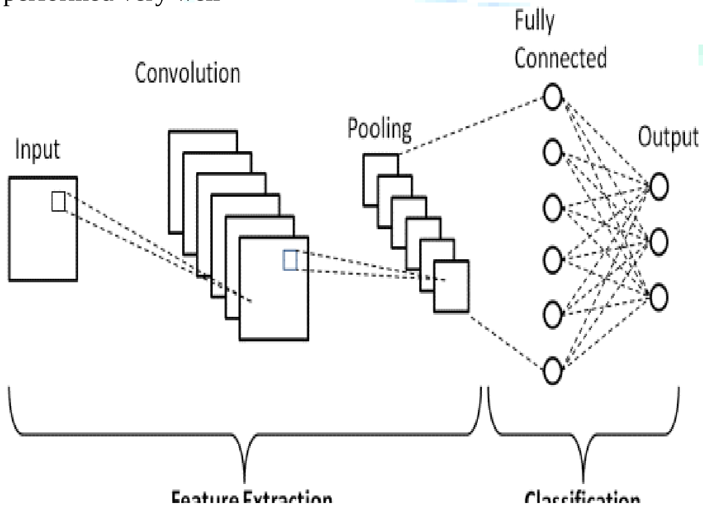
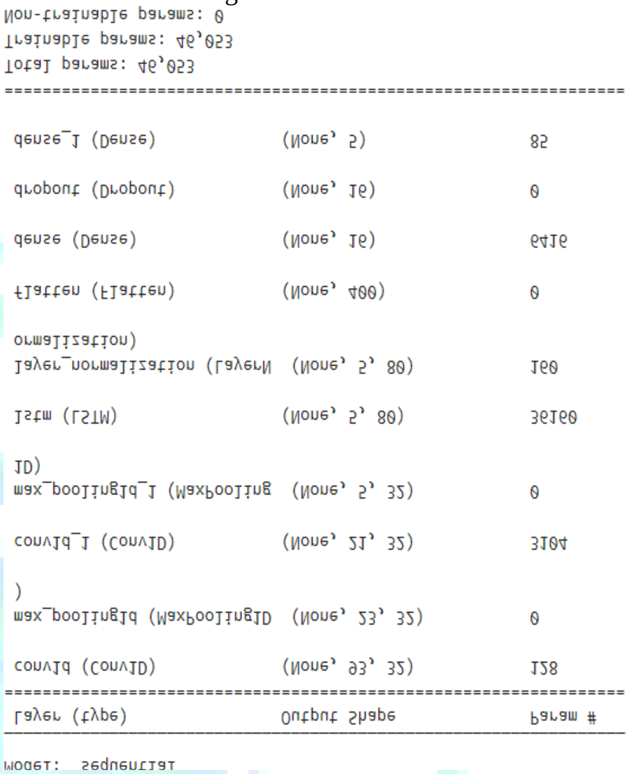


Fig 4 : CNN architecture



```
Model: "sequential"
_________________________________________________________________
Layer (type)                Output Shape              Param #
=================================================================
conv1d (Conv1D)             (None, 93, 32)            128

max_pooling1d (MaxPooling1D  (None, 23, 32)           0
)

conv1d_1 (Conv1D)           (None, 21, 32)            3104

max_pooling1d_1 (MaxPooling  (None, 5, 32)            0
1D)

lstm (LSTM)                 (None, 5, 80)             36160

layer_normalization (LayerN  (None, 5, 80)            160
ormalization)

flatten (Flatten)           (None, 400)               0

dense (Dense)               (None, 16)                6416

dropout (Dropout)           (None, 16)                0

dense_1 (Dense)             (None, 5)                 85

=================================================================
Total params: 46,053
Trainable params: 46,053
Non-trainable params: 0
```

