Annals of Science and Technology - A, Vol 10 (1): 11-23, 2025
Copyright: An Official *Journal* of the Nigerian Young Academy
ISSN: 2 544 6320

ARTICLE

RBM-CNN: A Combined Deep Learning Model for Intrusion Detection in IoT Networks

Olufunke G. Darley[1*], Adetokunbo A. Adenowo[2] and Abayomi I. O. Yussuff[1]

1 Department of Electronic & Computer Engineering, Lagos State University, Nigeria.
2 Department of Computer Science, Lagos State University, Nigeria.
Received 31st October, 2024, Accepted 12th March 2025
DOI: 10.2478/ast-2025-0002

**Corresponding author*
Olufunke G. Darley (PhD) E-mail: funke_darley@yahoo.com; Tel: +2348023065513

Abstracts
New devices join millions of existing ones in the Internet of Things (IoT) network. Associated threats/attacks/intrusions to network data and IoT devices themselves need to be identified and mitigating actions taken in a timely manner to secure data and protect the network. Network intrusions continually evolve due to the creation of new attacks and this has presented an ever-changing challenge. One of the ways to overcome this is the deployment of Intrusion Detection Systems (IDS). While IDS have been found to be effective in this regard, some studies have shown that there is a drop in IDS performance when datasets larger than the one with which they are trained are encountered. An anomaly-based IDS is proposed to overcome this challenge. The proposed IDS leverages two deep learning (DL) techniques – the Restricted Boltzmann Machine (RBM) and the one-dimensional Convolutional Neural Network (1D-CNN). Proposed model performances were evaluated using the NSL-KDD benchmark dataset (148k+ data points) and the much larger CSE CIC-IDS2018 dataset (11M+ data points). With the unbalanced NSL-KDD dataset, it was observed that the model was better suited for dealing with DOS attacks only. To improve performance, the balanced NSL-KDD dataset was used and it was observed that the model performed better for all metrics. The model was thereafter evaluated using the balanced CSE-CIC-IDS2018 dataset with the model performing very well overall with the exception of accuracy which experienced some reduction (from 0.9585 to 0.8564). While reduced accuracy is not preferred, precision and recall values improved from 0.8451 to 0.8689 and from 0.7423 to 0.8564 respectively. This is crucial in anomaly/intrusion detection with high precision indicating a low rate of false positives and high recall indicating the model is effectively capturing most of the anomalies. Thus, the proposed model will be very useful in IoT applications with its ever-expanding dataset.
Keywords: Keywords: Anomaly Detection, Deep Learning, Internet of Things, Intrusion Detection

1.0 Introduction

The term Internet of Things (IoT) has many definitions (Chacon, 2024; Desbiens, 2023; Rayes and Salam, 2019) but it essentially refers to the collective network of connected devices and the technology that facilitates communication between the devices and the cloud, as well as between the devices themselves. The IoT system is particularly vulnerable at the device layer due to the millions of unsecured IoT devices already connected and with many more expected to join. Statista.com estimated that the sum of IoT-connected devices will be 30.9 billion units worldwide by 2025 (Businesswire, 2021). These devices are heterogeneous having different manufacturers, protocols, specifications, and command interfaces with no existing standard followed; thus, opening up the entire IoT system to threats and attacks. While existing threats are easier to manage, emerging threats consist of different combinations of known threats and are therefore dynamic and more challenging. These threats and attacks leverage the huge data generated by the devices, the inadequate computing resources, and the limited storage capacity of IoT devices. One of the ways to overcome these threats/attacks/intrusions in order to ensure system security and privacy of data is the development and deployment of Intrusion Detection Systems (IDS). IDS are expected to detect existing threats as well as flag new trends of attacks and initiate mitigating actions to ensure system protection. An IDS placed within a network scans network traffic to identify and report a violation based on the preconfigured customized detection levels (Ashiku and Dagli, 2021). Also, IDS serve as additional network security to firewalls and anti-viruses. (Figure 1).

Anomalies are unknown threats (also called zero-day attacks) and are made up of those portions of data that do not conform to expected behaviour or deviate from expected trend/characteristic. Every device or system involved in the data cycle of generation, transportation, processing, dissemination and storage has behaviour that is considered normal or characteristic. Anomaly-based detection scans network traffic to identify patterns that deviate from normal or baseline behavior (Injadat *et a*l, 2018; Ukil et al, 2016; Chandola *et al*, 2009). Changes to normal behaviour are considered anomalous and will be flagged accordingly. New attacks/anomalies/intrusions leverage on the dynamic nature of IoT, with new attacks continuously generated based on combinations of known attacks. It is important that intrusions are detected early so that they can be ejected from the system before any damage is done to the data. IDS are designed based on the assumption that behavioral features of intrusions are different from legitimate users' behavior; therefore, IDS quantifies intrusion behavior in terms of its features. In practice, however, it is challenging to make out an exact distinction between normal and abnormal behavior. This sometimes leads to the categorization of normal behaviour as abnormal behaviour or intrusion thereby resulting in false positive identification (Wazid *et al*, 2021).

In IoT applications, the essence of IDS is to detect, react and report attacks or malicious activities that can have the potential to cripple the IoT network (Koroniotis *et al*, 2020). Machine learning (ML) has been very useful in this regard with the ability of its techniques to identify patterns in the massive data generated in IoT systems (Kubat, 2021; Shone *et a*l, 2018), thereby useful in the implementation of IDS. ML, however, has a limitation with the amount of data it can handle. To overcome the challenge of dealing with massive and ever-increasing IoT data, a subgroup of ML known as Deep Learning (DL) has been applied in IDS development. DL models are developed using artificial neural networks (ANN) to perform complex computations on huge amounts of data. It consists of artificial neurons or nodes organized in three layers – the input layer, the hidden layer(s) and the output layer (Rayes and Salam, 2019) with these neurons completely connected from one layer to another. The choice of the number of neurons for a hidden layer is important as it affects the performance of the neural network. Too few neurons means the network does not have enough resources to learn the general features of the training data while too many neurons increases the training time, and this does not necessarily yield a corresponding increase of performance. DL strives to learn more useful features by constructing multiple hidden layers, in order to obtain higher accuracy (Lansky *et al*, 2021; Praveena and Vivekanandan, 2021; Aleesa *et al*, 2020; Liu and Lang, 2019; Feng *et a*l, 2019). While both ML and DL techniques have been used in IDS, the latter has exhibited advantage over the former in terms of performance alongside the massive data generated by IoT devices. In other words, IDS have been shown to perform better when DL techniques are applied since they have "a high capacity for self-learning, self-adaptation, generalization and identification of unknown attack activity" (Manimurugan and Al-Mutairi, 2020). The superior performance of DL techniques has led to its emergence as a new approach to improve intrusion detection. However, DL

techniques have longer running times and more computational complexity; thus requiring more resources.

