IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

Enhanced Cybersecurity for Network Intrusion Detection System Based Artificial Intelligence (AI) Techniques

Mani Gopalsamy
Senior Cyber Security Specialist, Louisville, KY, USA
manigopalsamy14@gmail.com

Abstract: *Cybersecurity has risen to the pinnacle of technological concern due to the exponential growth in the variety and sophistication of cyberthreats. Network attacks are currently the most urgent problem facing contemporary civilization. To identify and stop hostile assaults inside networks, there has to be an intrusion detection system in place. In several industries, most notably information security, effective detection systems for intrusions are being developed using machine learning and deep learning. This is an investigation of how ML methods may be used to enhance cybersecurity defences, with a focus on network intrusion detection, prevention, and response. This study inspects the efficacy of machine learning, utilising CNN, ANN, and LSTM, and assesses them using F1-score, accuracy, recall, and precision. Outcomes demonstrate that CNN outperforms other models, achieving 99% in all key metrics, making them highly effective for detecting network intrusion. However, the study's reliance on the NSL-KDD dataset presents limitations, as it may not fully capture modern network intrusion. Future research should incorporate more recent datasets, real-time performance evaluations, and hybrid models to improve network intrusion detection accuracy and efficiency.*

Keywords: Cybersecurity, network intrusion classification, detection, NSL-KDD dataset, machine learning, CNN

I. INTRODUCTION

The growth of networking technologies and cyber dangers has propelled cybersecurity to the top of the priority list[1]. Finding and stopping bad actors from getting into computer networks is an important part of cybersecurity. The cybersecurity of contemporary network systems is seriously threatened by attacks from malicious third parties[2]. Real individuals or malicious software may both infiltrate a company's resources and cause problems. Malicious actors may access resources via a variety of means, including illicit logins or the acquisition of access credentials; software invaders can manifest as viruses, worms, or ransomware[3]. A plethora of other forms of assault are also discernible. Companies and governments may be severely impacted by undetected intrusions. Threatening national security, causing financial losses and data breaches, and damaging firms' reputations are all consequences of malevolent infiltration. These outcomes provide a formidable obstacle for society[4][5].

The increasing complexity of network intrusion threats has led to the widespread use of NIDS. To protect business networks from cyberattacks, developers created NIDSs. A large number of low-quality warnings are only one of the many problems with NIDS networks[6].

In recent years, AI has gained traction in the cybersecurity industry, and experts in this field are working tirelessly to develop cutting-edge cybersecurity measures. Nevertheless, several detection methods based on AI are insufficient for learning massive amounts of network traffic data, which is becoming more problematic as the network grows in size and complexity[7]. Therefore, due to their improved performance with complex and large-scale network traffic as well as their capacity to learn feature representations from raw data, making them flexible enough to handle a variety of attack scenarios, machine-deep learning-based detection techniques have drawn increased attention in recent years[8].Machine learning may be very helpful in the area of cyber security by drawing insightful conclusions from data.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

A. Aim and contribution

This study is noteworthy for its thorough assessment of several ML and DL models for NIDS using the NSL-KDD dataset. The primary inputs are:

- To enhance NID performance via the use of DL and ML methods.
- Data pretreatment methods to improve model performance, such as min-max normalisation, One-Hot Encoding, controlling null values, and eliminating duplicates.
- Use of feature importance techniques to identify the most relevant attributes for NID.
- Several ML and DL models (CNN, LSTM, and ANN) for NID are tested with the NSL-KDD dataset.
- The comprehensive assessment of classification models using a variety of performance metrics, such as precision, recall, accuracy, and F1-score.
- It offers a well solution for current intrusion detection in cybersecurity due to its substantial benefits in accuracy, efficiency, and scalability.

B. Structure of the paper

The arrangement of the study is as follows: Relevant research on NID is included in Section II. The methods and supplies utilised are described in Section III. This section contains a thorough report on the tests conducted using the recommended system. Section V concludes the investigation and provides an overview of its findings.

II. LITERATURE REVIEW

This section delves into numerous academic publications that put forth a NIDS using different DL and ML techniques. Numerous research studies have focused on intrusion detection systems. The literature review's summary and important phases are shown in Table I.