Figure 5 : Hybrid Proposed Model Summary

IV. RESULTS & DISCUSSION

***A.* Dataset Analysis with EDA :-** In the field of IDS research, the KDD99 is a frequently used tool. In the dataset, there are 41 characteristics that fall into one of the following five categories: Normal, DoS, Probe, Remote-to-Local (R2L), or User-to-Root (U2R) (U2R). The KDD99 (ten percent variation) has 494,021 records in the training set and 311,029 records in the testing set. Class imbalances are found in both training and testing sets of the KDD99. The DoS class has the most records, with the Normal class in second place. More records are classified as R2L than any other type in the testing set. An investigation revealed a large number of records in this collection were duplicates.Exploratory Data Analysis[65] [66], Any project involving data analysis or data science will require EDA at some point. Based on what we know about the dataset, we look for patterns, outliers, and hypotheses in the data. EDA creates statistical results for the dataset's statistical information and creates a variety of graphical in order to better understand the data. This paper uses the KDD99 dataset to shed light on EDA. using EDA on the dataset to discover that EDA can be done in binary or multi-class mode. Python is used for this task. Figures 13 to 16 show all EDA graphs based on the KDD99 dataset, as shown in the following figure.

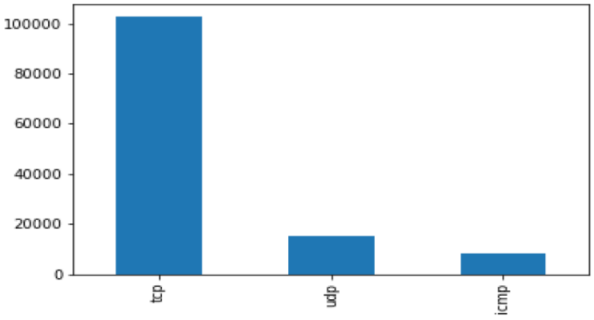


Fig. 6: Bar graph of Protocol Type

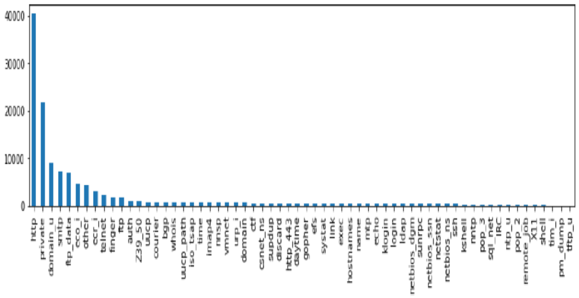


Figure 7: Every Sevrcie Graph

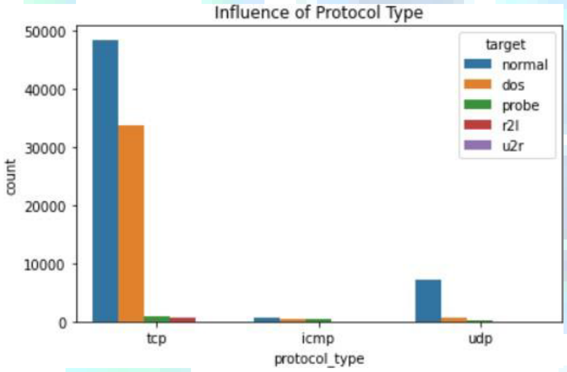


Fig. 8 : Protocol type influence on target

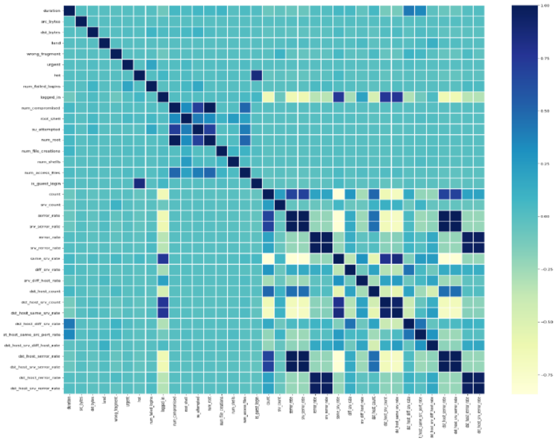


Fig. 9: Correlation between whole data

Coefficients of correlation for different variables are shown in a correlation matrix. Each row of the matrices represents a potential correlation between each of the rows of data. Figure 5.4shows how it can be used to summaries a large dataset, and it can be used to identify trends in the data.