From existing studies, one of the limitations of IDS is that a drop in its performance is experienced when a dataset that is much larger than the one with which it is trained with, is involved (Hu *et al*, 2023; Mezina *et al*, 2021; Kimbugwe *et al*, 2021; Hassija *et al*, 2019; Vinayakumar *et al*., 2019). This research will therefore explore the advantages DL techniques offer in order to develop an anomaly-based IDS that will maintain model performance (e.g. accuracy, precision, recall) when larger datasets are encountered as is the case with IoT dataset. The growing nature of the IoT dataset means that it is exposed to a growing number of attacks. This work thereby proposes a combined DL model consisting of the Restricted Boltzmann Machine (RBM) and the One-Dimensional Convolutional Neural Network (1D-CNN). The proposed model will be evaluated using publicly available IDS datasets, namely the benchmark NSL-KDD[1] and the CSE-CIC-IDS2018[2] datasets.

2.0 Related Work

Several studies have been carried out on utilizing DL models for intrusion detection and summaries of some relevant studies are presented below.

A systematic literature review carried out on using Generative Adversarial Networks (GANs) network for anomaly detection showed that that GANs were able to overcome the challenge of paucity of abnormal data that usually occurs in anomaly detection (Lim *et al*, 2024). Also, selecting mini-batch training as a key optimization strategy reduced computation time GANs suitable for real-time or near-real-time analysis of incoming network traffic. The limitations identified, however, included susceptibility to class imbalance in network data and the high computational resource required to train and use them.

A review which sought to compare several new intrusion detection models with existing ones was presented by (Rafique *et al*, 2024). Various performance and security metrics were used to determine the suggested models' efficiency and accuracy. It was concluded that while the model performances were good, further areas of research should include the use of varied datasets, real-time testing and scalability of systems to serve as a means of improving anomaly detection.

A new network traffic anomaly detection model which combined two DL-based techniques was proposed by (Hu *et al*, 2023). The model combined the Long Short Term Memory (LSTM) and the recurrent neural network (RNN) techniques and used a newly proposed feature extraction method. The UNSW-NB15dataset was used for performance evaluation. Results showed that the model performed better than other classic ML models using the newly-constructed dataset. Limitation was that the model was used for binary classification only. Further work can be carried out by applying it to multi-class classification.

A 1D-CNN-based model was proposed by (Hooshmand and Hosahalli, 2022) with the SMOTE sampling method used to solve the class-imbalance issue in the dataset. The UNSW-NB15dataset was used for performance evaluation. The dataset was split into different protocol categories and each category was treated independently. Results showed that treating independent categories yielded better results in the case of Recall and F1-score; however, that was not the case for Precision. The limitation was the small sample size used. It proposed carrying out more hyper-parameter tuning on the model, using a full-sized benchmarked dataset and employing different sampling techniques as a means of improving model performance.

The multi-layer perceptron was used to develop two IDS models by (Maithem and Al-sultany, 2021) for binary and multi-class classification. The KDD Cup1999 dataset used was categorized into Denial of Service (DoS), Remote to Local (R2L), User to Root (U2R) and Probe attacks. Results showed that high accuracy (99.98 %) was obtained for both binary and multi-class classification with that for DoS attacks achieving 99.99%. The limitation was that other attack types such as infiltration and web attacks were not considered and there was the problem of overfitting. It was proposed that further work be done with a more current dataset with more attack types.

Two detection models: one based on U-Net and the other based on Temporal Convolutional Network (TCN) and LST was proposed by (Mezina *et al*, 2021). U-Net is a CNN that was

[1] Network Security Lincoln Laboratory – Knowledge, Discovery and Data Mining

[2] Canadian Security Establishment – Canadian Institute for Cybersecurity

developed for image segmentation. Two datasets, the KDD99 and CSE-CIC-IDS2018 datasets were used. Results showed that both models outperformed existing models on the CSE-CIC-IDS2018, and the U-Net model achieved better results on the KDD99 with accuracy of 93% and 94% on CSE-CIC-IDS2018. TCN+LSTM had 92% and 97% respectively. Also, the U-Net model was better at classifying attacks, in spite of the fact, that there were small numbers of samples in training and testing sets. It was observed that there was overfitting of the models using the KDD99 dataset, but models trained on CSE-CIC-IDS2018 had almost the same results for training and testing sets. Limitation was that model performance which is successful on the old dataset experienced lower performance accuracy when a modern (larger with more attack types) dataset is used.

Various DL-based IDS models were presented by (Aleesa *et al*, 2020). These included Long Short-Term Memory - Recurrent Neural Network (LSTM-RNN), Artificial Neural Network (ANN) and Deep Neural Network (DNN), for both binary and multi-class classification. The UNSW-NB15 dataset was used for performance evaluation. The performance of the models varied from 97.89% to 99.59% (except for the LSTM-RNN which varied only slightly from 85.38% to 85.42%). The study was limited by hardware capability which restricted the number of hidden layers and neurons that could be used. It proposed using more resources to improve performance.

A deep belief network (DBN) based model was proposed by (Manimurugan and Al-Mutairi, 2020) for anomaly detection. The CICIDS2017 dataset was used for evaluation. Results showed very good performance with accuracy for the different attack types obtained as follows: Normal (99.37%), Botnet (97.93%), Brute Force (97.71%), DoS/DDoS (96.67%), Infiltration (96.37%), PortScan (97.71%), and Web attack (98.37%). Limitation was that a more current dataset (CSE CICIDS2018) with more attack types was not used. For further work, it was proposed that the model be evaluated using more recent datasets.

An intrusion detection model based on two unsupervised DL techniques - RBM and Autoencoder (AE) was presented by (Dawoud *et al*, 2020). The CIC-IDS2017 dataset was used for performance evaluation. The model was based on the assumption that anomalies occur less frequently than normal samples. Limitation was that when the assumption failed (high frequency of DDoS attacks), it suffered a high false alarm rate. Also the more current CSE-CIC-IDS2018 dataset with more attack types was not used. Further work can address these issues.