Lee, Pak and Lee, (2020) provide a technique for classifying DL using extracted features, not for classification purposes but as a technique for preprocessing feature extraction. Features are extracted from a standard unsupervised DL autoencoder model using the RF classification technique by means of a deep sparse autoencoder. Results show that both the classification accuracy and the detection speed have improved. Using the most recent data and comparing with other algorithms like Pearson-RF, SA-RF, and DSA-SVC, it is possible to reach a 99% accuracy rate when classifying normal and attack traffic. But because the sparse class's performance is lower than the other classes, further study is needed to make it better [9].

In propose Dong, Wang and He, (2019) utilises DL, NLP, and big data technologies to identify network intrusions in real-time. Here are the key points of our contributions: (1) Implement huge log collecting in real-time using Flume as the agent and Flink as the real-time computing engine. (2) The traffic data presents a high-dimensional difficulty, therefore one solution is to preprocess the intrusion detection data by cleaning, coding, extracting, integrating, and normalising it. Then, a self-encoder-based intrusion detection dimension reduction approach is suggested. (3) Make a case for AE-AlexNet, an intrusion detection model that employs deep learning and the Auto-Encoder AlexNet neural network. Results from experiments using the KDD 99 intrusion detection dataset demonstrate that the AE-AlexNet model achieves an accuracy of up to 94.32%[10].

This study by Atefi, Hashim, and Kassim (2019) intends to apply anomaly analysis for intrusion detection system categorisation using an extremely current dataset, CICIDS-2017, which may be used for intrusion monitoring assessment.

. This study used the DL approach to perform anomaly analysis for classification purposes utilising KNN for ML and DNN for DL.The findings show that ML and DL perform well in classification when using MCC. With a score of 0.9293%, DNN is clearly the superior classifier when compared to KNN's 0.8824%. An improved security response for networked systems is possible due to this study, which serves as a benchmark for IDS development[11].

In Hakim, Fatma and Novriandi, (2019), see how the IDS reacts when feature selection is used. The impact would be shown by examining. The J48, RF, NB, and KNN algorithms employ. The chi-square, Information Gain, Gain Ratio, and Relief selection methods. Even while it reduces accuracy somewhat, the findings demonstrate that feature selection may greatly improve IDS performance[12].

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

Gain Ratio, Chi-squared, Information Gain, additionally Singh and Mathai (2019) compared the performance of the DBN and SPELM systems for intrusion detection systems. The researcher examined two ML classifiersRelief selection strategies in the J48, RF, NB, and KNN algorithms by means of the NSL KDD dataset: one, the proposed SPELM method, and the other, the DBN algorithm. The researcher conducted an experiment that compared the suggested SPELM algorithm to the current DBN method in terms of computational time, precision, accuracy and recall. The results demonstrate that SPELM performs better than DBN. Specifically, SPELM's accuracy is 93.20 percent, while DBN's is 52.8 percent; SPELM's precision is 69.492 percent, whereas DBN's is 66.83 percent; and SPELM's computational time is 90.8 seconds, compared to 102 seconds for DBN [13].

InRezaeipanah, Afsoon and Ahmadi, (2020) with the goal of enhancing computer network security, a technique is introduced that merges DL with observer learning to identify patterns of infiltration. An method for deep neural networks that makes use of linear combinations and representations of useful characteristics may have its parameters taught by an observer. Experiments conducted on the NSL-KDD dataset demonstrate that the suggested approach outperforms MARS and DLNN by 97.64% [14].

Table 1: Summary of Literature Review for Detecting Network Intrusion Using Machine Learning

References	Methods	Dataset	Results	Limitation/Future Work
Lee, Pak, And Lee[9]	Deep Sparse Autoencoder for feature extraction, classified using Random Forest (RF)	Latest data	99% precision for both assault and regular traffic	Sparse class performance is lower than other classes; additional research needed to improve it.
Dong, Wang, And He [10]	AE-AlexNet neural network for real-time network intrusion detection, including Flume for log collecting	KDD 99	94.32% accuracy	Efficient for high-dimensional data, but further exploration needed for real-time intrusion detection in complex networks.
Atefi, Hashim, And Kassim[11]	Anomaly analysis using KNN and DNN for classification in IDS	CICIDS-2017	DNN: 0.9293% MCC, KNN: 0.8824% MCC	DNN significantly outperforms KNN. Contribution to improving anomaly-based intrusion detection systems.
Hakim, Fatma, And Novriandi[12]	Feature selection impact on IDS using Information Gain, Gain Ratio, Chi-squared, and ReliefF	Public data	Slight reduction in accuracy, improved model performance	Contribution to demonstrating the value of feature selection methods in enhancing intrusion detection performance.
Singh And Mathai [13]	An analysis of the methods used by DBN and SPELM	NSL-KDD	SPELM: 93.20% accuracy, DBN: 52.8% accuracy	SPELM shows better precision, accuracy and computational time compared to DBN. Significant improvement in computational speed.
Rezaeipanah, Afsoon, And Ahmadi [14]	Combination of deep learning and observer learning to detect intrusion patterns	NSL-KDD	97.64% accuracy	Observer learning improves the performance of DNN. Contribution to combining DL and observer learning for enhanced IDS accuracy.