B. Evaluation Metrics

There are a number of criteria that are used to evaluate machine learning algorithms. These criteria are used to select the best models. Detection impact is often quantified using a variety of metrics in IDS research.

The Accuracy, Prediction, Recall, and F1-Score metrics are used to evaluate the experimental model's performance. Flow identification accuracy and false alarm rates are measured using these assessment criteria. Model prediction results and the true label can be combined in four different ways: The term "False Negative" (FN) refers to a positive sample that has been misinterpreted as a negative sample. When negative samples are mistakenly identified as positive, the result is a false positive (FP). Samples that are genuinely negative (TN) are analysed as such and are interpreted as such. To be considered positive, True Positive (TP) samples must be present. These values are derived using Equations 1-6.

Accuracy:

The percentage of samples that have been correctly identified is known as the sample identification rate. Accuracy is a good metric to use when the dataset is well-balanced. However, in reality, normal samples outnumber aberrant samples, so accuracy may not be an acceptable statistic.

$$\text{Accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)} \ldots (1)$$

Precision (P):

To put it another way, it is a ratio between the number of positive samples and the number of expected positive samples.

$$\text{Precision} = \frac{(TP)}{(TP + FP)} \ldots (2)$$

Recall (R):

Recall is determined by dividing the total number of positive samples by the number of genuine positive samples. The detection rate is a critical parameter in intrusion detection systems because it represents the model's ability to identify attacks.

$$\text{Recall} = \frac{(TP + TN)}{(TP + TN + FP + FN)} \ldots (3)$$

F-measure (F) is calculated by taking the harmonic average of the precision and recall values.

$$\text{F1} - \text{Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{(\text{Precision} + \text{Recall})} \ldots (4)$$

The false negative rate (FNR)

Erroneous negative results are measured as a percentage of all positive results. When discussing assault detection, the FNR is also referred to it as the missed alert rate.

$$\text{FNR} = \frac{(\text{FN})}{(\text{TP} + \text{FN})} \ldots (5)$$

The false positive rate (FPR) Is defined as the percentage of false positive samples compared to true positive samples. The false alarm rate (FPR) is also known as the false alarm rate when it comes to attack detection.

$$\text{FPR} = \frac{(\text{FP})}{(\text{TP} + \text{FP})} \ldots (6)$$

C. Experimented Results

Some more traditional machine learning and deep learning classification techniques, such as 1DCNN and Long Short-Term Memory, are used in this experiment (LSTM). As well as compared to other strategies already in use. A variety of graphs, metrics, and tables are used to present the results of the experiment. In the time since the experiment, we've gone over all of the results in great detail. Classification and feature extraction were both addressed using a deep learning model developed as part of this research.

Model	Sequential
Hybrid Model	1D-CNN-LSTM
Pooling	Max pool
Activation	Relu and Softmax

Optimizer	Adam
Loss	Categorical cross entropy
Epochs	10
Batch size	512
Train data	94479, 93, 1
Test data	31494, 93, 1
Train, test ratio	80:20
Metrics	Accuracy

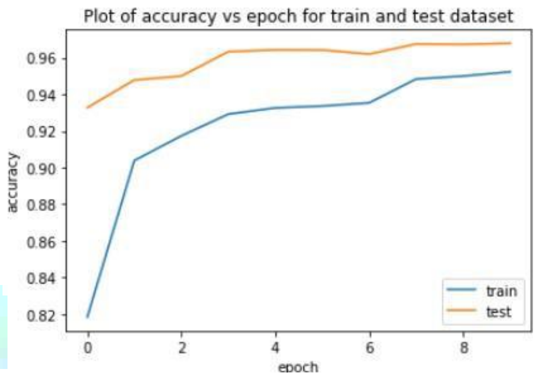


Fig. 10 Plot of vs epochs for train and test data

Testing and training results are shown in Figure 10. Data points are plotted along two lines, one for each epoch, and the other for the accuracy value. During training on the training dataset, the model's accuracy in identifying the two inputs is measured by its training accuracy. As a result, the training loss is a measure of how well the model relates to the training data.