A survey of studies on intrusion detection was carried out by (Leevy and Khoshgoftaar, 2020) . These were those that specifically used the CSE-CIC-IDS2018 dataset; for performance evaluation; the dataset being the most current with a wide range of attack types. Observations included unusually high performance scores which may be a consequence of over-fitting, effect of class imbalance with its attendant bias and lack of information on data cleaning techniques used which can limit data reproducibility and usability.

An intrusion detection model made up of a combination of CNN and RNN was presented by (Wu *et al*, 2020). The NSL-KDD and UNSW-NB15 datasets were applied for performance evaluation. Results showed that Accuracy was 99.21% and 86.64%, for NSL-KDD and UNSW-NB15 datasets, respectively. The limitation was that there was a reduction in accuracy when a larger dataset was used. The study proposed that more experiments be carried out to improve model performance.

3.0 Experimental

For this study, a combination of two deep learning models – the RBM and the 1D-CNN was used. Two IDS datasets were used for performance evaluation.

3.1 Datasets

In this study, the first dataset that was used to determine the performance of the proposed IDS is the benchmark NSL-KDD with its various attack types characterized into 4 primary categories. Benchmark datasets are created with the aim of providing a unified test bed for evaluating newly developed intrusion detection methods. The NSL-KDD dataset (with 148k+data points) is a clean dataset which aims to offer complete and accurate samples for models by eliminating irrelevant features, null values, duplicates, unwanted information, missing values etc., from the dataset in order to improve the performance of the detection system. The CSE-CIC-IDS2018 datasets is the most current IDS dataset with 16M+ data points and several attack type which can be characterized into 8 primary categories.

3.2 Deep Learning Models

To develop the IDS models, two DL techniques, the RBM and the 1D-CNN are used; network data being one-dimensional. The RBM is a powerful technique that is efficient in computation, can encode any distribution and its output can be used further by other models in order to learn more features and enhance

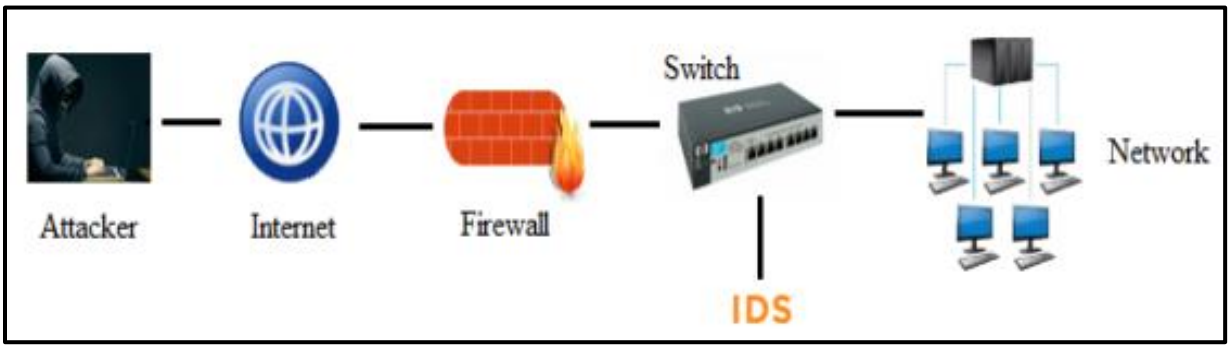

Figure 1: Overview of a network with IDS

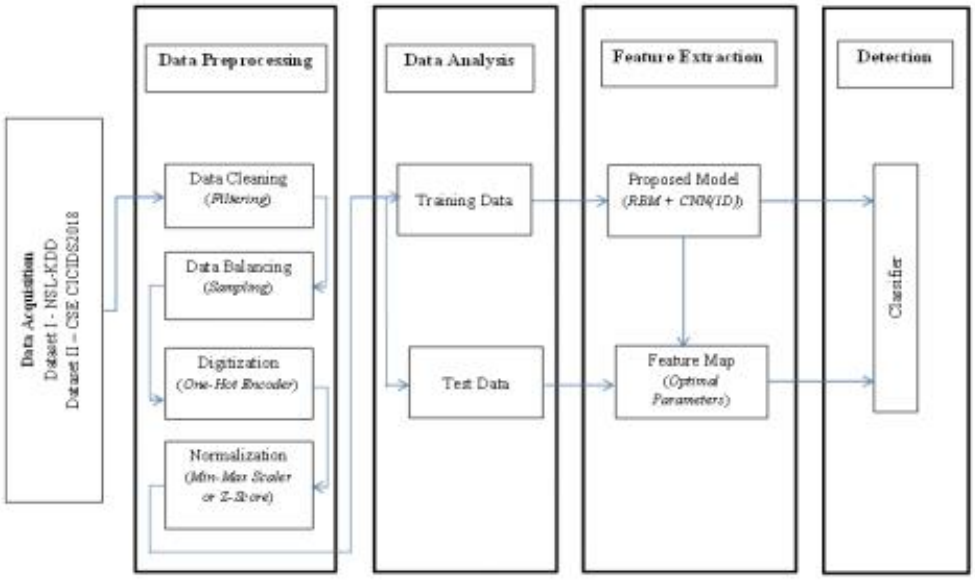

Figure 2: Framework for the proposed IDS

model performance. The 1D-CNN has the advantages of parameter efficiency and data efficiency; convolution being translational equivariant. This means that patterns can be recognized in the data regardless of their position along the input sequence.

3.3 Framework

The proposed framework for the proposed IDS model is presented in Figure 2 below.

Specifically, the following steps were taken:

1. Propose IDS model that is capable of maintaining high detection performance with large datasets.
2. Evaluate proposed IDS model using the selected benchmark IDS dataset (NSL- KDD).
3. Evaluate proposed IDS model using the current IDS dataset (CSE-CIC-IDS2018).
4. Carry out a comparative analysis of the performance results of the IDS using both the NSL-KDD and CSE-CIC-IDS 2018 datasets.

4.0 Results and Discussion

The performance metrics (accuracy, precision, recall and F1-Score) of the proposed IDS (RBM + 1D-CNN) are presented and the results interpreted. Precision and Recall are of particular importance in intrusion detection; the former being an

indication of rate of false positives (a measure of the exactness of the model) and the latter being an indication of the model's ability to effectively capture most of the anomalies (a measure of the completeness of model). The F1-Score is the harmonic average of Precision and Recall and can be useful for unbalanced datasets.