III. METHODOLOGY

The research methodology for Detecting network intrusion for Cybersecurity entails various steps and phases. In the first step of the research process is NSL-KDD data collection, then data is preprocessed to eliminate duplicate entries and manage null values by either eliminating incomplete rows or impute missing data. The next step is to utilise One-Hot Encoding to assign numerical values to the category labels in order to enable the models to cooperate. After that,

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

min-max normalisation is used to scale the numerical features between 0 and 1, ensuring that the ranges of values are consistent. Following this, the relevance of each characteristic to the prediction of the result is ranked using feature importance approaches. Then the data is divided into two subsets: the training subset receives 80% of the data, while the testing subset receives the remaining 20%. Lastly, a confusion matrix is used for training and assessing deep learning and CNN, LSTM, and ANN are examples of machine learning models. These metrics include F1 Score, recall, accuracy, and precision. All processes of research design for software attack are displayed in Figure 1 data flow diagram.

A. Data collection

The 1999 KDD Cup dataset was improved upon to create the NSL-KDD dataset, which increases model clarity and decreases redundancy for intrusion detection and classification. Research on network security benefits from the examples and reduced attributes of the four categories. The dataset consists of KDD Train+ and KDDTest+ sets, with evaluation-specific subcategories such as KDDTrain+_20Percent and KDDTest-21. The 40 labels on the dataset's "attack" label classify assaults as revised, U2R, DoS, R2L, and probing. There are subclasses with attack kinds for every primary class. DoS interferes with network traffic, R2L obtains local access through distant systems, U2R increases user rights, and Probe gathers data. There will be 40 subclasses in all, with 39 attack kinds in addition to the "normal" class.

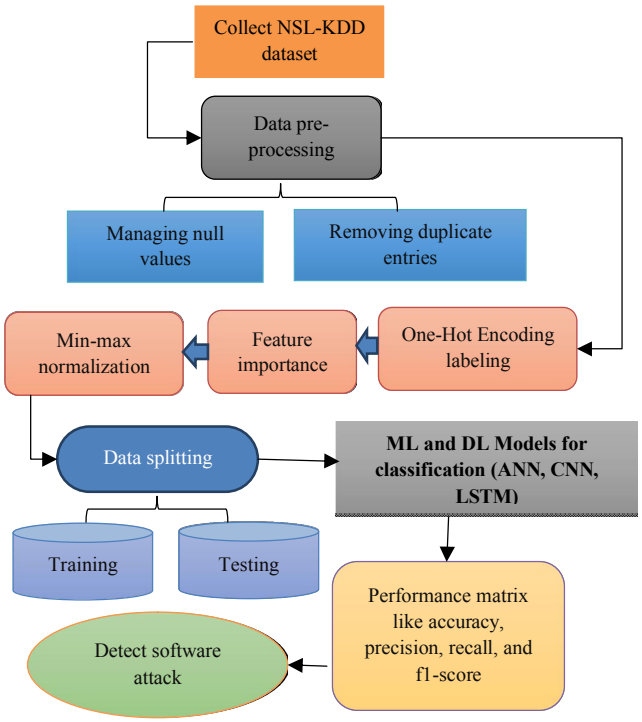

Figure 1: Flowchart for Software attack detection

The following Figure 1 steps are discussed below:

B. Data preprocessing

Because raw data frequently tends to be inconsistent and noisy and may contain missing, redundant, and irrelevant data, pre-processing of data is an important part of the process in ML approaches. For a model to be constructed with excellent performance and accuracy, accurate pre-processing is therefore necessary. The subsequent stages provide a summary of the pre-processing of NSL-KDD data used in this investigation.

- **Removing duplicate entries:** By identifying and eliminating repeated records, the integrity and quality of the data improve, leading to more reliable outcomes.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

- **Managing null values:** The consistency of the dataset is preserved by employing techniques like deleting rows with null values or imputing missing values using the mean, median, or mode.