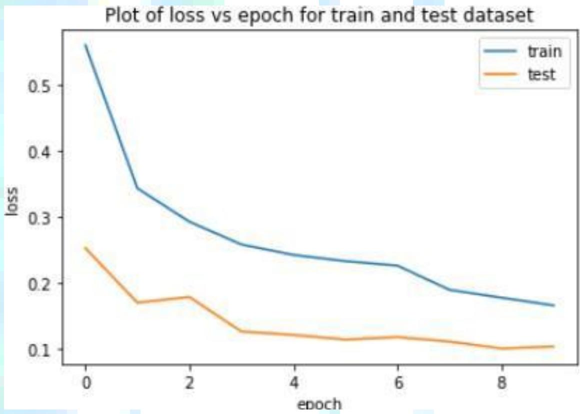


Fig. 11: Plot of vs epochs for train and test data

The developed framework's train and test losses are shown in Figure 11. Ten training epochs were used to fine-tune the model. After each iteration of optimization, the train loss value shows how well or how poorly a model performs The algorithm's efficiency is frequently assessed using a test loss measure.

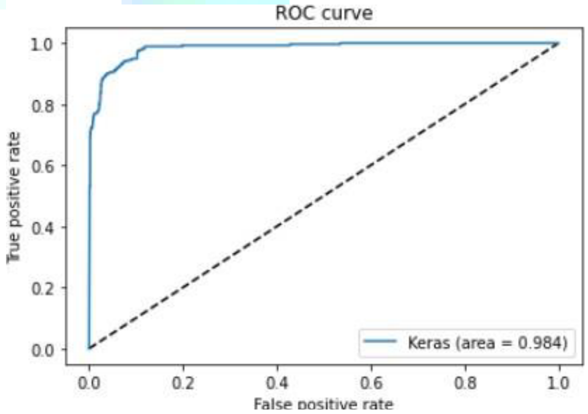


Figure 12: ROC curve of hybrid model\

Figure 12 depicts the hybrid mode ROC Curve's confusion matrix. The ROC curve, or receiver operating characteristic curve, is a graph that shows how well a classification model performs across all levels of classification.

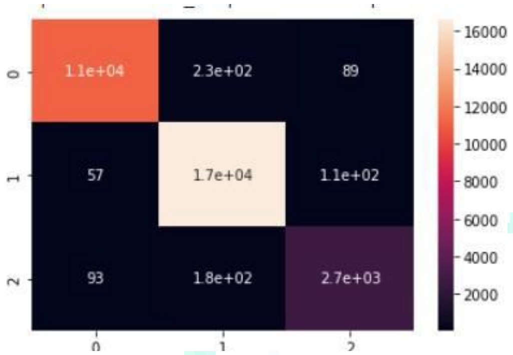
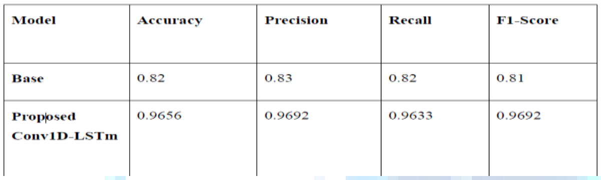


Fig. 13: Confusion matrix of hybrid model

Figure 13 depicts the hybrid DL model's confusion matrix. When using the confusion matrix, the true positive values for 0 label, 1 label, and 2 labels are represented by the diagonal value of 11e+04, 17e+04, and 2.7e+03, respectively

.

Table :II Comparison between base and proposed model using performance parameters.