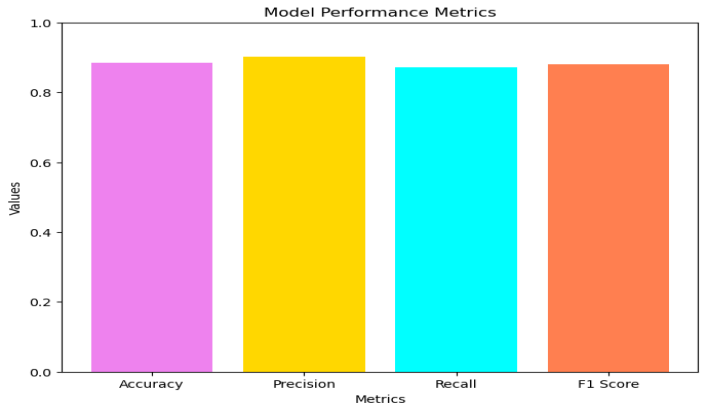

Fig. 3a: Model Performance Metrics for DOS

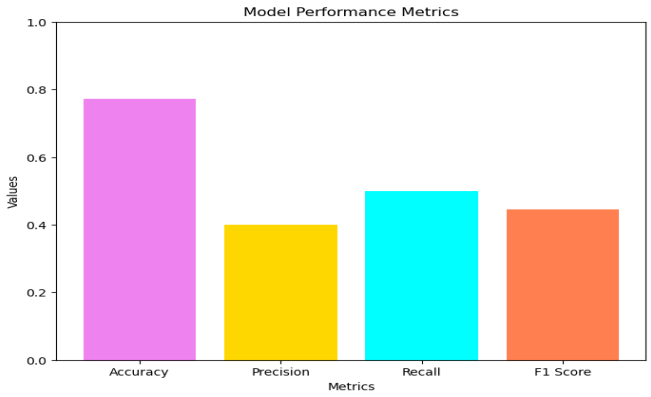

Fig. 3b: Model Performance Metrics for Probe

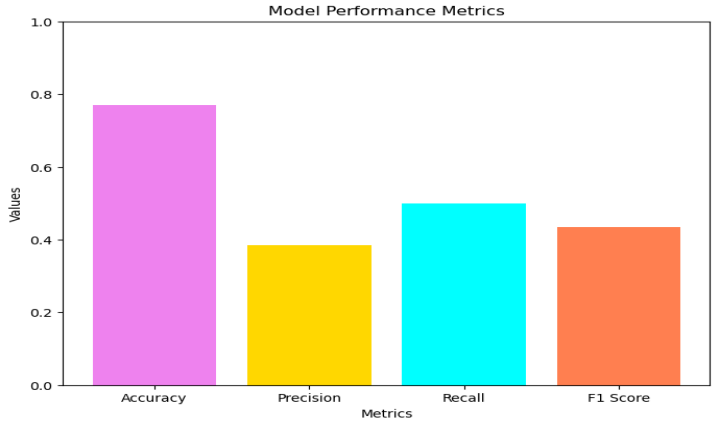

Fig. 3c: Model Performance Metrics for R2L

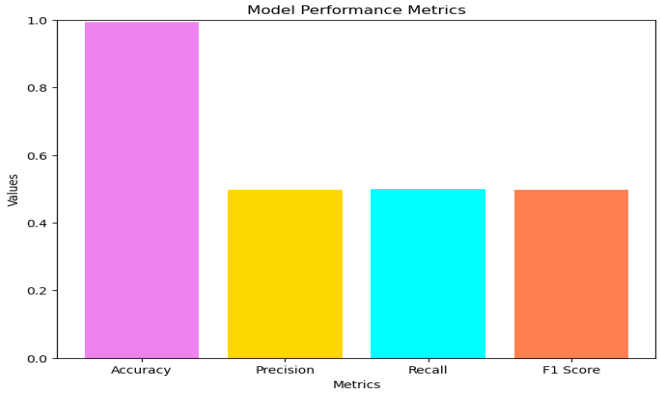

Fig. 3d: Model Performance Metrics for U2R

4.1 Using the Unbalanced NSL-KDD Dataset

When the unbalanced NSL-KDD dataset was used, the performance metrics of the model for each attack category, namely DoS, Probe, R2L and U2R is as shown in Table 1 and Figs. 3a-3d.

From Table 1, it can be seen that for each attack category, accuracy was high with the highest being for U2R at 0.99314. For Precision, DoS attack had the highest value at 0.90211 with other attack categories having rather low values. This pattern was about the same for Recall and FI-Score. Thus, with the exception of DoS, the performance metrics for other attack categories were poor. These are as presented in Figures 3a-3d which show the performance metrics for each attack category of attack versus their values.

From this, the following observations and recommendations are made:

1. The model may be better suited for dealing with DOS attacks only.
2. Further work can be done by running the model using the DDoS Evaluation Dataset (CIC-DDoS2019) which is specifically for DoS attacks.
3. For next steps, data balancing was used to eliminate bias and obtain optimal model performance.

4.2 Using the Balanced NSL-KDD Dataset

When the balanced NSL-KDD dataset was used, model performance metrics (Table 2) and training/validation accuracy and loss plots (Figures 4a & 4b) as obtained; were presented. It can be seen from Table 2 that accuracy was high at 0.9585. Precision and Recall values were also high at 0.84511 and 0.7509, respectively.