C. One-hot encoding for labeling

Another method for turning categorical data into a numerical format is called one-hot encoding. Rather than assigning a single number to each category, it generates a new binary column with the values 1 or 0.

D. Min-max normalization

The NSL-KDD dataset's numerical column values were normalised using the min-max technique to a standard scale between 0 and 1, without altering the value ranges. Applying Equation (1)[15].

$$Y = a - \frac{\min(a)}{\max(a)} - \min(a) \qquad (1)$$

where Y represents the normalised value and a represents the original value.

E. Feature importance

Feature significance refers to methods that, for a given model, assign a number to each input characteristic. The ratings show the "importance" of each trait. A higher score denotes a feature's larger influence on the model's ability to predict that specific variable.

F. Data splitting

The training set and the testing set were created by dividing the data into two sections. Eighty percent of the data are for training, while twenty percent are for testing.

G. Classification models

This study classified network intrusion threats using ML and DL models (CNN, LSTM, and ANN).

1. Convolutional neural network (CNN)

An instance of a feedforward neural network is the CNN, which has five layers: an input layer, a convolutional layer, a layer for pooling, a full connection layer, and an output layer. [16]. Every so often, the convolutional and pooling layers switch places [17]. There are many different kinds of structures, and each one has a distinct CNN activation function.The CNN convolutional layer has one or more feature planes. In a feature plane, each neuron has a unique pattern; Weights are also shared by neurones on the same feature plane.The convolution kernel is linked to the shared weights; nevertheless, suitable weights are obtained through model training in order to maximise the network's parameters. The CNN network gathers and combines local features to obtain global information while also using fewer neurone nodes. Setting the weight of each neuron equally can significantly minimize the amount of network parameters, which is especially useful given the high number of neurons at this time. The output of the first k convolution kernels is ykm, while the output of the first m convolution layer is ym. Equation (2) is utilised in this:

$$y_k^m = \delta(\sum_{y_i^{n-1} \in Mk} y_i^{m-1} * w_{ik}^m + b_k^m) \qquad (2)$$

where δ $(\cdot)$ is the function of activation, Wikm is the convolution kernel, $*$ is the convolution, and Mk is a layer of characteristic collection. Bias or offset is bkm. In instruction to decrease the dimension of the input data and quicken the convergence of the network training, the pooling layer comes after the convolutional layer. The second is to keep the network from being overfit and remove superfluous features.The neurons of this present layer are directly connected to each neuron in the complete connection layer below it. The overall features may comprise a fusion of all the local characteristics identified in the previous layer throughout the complete connection layer. By a process which is called the complete connection layer, the activation functions which are used by each neuron are transmitted to the output layer.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

2. Long Short-Term Memory (LSTM)

LSTM networks were created expressly to solve the standard RNN's disappearing gradient issue. This problematic hinders the RNN's ability to efficiently capture temporal relationships by limiting its capacity to retain and convey pertinent information across lengthy periods[18]. The components that make up an LSTM cell are as follows:

- **Cell State (C_t):** The cell state, which is proposed as the reminiscence component, helps the LSTM to remember important data over time. Through specialised gates, it controls which data is saved and which is deleted, enabling the network to deduce from the data long-term dependencies.
- **Input Gate (i_t):** The input gate regulates how fresh data enters the cell state. It determines which values to add to the current input and the cell state from the previous hidden state.
- **Forget Gate (f_t):** The forget gate determines which cell state data ought to be erased. It enables the LSTM to deliberately remove information from earlier time steps that is no longer relevant or current.
- **Output Gate (o_t):** The output gate regulates the filtering process used to extract the current hidden state from the data from the cell state. It controls the data that is sent to the following time step.

3. ANN

The notion of the biological neural network serves as the foundation for ANNs. To offer the appropriate answer to a problem, an ANN is made up of many densely linked neurones organised into three layers: the IL, the HL, and the OL. Weighted connections hold these layers together. To improve its performance on a given task, the network can alter the weights assigned to each link between nodes. The number of HL and neurones in an ANN can affect how well it performs. Having a large number of neurones in the HL or HLs may assure accurate learning even if it may also increase network complexity [19].