Model	Accuracy	Precision	Recall	F1-Score
Base	0.82	0.83	0.82	0.81
Proposed Conv1D-LSTm	0.9656	0.9692	0.9633	0.9692

Accuracy and precision are shown in table II above along with f1-score for the base model and a proposed model based on the aforementioned performance parameters. The base model's accuracy, precision, recall, and f1-score are all 82%, while my hybrid model's accuracy, precision, and f1-score are all 96%, and recall is 96%, respectively, while my proposed hybrid model's recall is 81%. My proposed 1DCNN and LSTM hybrid model outperforms the base model in this regard.

V.CONCLUSION AND FUTURE WORK

Detection of network intrusions is becoming increasingly important as network attack methods advance. On account of the unequal network traffic, intrusion detection systems are having difficulty anticipating the propagation of malicious attacks, placing cyberspace security at serious risk. Unbalanced network data can be improved by a distinctive hybrid deep learning (IDCNN with LSTM) developed in this study. It is necessary to learn more minority samples so that the internet traffic imbalance and the minority's capacity for learning under difficult samples can be improved. IDCNN and

LSTM deep learning algorithms were used in this project. Using unbalanced network traffic as a testbed, we were able to find the samples that needed to be extended and improve attack detection rates. We found that deep learning outperformed the current algorithms during the trial. Some datasets for deep learning have been preprocessed and therefore do not enable for automated extracting features, deny the reality that neural nets help to improve the data's representation.

Using a deep learning method for extracting features and training just on original data, we are hoping to take advantage of deep learning's advantages while also reducing the amount of unbalanced data. A new dataset and a responsive IDS approach will be developed using real-world traffic on a network.

REFERENCES

[1.] V. B. Reddy, A. Negi, S. Venkataraman, and V. Raghu Venkataraman, “A Similarity based Trust Model to Mitigate Badmouthing Attacks in Internet of Things (IoT),” *IEEE 5th World Forum Internet Things, WF-IoT 2019 - Conf. Proc.*, pp. 278–282, 2019, doi: 10.1109/WF-IoT.2019.8767170.

[2.] R. Vishwakarma and A. K. Jain, “A honeypot with machine learning based detection framework for defending IoT based botnet DDoS attacks,” *Proc. Int. Conf. Trends Electron. Informatics, ICOEI 2019*, no. Icoei, pp. 1019–1024, 2019, doi: 10.1109/ICOEI.2019.8862720.

[3.] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” *Comput. Networks*, vol. 54, no. 15, pp. 2787–2805, 2010, doi: 10.1016/j.comnet.2010.05.010.

[4.] J. Deogirikar, “Security Attacks inIoT : A Survey,” pp. 32–37, 2017.

[5.] L. Liang, K. Zheng, Q. Sheng, W. Wang, R. Fu, and X. Huang, “A denial of service attack method for iot system in photovoltaic energy system,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10394 LNCS, pp. 613–622, 2017, doi: 10.1007/978-3-319-64701-2_48.

[6.] K. N. Qureshi, S. S. Rana, A. Ahmed, and G. Jeon, “A novel and secure attacks detection framework for smart cities industrial internet of things,” *Sustain. Cities Soc.*, vol. 61, p. 102343, 2020, doi: 10.1016/j.scs.2020.102343.

[7.] N. Zhang, R. Wu, S. Yuan, C. Yuan, and D. Chen, “RAV: Relay Aided Vectorized Secure Transmission in Physical Layer Security for Internet of Things under Active Attacks,” *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8496–8506, 2019, doi: 10.1109/JIOT.2019.2919743.

[8.] M. Ingham, J. Marchang, and D. Bhowmik, “IoT security vulnerabilities and predictive signal jamming attack analysis in LoRaWAN,” *IET Inf. Secur.*, vol. 14, no. 4, pp. 368– 379, 2020, doi: 10.1049/iet-ifs.2019.0447.

[9.] A. Y. Khan, R. Latif, S. Latif, S. Tahir, G. Batool, and T. Saba, “Malicious Insider Attack Detection in IoTs Using Data Analytics,” *IEEE Access*, vol. 8, pp. 11743–11753, 2020, doi: 10.1109/ACCESS.2019.2959047.