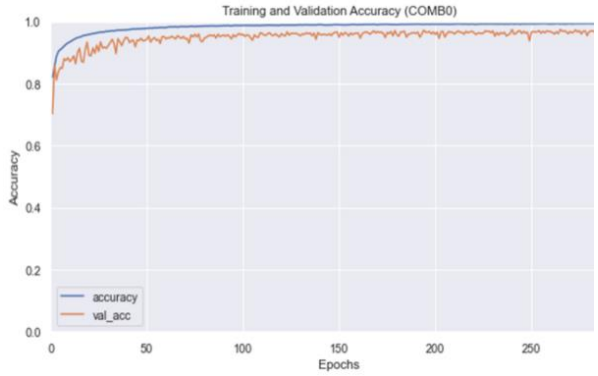

Fig. 4a: Training/Validation Accuracy plots using the NSL-KDD Dataset

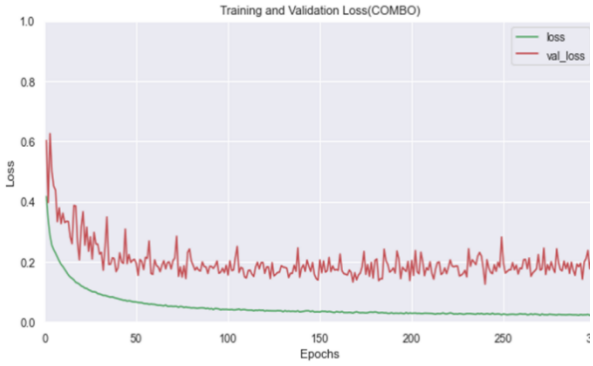

Fig 4b: Training/Validation Loss plots using the NSL-KDD Dataset

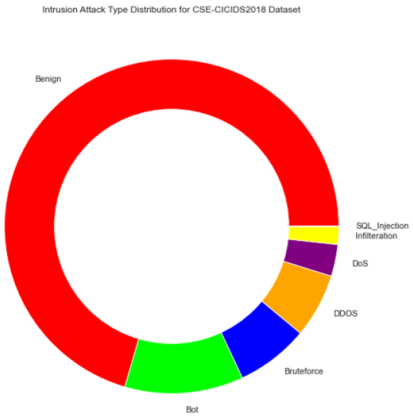

Fig. 5a: Unbalanced Dataset for CSE-CIC-IDS2018 Attack Categories

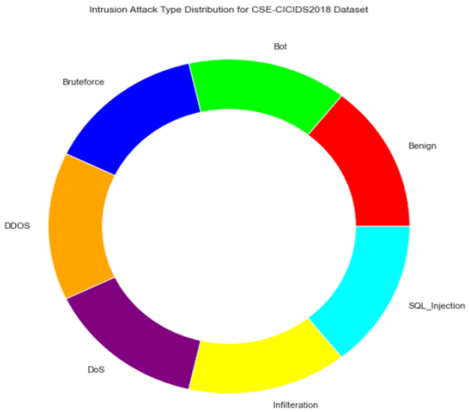

Fig. 5b: Balanced Dataset for CSE-CIC-IDS2018 Attack Categories

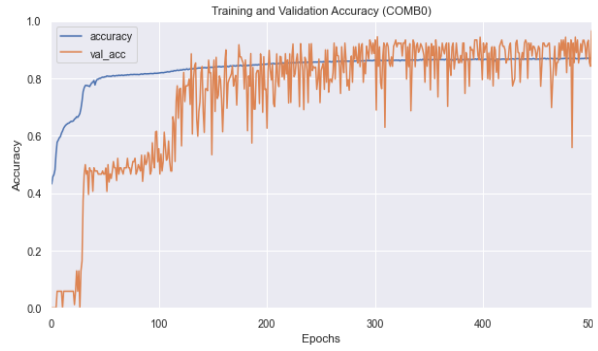

Fig. 6a: Training/Validation Accuracy plot using the CSE-CIC-IDS2018 Dataset

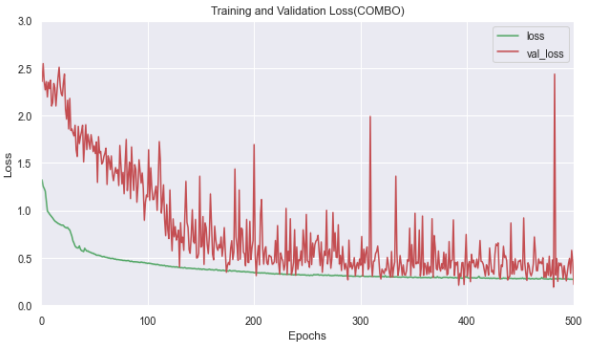

Fig. 6b: Training/Validation Loss plot using the CSE-CIC-IDS2018 Dataset

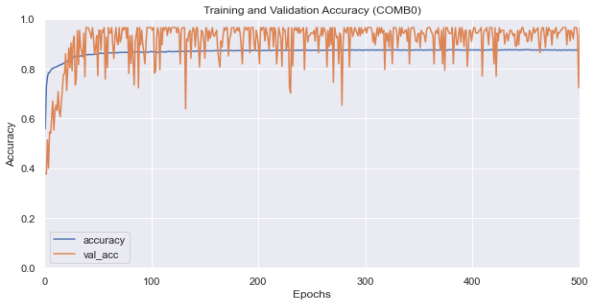

Fig. 7a: Training/Validation Accuracy plots using the CSE-CIC-IDS2018 Dataset (Drop-out (0.2))

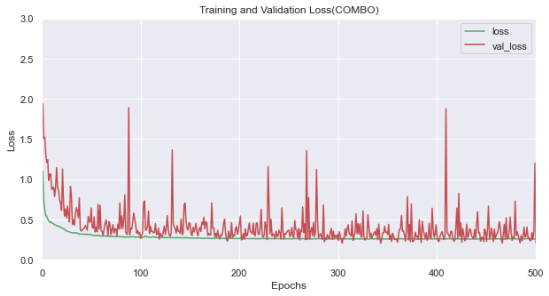

Fig. 7b: Training/Validation Loss plots using the CSE-CIC-IDS2018 Dataset (Drop-out (0.2))

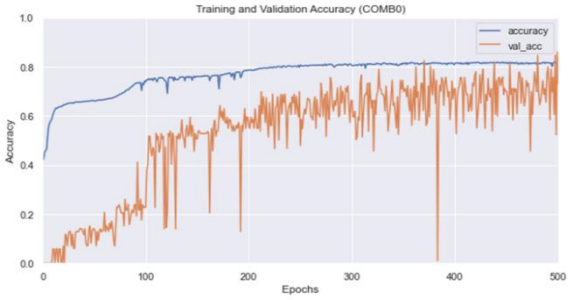

Fig. 8a: Training/Validation Accuracy plots using the CSE-CIC-IDS2018 Dataset (Drop-out (0.5))

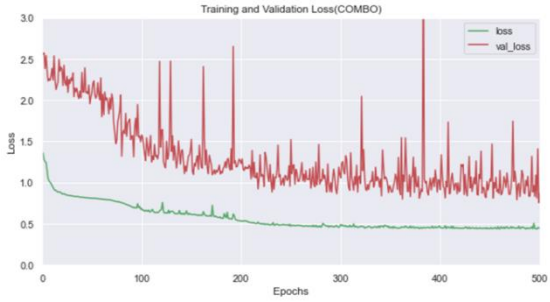

Fig. 8b: Training/Validation Loss plots using the CSE-CIC-IDS2018 Dataset (Drop-out (0.5))