IV. RESULT ANALYSIS AND DISCUSSION

This part provides the results of the best model for network intrusion detection for enhance cyber security based on AI models. This section is broken into various subsections. Firstly, provide the NSL-KDD data analysis with EDA. Then utilised performance matrix for evaluation. And lastly, provide the comparative analysis between ML and DL models.

A. Data analysis and visualisation

Data visualisation is the process of representing information and data through graphics. Data visualisation tools employ visual elements like as charts, graphs, and maps to facilitate the observation and understanding of trends and patterns in data. Below are the NSL-KDD data visualisation images:

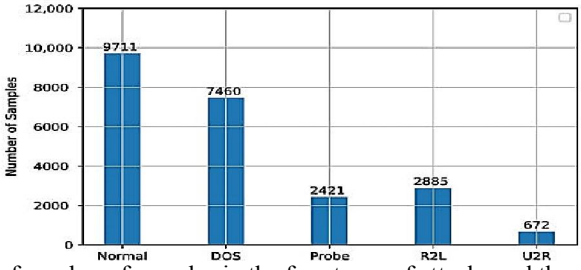

Figure 2: Count Plot of number of samples in the four types of attacks and the normal samples

Figure 2 presents a count plot illustrating the distribution of attack types and normal samples within the test dataset. The x-axis enumerates the attack types (Normal, DOS, Probe, R2L, U2R), whereas the matching number of samples is shown on the y-axis. From the visualization, it's evident that the dataset is imbalanced. The "Normal" class significantly outnumbers all other categories, indicating a predominance of benign traffic. The "DOS" attacks follow in frequency, suggesting a prevalence of denial-of-service attempts. The remaining attack types, "Probe," "R2L," and "U2R," exhibit considerably fewer instances.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

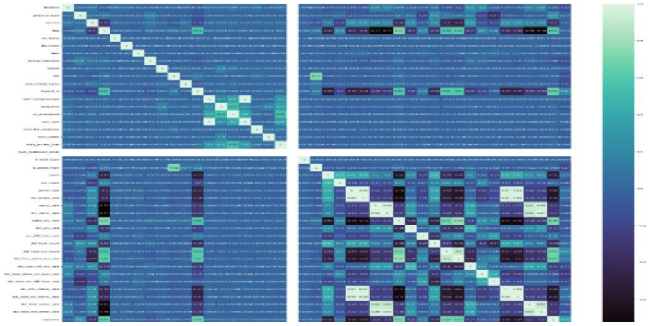

Figure 3: Heatmap for NSL-KDD Dataset

A heatmap depicting the association between the 43 significant NSL-KDD dataset features is illustrated in Figure 3. Bright colours in this heat map represent positive association, whereas dark colours suggest negative correlation.

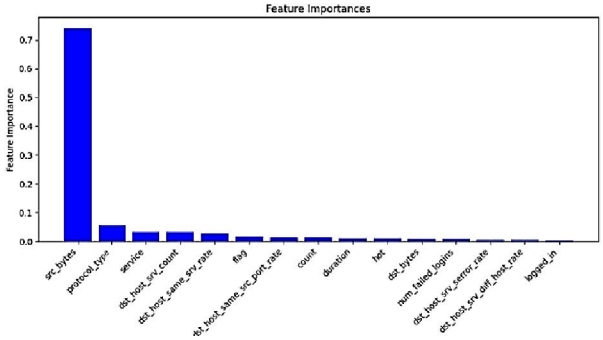

Figure 4: Feature Importance Bar Graph using Dataset

A bar graph of the dataset's feature relevance is displayed in Figure 4. Features like srcbytes, protocol type, and service are listed on the x-axis, while their relevance ratings are displayed on the y-axis. With a score greater than 0.7, the "scribers" characteristic has the highest relevance.

B. Performance metrics

In this paper, the models developed using classification algorithms were evaluated using a confusion matrix. Four statistical measures were employed for performance evaluation: accuracy, precision, and F. Sensitivity represents the probability that the TP class—where "Y" denotes "Yes"—will be correctly identified, and the TN class—where "Y" denotes "No"—will be correctly identified—is represented by specificity. The terms FN and FP denote scenarios in which the model predicts a negative class when the real class is positive, and negative class and positive class, respectively. Following performance measures are as follows:

1. Accuracy

Vital variables accuracy Outcome the classifiers' accuracy is a vital consideration while bringing data estimations to the table. Any prediction model's accuracy may be expressed as (3):

$$Accuracy = \frac{TP+TN}{TP+Fp+TN+FN} \quad (3)$$

2. Precision

It is the percentage of documents correctly categorised as belonging to the positive prediction class relative to all documents in that class. It is written as (4).