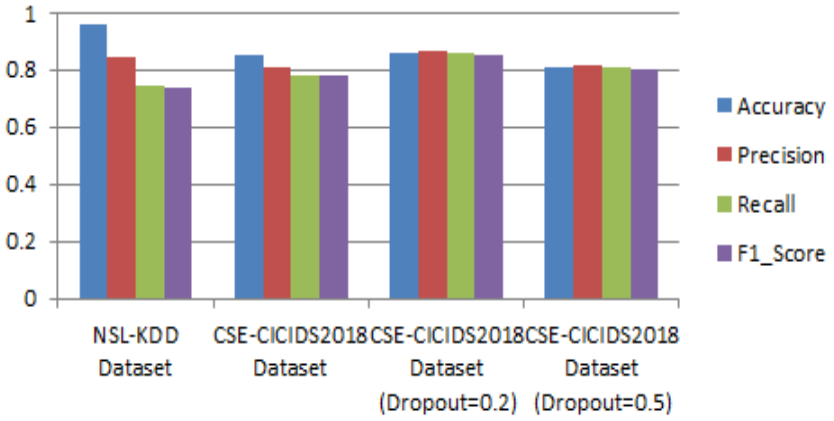

Fig. 9: Comparison of Model Performance Metrics using the various Datasets

From Figures 4a & 4b, model performance can be explained as follows. At the beginning of the training (0-50 epochs), training and validation accuracy were low as the model was just getting to learn about the data. In other words, the model's parameters (weights and biases) were still being adjusted/updated. As the number of epochs increased (50-100), both plots increased steadily and converged toward similar values, with only a small gap between them. This indicated that the model was learning effectively, fitting the training data well and generalizing to the validation (unseen) data. With increase in number of epochs (100+), the training accuracy approached the value of 1.0 (100%), indicating that the model was able to learn to accurately predict the training data. Likewise, for the associated training and validation loss, training loss was high at the beginning (0-50 epochs) with the validation loss being higher than the training loss. This was expected since the model was

Table 1: Performance Metrics (unbalanced NSL-KDD Dataset)

	Attack Category			
Metrics	**DoS**	**Probe**	**R2L**	**U2R**
Accuracy	0.88556	0.77176	0.77095	0.99314
Precision	0.90211	0.40022	0.38547	0.49657
Recall	0.87229	0.50000	0.50000	0.50000
F1-Score	0.87998	0.44458	0.43533	0.49828

Table 2: Performance Metrics for the Model using the balanced NSL-KDD Dataset

Metrics	**NSL-KDD Dataset**
Accuracy	0.9585
Precision	0.8451
Recall	0.7509
F1-Score	0.7459

Table 3: Metrics for the model using the CSE-CIC-IDS2018 Dataset

Metrics	**CSE-CIC-IDS2018 Dataset**
Accuracy	0.8490
Precision	0.8115
Recall	0.7802
F1-Score	0.7775

Table 4: Performance Metrics for the Model using CSE-CIC-IDS2018 dataset with varying drop out values

Metrics	**CSE-CIC-IDS2018 Dataset**	
	Dropout = 0.2	**Dropout = 0.5**
Accuracy	0.8564	0.8089
Precision	0.8689	0.8174
Recall	0.8564	0.8089
F1-Score	0.8547	0.8026

Table 5: Comparison of Model Performance using the NSL-KDD and CSE-CIC-IDS2018 datasets

Metrics	NSL-KDD Dataset	CSE-CIC-IDS2018 Dataset	CSE-CIC-IDS2018 Dataset (Dropout=0.2)	CSE-CIC-IDS2018 Dataset (Dropout=0.5)
Accuracy	0.9585	0.8490	0.8564	0.8089
Precision	0.8451	0.8115	0.8689	0.8174
Recall	0.7423	0.7802	0.8564	0.8089
F1-Score	0.7367	0.7775	0.8547	0.8026

just beginning to learn patterns from the training data and its parameters were just being updated. As the number of epochs increased (50-100), the model got better at fitting the training data and generalized well to unseen data. A small gap between the two indicated that the model is generalizing well with both loss values decreasing together. With increased number of epochs, the training loss approached the value of 0.0, indicating that the model is fitting the training data very well.

It can be observed that model performance with the balanced NSL-KDD dataset was better with the close matching of the training plot by the validation plot. This highlighted the importance of data balancing in intrusion detection. Therefore, a balanced CSE-CIC-IDS2018 dataset was used in the next section and results presented.

4.3 Using the Balanced CSE-CIC-IDS2018 Dataset

The unbalanced and balanced CSE-CIC-IDS2018 dataset are as presented in Figs. 5a &5b.

In the balanced CSE-CIC-IDS2018 dataset, all attack categories were equally represented thereby eliminating bias towards the larger class which can lead to poor performance of the IDS. Model training/validation accuracy and loss plots are as presented in Figures 6a & 6b.

The model was observed to have the following behaviour. From Fig. 6a, it can be seen that training and validation accuracy were low at the beginning of the training (0-50 epochs), as the model was just getting to learn about the data. At this time, the model's parameters (weights and biases) were still being adjusted/updated. It was also observed that validation took a while (0-150 epochs) to catch up to training accuracy. This can be attributed to the size and characteristic of the dataset. As the number of epochs increased (150+), both plots increased steadily, the gap between them narrowing and then converged toward similar values (about 220 epochs). This indicated that the model was learning effectively, fitting the training data well and generalizing to the validation (unseen) data. With increased number of epochs, the validation accuracy fluctuated about training accuracy; with higher values being an indication of overfitting. With increased number of epochs, the trend is that the training accuracy will approach the value of 1.0 (100%), indicating that the model was able to learn to accurately predict the training data. From Fig. 6b, associated training and validation loss were high at the beginning (0-50 epochs) with the validation loss being higher than the training loss. Again, this was expected since the model was just beginning to learn patterns from the training data and its parameters were just being updated. Both dropped steadily till about 200 epochs with the gap between them narrowing. It was also observed that validation took a while (0-200 epochs) to catch up to training loss which can be ascribed to the attributes of the dataset. With increased number of epochs (200+), the model got better at fitting the training data and generalized well to unseen data. With increased number of epochs, the trend is that the training loss will approach the value of 0.0, indicating that the model is fitting the training data very well.

The performance metrics of the model are as presented in Table 3 and this showed good performance values. However, there is room for improvement; especially for Precision and Recall due to their importance in intrusion detection. Thus, in a bid to improve performance, the dropout technique (a regularization technique) was employed. Its impact on performance can vary depending on the specific dataset and model architecture concerned.

Dropout values of 0.2 and 0.5 were applied and their associated training/validation accuracy and training/validation loss plots are presented (Figures 7a - 8b). Figs. 7a & 7b show the plots for a dropout value of 0.2. It can be seen that the validation accuracy rose quickly to meet up with training accuracy (less than 50 epochs) which means the model is able to learn quickly. Thereafter, validation accuracy stayed above the training accuracy which can be an indication of over-fitting. For training

and validation loss, both plots also converged quickly (less than 50 epochs). Figs. 8a & 8b show the plots for a dropout value of 0.5 where it can be seen that validation accuracy rose at a slower pace (0-200 epochs) towards training accuracy ((longer training) while validation loss also dropped at a slower pace towards training loss. Thereafter (above 200 epochs), there was gradual increase in validation accuracy and gradual decrease in validation loss. For training and validation accuracy, the trend is that the gap between them will continue to reduce and likewise for training and validation loss. This means that the model is generalizing well to unseen data.

The model performance metrics are presented in Table 4. It can be seen that for dropout value of 0.2, values improved significantly for precision, from 0.8115 to 0.8689 and for recall from 0.7802 to 0.8564. For dropout value of 0.5, improvement was marginal, from 0.8115 to 0.8174 for precision and from 0.7802 to 0.8089 for recall.

4.4 A Comparative Analysis of the Model Performance

A comparative analysis of model performance with the various datasets used is presented in Table 5. The selected CSE-CIC-IDS2018 dataset is the one with a dropout value of 0.2 as it gives the best performance. This is then compared to the benchmark NSL-KDD dataset. It can be observed that there is an improvement in precision and recall values; from 0.8451 to 0.8689 and from 0.7423 to 0.8564 respectively.

Another representation of the comparison is shown in Figure 9. It can be seen that with the exception of accuracy which experienced a reduction, the model performed well for the other metrics – precision, recall and F1-Score with increased values. This is despite the fact that the CSE-CIC-IDS2018 is a much larger dataset than the NSL-KDD. Usually, there is degradation in IDS performance when it is presented with a more complex dataset. Most models are trained with small datasets and exhibit performance reduction when large datasets are encountered, i.e. they are not scalable; scalability being the model's ability to generalize to new, unseen data. This proposed model is able to overcome this challenge as shown in Figure 9.

Of note is that, in anomaly detection, precision and recall are much more important than accuracy; with high precision indicating a low rate of false positives and high recall indicating the model is effectively capturing most of the anomalies. Thus, improved values for precision and recall indicate that the model can perform well with large datasets and is scalable. This will be very useful in IoT applications with its ever-increasing dataset due to additional devices joining the network.

5.0 Conclusion

The study proposed a DL-based IDS model that is capable of maintaining detection performance in IoT network when a dataset larger than that with which it was trained is encountered. A combination of two DL techniques, the RBM and the 1D-CNN was proposed. Two datasets - the benchmark NSL-KDD and CSE-CIC-IDS2018 were used for evaluation. With the unbalanced dataset, it was observed that the model was better suited for detecting DoS attacks (Accuracy=0.8856, Precision=0.9021, Recall=0.8723, F1-Score=0.8800). For the balanced NSL-KDD dataset, it was observed that the model performed well for all metrics – accuracy (0.9585), precision (0.8451) and recall (0.7423). For the selected CSE-CIC-IDS2018 dataset, the model performed well although accuracy experienced some reduction (from 0.9585 to 0.8564). For the other metrics, the model performed better with increase in values of precision from 0.8451 to 0.8689, recall from 0.7423 to 0.8564 and F1-Score from 0.7367 to 0.8547. This is despite the fact that the CSE-CIC-IDS2018 dataset is a much larger one. Improved precision and recall values indicate that the model performs well for intrusion detection in a large dataset. This will be very useful in IoT applications wherein the IoT dataset is continuously increasing due to addition of new devices to the network.

The study was limited mainly by hardware capability which restricted the number of hidden layers and neurons that was used. There was also the issue of overfitting. While there is room for improvement of model performance, the dynamic nature of attacks makes it difficult for any particular IDS to detect all types of attacks. Future work could explore possible performance improvement by increasing the number of hidden layers and neurons in the model while bearing in mind the likely shortcoming of increased computational and time complexity. Also, the use of better computational resources such as graphical processing units (GPUs) can be employed. In addition, the potential for the proposed IDS to be used in other areas of applications such as driverless vehicles, leak detection, etc., should be considered.

Acknowledgement

The Authors are grateful to their departments for providing a conducive research environment.

Conflict of interest

No financial or non-financial interests that are directly or indirectly related to the work submitted for publication.

Individual author's contributions

Conception: OGD, AAA
Design: OGD, AAA
Execution: OGD, AAA, AIOY
Interpretation: OGD, AAA, AIOY
Writing the paper: OGD, AAA, AIOY

References

Aleesa, A., Zaidan, B., Zaidan, A. and Sahar, N. M. (2020). Review of intrusion detection systems based on deep learning techniques: Coherent taxonomy, challenges, motivations, recommendations, substantial analysis and future directions. Neural Computing and Applications, 32(14), 9827-9858 https://doi.org/10.1007/s00521-019-04557-3

Ashiku, L. and Dagli, C. (2021). Network Intrusion Detection System using Deep Learning Network Intrusion, Procedia Computer Science, 185(June), 239–247. https://doi.org/10.1016/j.procs.2021.05.025

Businesswire. (2021). Global IoT Connectivity Market Analysis and Forecast Report 2021. In Businesswire. https://www.businesswire.com/news/home/20211015005387/en/Global-IoT-connectivity-Market-Analysis-and -Forecast-Report-2021/

Chacon, R. (2024). IoT: Technologies, Markets, and Forecasts for 2024. Retrieved June 8, 2024, from Integra Sources website: https://www.iotforall.com/all-about-iot-technologies-markets-and-forecasts-for-2024