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

3. Recall

The computation involves dividing the entire number of pertinent samples by the quantity of precise positive results. It is expressed mathematically as (5)

$$Recall = \frac{\text{TP}}{TP+FN} \quad (5)$$

4. F1-score

The F-score is the precision and recall weighted average. The F-Score considers both false positives and false negatives. It is expressed mathematically as (6)

$$F1 - Score = \frac{2(Precision*Recall)}{\text{Precision+Recall}} \quad (6)$$

These matrices are utilized to determine the machine and deep learning models.

5. Experiment results

This section contains the CNN model experiment findings that are used for network intrusion detection in cybersecurity. Table 2 demonstrates that the CNN model achieves 99% network intrusion elimination performance.

CNN model Performance on NSL-KDD dataset

Measures	CNN
Accuracy	99.9
Precision	99
Recall	99
F1-score	99

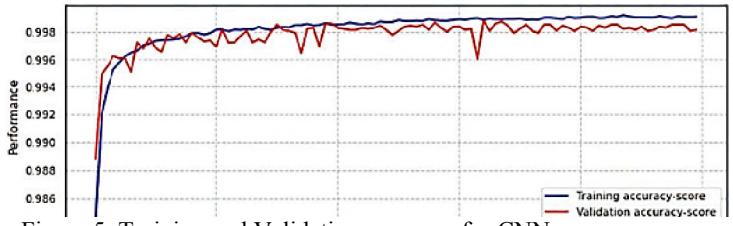

Figure 5: Training and Validation accuracy for CNN

The accuracy of CNN models during training and validation throughout 100 epochs is displayed in Figure 5. With a range of around 0.984 to 0.998, the training (blue line) and validation (red line) accuracies are both extremely high. This suggests that the model has high generalisation without appreciable overfitting, as it performs well on both the training and validation datasets.

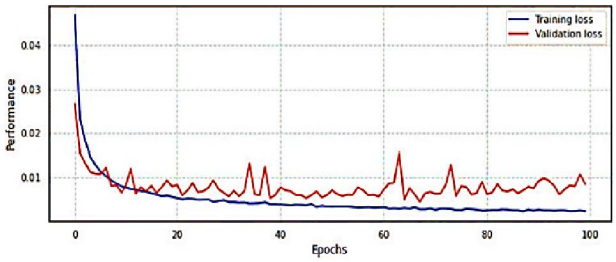

Figure 6: Training and Validation Loss for CNN Model

The validation and training losses for CNN models across 100 epochs are displayed in the accompanying Figure 6. The validation loss reduces at a slower pace than the training loss, which is shown by the red line. The blue line indicates the training deficit. This indicates that the model is learning and improving its performance over time.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

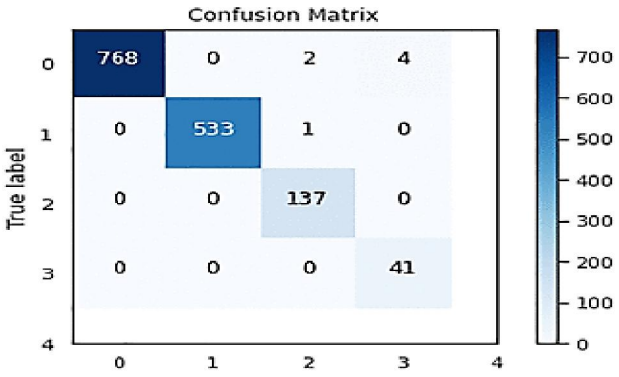

Figure 7: Confusion matrix for CNN Model

Figure 7. illustrates the confusion matrix, which is created on the NSL-KDD dataset and compares the CNN Intrusion Detection System's genuine labels against its anticipated labels. The confusion matrix reveals strong performance across four classes, with near-perfect accuracy for classes 0 and 1, where the model correctly predicted 768 out of 770 instances and 533 out of 534 instances, respectively. Classes 2 and 3 exhibit perfect accuracy, as all 137 and 41 instances were correctly classified. However, class 4 shows a slight decline in performance, with the model correctly predicting only 7 out of 10 instances, indicating some misclassifications and room for improvement in this class. Overall, the model performs exceptionally well in most categories, with minor inaccuracies in class 4.