Chandola, V., Banerjee, A. and Kumar, V. (2009). Anomaly detection: A survey. Association for Computing Machinery (ACM) Computing Surveys, 41(3), 71–97. https://doi.org/10.1145/1541880.1541882

Desbiens, F. (2023). What Is IoT? In What Is IoT? In Building Enterprise IoT Solutions with Eclipse IoT Technologies (pp. 3–23). https://doi.org/10.1007/978-1-4842-8882-5_1

Feng, F., Liu, X., Yong, B., Zhou, R. and Zhou, Q. (2019). Anomaly detection in ad-hoc networks based on deep learning model: A plug and play device. Ad Hoc Networks, 84, 82–89. https://doi.org/10.1016/j.adhoc.2018.09.014

Hassija, V., Chamola, V., Saxena, V. and Jain, D. (2019). A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures. IEEE Access, 7, 82721–82743. https://doi.org/10.1109/ACCESS.2019.2924045

Hooshmand, M.K. and Hosahalli, D. (2022). Network anomaly detection using deep learning techniques. Chinese Association for Artificial Intelligence (CAAI) Transactions on Intelligence Technology, 7(2), 228–243. https://doi.org/10.1049/cit2.12078

Hu, W., Cao, L., Ruan, Q., Wu, Q. (2023). Research on Anomaly Network Detection Based on Self-Attention Mechanism. Sensors, 23(5059). https://doi.org/10.3390/s23115059

Injadat, M., Salo, F., Nassif, A. B., Essex, A. and Shami, A. (2018). Bayesian optimization with machine learning algorithms towards anomaly detection. In Proceedings of IEEE Global Communications Conference (GLOBECOM), 1–6. https://doi.org/0.1109/GLOCOM.2018.8647714

Kimbugwe, N., Pei, T., & Kyebambe, M. N. (2021). Application of Deep Learning for Quality of Service Enhancement in Internet of Things : A Review. Energies, 14(19), 6384. https://doi.org/10.3390/en14196384

Koroniotis, N., Moustafa, N. and Sitnikova, E. (2020). A new network forensic framework based on deep learning for Internet of Things networks: A particle deep framework,''. Future Generation Computer Systems, 110, 91–106. https://www.sciencedirect.com/science/article/pii/S0167739X19325105

Kubat, M. (2021). An Introduction to Machine Learning (3rd ed.). Springer.

Lansky, J. A. N., Ali, S., Mohammadi, M., Majeed, M. K., & Rahmani, A. M. (2021). Deep Learning-Based Intrusion Detection Systems : A Systematic Review. IEEE Access, 9, 101574–101599. https://doi.org/10.1109/ACCESS.2021.3097247

Leevy, J. L., & Khoshgoftaar, T. M. (2020). A survey and analysis of intrusion detection models based on CSE - CIC - IDS2018 Big Data. Journal of Big Data. https://doi.org/10.1186/s40537-020-00382-x

Liu, H. and Lang, B. (2019). Machine learning and deep learning methods for intrusion detection systems: A survey. Applied Sciences, 9(20), 4396.

Maithem, M. and Al-sultany, G. A. (2021). Network intrusion

detection system using deep neural networks Network intrusion detection system using deep neural networks. Journal of Physics: Conference Series 1804 https://doi.org/10.1088/1742-6596/1804/1/012138

Manimurugan, S., & Al-mutairi, S. (2020). Effective Attack Detection in Internet of Medical Things Smart Environment Using a Deep Belief Neural Network. IEEE Access, 8. https://doi.org/10.1109/ACCESS.2020.2986013

Mezina, A., Burget, R. and Travesio-Gonzalez, C. M. (2021). Network Anomaly Detection With Temporal Convolutional Network and U-Net Model. IEEE Access, 9, 143608–143622. https://doi.org/10.1109/ACCESS.2021.3121998

Praveena, N. and Vivekanandan, K. (2021). A Review on Deep Neural Network Design and Their Applications. 7th International Conference on Advanced Computing & Communication Systems (ICACCS), 1495–1501. https://doi.org/10.1109/ICACCS51430.2021.9441826

Rafique S.H., Abdallah A., Musa N.S. and Murugan T. (2024). Machine Learning and Deep Learning Techniques for Internet of Things Network Anomaly Detection-Current Research Trends. Sensors (Basel), 24(6). https://doi.org/10.3390/s24061968

Rani, S., Kataria, A., Sharma, V., Ghosh, S., Karar, V., Lee, K. and Choi, C. (2021). Threats and Corrective Measures for IoT Security with Observance of Cybercrime: A Survey. Hindawi - Wireless Communications and Mobile Computing, 2021. https://doi.org/10.1155/2021/5579148

Rayes, A. and Salam, S. (2019). Internet of Things From Hype to Reality: The Road to Digitization (2nd ed.). Springer Nature Switzerland AG 2017, 2019. https://doi.org/10.1007/978-3-319-99516-8

Shone, Nathan, Tran Nguyen Ngoc, Vu Dinh Phai, and Q. S. (2018). A deep learning approach to network intrusion detection. IEEE Transactions on Emerging Topics in Computational Intelligence, 2(1), 41–50. https://ieeexplore.ieee.org/document/8264962

Ukil, Arijit, Bandyoapdhyay, Soma, Puri, Chetanya and Pal, A. (2016). IoT Healthcare Analytics : The Importance of Anomaly Detection. International Conference on Advanced Information Networking and Applications (AINA), 994–997. https://doi.org/10.1109/AINA.2016.158

Vinayakumar, R., Alazab, M., Soman, K., Poornachandran, P., & Al-nemrat, A. and Venkatraman, S. (2019). Deep Learning Approach for Intelligent Intrusion Detection System. IEEE Access, 99. https://doi.org/10.1109/ACCESS.2019.2895334

Wazid, M., Das, A. Shetty ,S., Gope, P. and Rodrigues, J. (2021). Security in 5G-Enabled Internet of Things Communication : Issues, Challenges, and Future Research Roadmap. IEEE Access (2021), 9, 4466–4489. https://doi.org/10.1109/ACCESS.2020.3047895

What is IoT? (2024). Retrieved from Amazon Web Services website: https://aws.amazon.com/what-is/iot/

Willone Lim, Kelvin Sheng Chek Yong, Bee Theng Lau, C. C. L. T. (2024). Future of generative adversarial networks (GAN) for anomaly detection in network security: A review. Computers & Security, 139(ISSN 0167-4048). https://www.sciencedirect.com/science/article/pii/S0167404824000348