D. Comparative Analysis

The comparative analysis for detecting network intrusion attacks for cybersecurity on the NSL-KDD dataset is provided in this unit. The comparison of Machine Learning and Deep Learning models based on performance matrices like accuracy, recall, f1-score and precision.

Table 3: ML and DL models comparison on the NSL-KDD dataset for Cybersecurity

Models	Accuracy	Precision	Recall	F1-score
LSTM[20]	84.25	84.20	83	84.20
ANN[21]	78	96	62.05	75.57
CNN	99	99	99	99

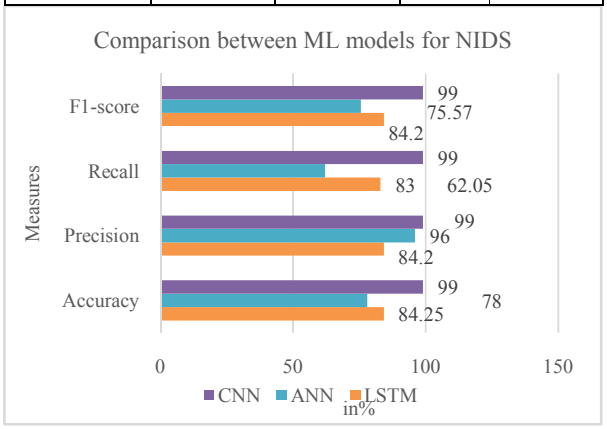

Figure 8: Bar graph of comparison ML models for NIDS

The performance of three ML models—LSTM, ANN, and CNN—is compared in Table 3 and Figure 8 using a variety of assessment criteria, including recall, accuracy, precision and F1-score. The CNN model significantly outperforms the others, achieving 99% across all metrics, indicating exceptional classification ability with minimal errors. The LSTM model follows with an accuracy of 84.25%, showing Implementing an intrusion detection and prevention system using software-defined networking: It "balanced performance in precision 84.20%, recall 83%, and F1-score 84.20%, making

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

it effective but not as precise as CNN" in thwarting denial-of-service and port-scanning assaults. Meanwhile, the ANN model exhibits moderate accuracy of 78% and a high precision 96%, but its recall is relatively low at 62.05%, resulting in a lower F1-score of 75.57% compared to the other models. This suggests that CNN is the most robust model for the task.

V. CONCLUSION AND FUTURE WORK

The threat that network intrusion poses to computer systems throughout the globe is well-known, and security professionals are always on the lookout for and swiftly classifying and detecting network intrusion. Static analysis and security systems can be circumvented by network incursion using evasion strategies; hence, dynamic analysis approaches are more effective in precisely analysing malware behavioural patterns. Precise categorisation of network intrusions is advantageous for the creation of intrusion signatures, which benefits antivirus software providers. Organisational security specialists may also benefit from this capacity, since it empowers them to counteract intrusion attempts and address security events.The NSL-KDD dataset was utilised in this investigation. to thoroughly assess how well different ML and DL models performed in detecting network intrusion. The models were contrasted using important measures such as F1-score, recall, accuracy, and precision. Among the models, CNN demonstrated the highest performance, with 99% accuracy and precision, making them the most reliable for network intrusion detection in cybersecurity. But other models, such as LSTM and ANN, also offered insightful information on their relative advantages and disadvantages in identifying various attack types. Finally, further research on explain ability in network intrusion detection models could improve interpretability and trust in cybersecurity systems. Further will used hybrid models and other intrusion datasets.

REFERENCES

[1] C. Birkinshaw, E. Rouka, and V. G. Vassilakis, “Implementing an intrusion detection and prevention system using software-defined networking: Defending against port-scanning and denial-of-service attacks,” *J. Netw. Comput. Appl.*, 2019, doi: 10.1016/j.jnca.2019.03.005.

[2] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, “Survey of intrusion detection systems: techniques, datasets and challenges,” *Cybersecurity*, 2019, doi: 10.1186/s42400-019-0038-7.

[3] V. K. Yarlagadda and R. Pydipalli, “Secure Programming with SAS: Mitigating Risks and Protecting Data Integrity,” *Eng. Int.*, vol. 6, no. 2, pp. 211–222, Dec. 2018, doi: 10.18034/ei.v6i2.709.

[4] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, “Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study,” *J. Inf. Secur. Appl.*, 2020, doi: 10.1016/j.jisa.2019.102419.

[5] V. V. Kumar, A. Sahoo, and F. W. Liou, “Cyber-enabled product lifecycle management: A multi-agent framework,” in *Procedia Manufacturing*, 2019. doi: 10.1016/j.promfg.2020.01.247.

[6] A. A. Reyes, F. D. Vaca, G. A. C. Aguayo, Q. Niyaz, and V. Devabhaktuni, “A machine learning based two-stage wi-fi network intrusion detection system,” *Electron.*, 2020, doi: 10.3390/electronics9101689.

[7] H. Liu and B. Lang, “Machine learning and deep learning methods for intrusion detection systems: A survey,” *Applied Sciences (Switzerland)*. 2019. doi: 10.3390/app9204396.

[8] Q. Meng, Y. Yang, F. Wu, X. Chen, and X. Chen, “Research on Network APT Attack Intrusion Detection Technology Based on Machine Learning Algorithm,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 799, no. 1, p. 012029, Mar. 2020, doi: 10.1088/1757-899X/799/1/012029.

[9] J. Lee, J. G. Pak, and M. Lee, “Network Intrusion Detection System using Feature Extraction based on Deep Sparse Autoencoder,” in *International Conference on ICT Convergence*, 2020. doi: 10.1109/ICTC49870.2020.9289253.

[10] Y. Dong, R. Wang, and J. He, “Real-time network intrusion detection system based on deep learning,” in *Proceedings of the IEEE International Conference on Software Engineering and Service Sciences, ICSESS*, 2019. doi: 10.1109/ICSESS47205.2019.9040718.

[11] K. Atefi, H. Hashim, and M. Kassim, “Anomaly analysis for the classification purpose of intrusion detection system with K-nearest neighbors and deep neural network,” in *Proceeding - 2019 IEEE 7th Conference on Systems, Process and Control, ICSPC 2019*, 2019. doi: 10.1109/ICSPC47137.2019.9068081.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 5.731
Volume 12, Issue 1, December 2021

[12] L. Hakim, R. Fatma, and Novriandi, "Influence Analysis of Feature Selection to Network Intrusion Detection System Performance Using NSL-KDD Dataset," in *Proceedings - 2019 International Conference on Computer Science, Information Technology, and Electrical Engineering, ICOMITEE 2019*, 2019. doi: 10.1109/ICOMITEE.2019.8920961.

[13] K. Singh and K. J. Mathai, "Performance Comparison of Intrusion Detection System Between Deep Belief Network (DBN)Algorithm and State Preserving Extreme Learning Machine (SPELM) Algorithm," in *Proceedings of 2019 3rd IEEE International Conference on Electrical, Computer and Communication Technologies, ICECCT 2019*, 2019. doi: 10.1109/ICECCT.2019.8869492.

[14] A. Rezaeipanah, E. Afsoon, and G. Ahmadi, "Improving the Performance of Intrusion Detection Systems Using the Development of Deep Neural Network Parameters," in *2020 10h International Conference on Computer and Knowledge Engineering, ICCKE 2020*, 2020. doi: 10.1109/ICCKE50421.2020.9303701.

[15] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Appl. Soft Comput.*, 2020, doi: 10.1016/j.asoc.2019.105524.

[16] J. Gu *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognit.*, 2018, doi: 10.116/j.patcog.2017.10.013.

[17] C. Wigington, S. Stewart, B. Davis, B. Barrett, B. Price, and S. Cohen, "Data Augmentation for Recognition of Hadwritten Words and Lines Using a CNN-LSTM Network," in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2017. doi: 10.1109/ICDAR.2017.110.

[18] G. Van Houdt, C. Mosquera, and G. Nápoles, "A review on the long short-term memory model," *Artif. Intell. Rev.*, 2020, doi: 10.1007/s10462-020-09838-1.

[19] Y. chen Wu and J. wen Feng, "Development and Application of Artificial Neural Network," *Wirel. Pers. Commun.*, 2018, doi: 10.1007/s11277-017-5224-x.

[20] T. Su, H. Sun, J. Zhu, S. Wang, and Y. Li, "BAT: Deep Learning Methods on Network Intrusion Detection Using NSL-KDD Dataset," *IEEE Access*, 2020, doi: 10.1109/ACCESS.2020.2972627.

[21] S. Sapre, P. Ahmadi, and K. Islam, "A Robust Comparison of the KDDCup99 and NSL-KDD IoT Network Intrusion Detection Datasets Through Various Machine Learning Algorithms," 2019.