Intrusion Detection System using CNNs and GANs

NABEEL REFAT AL-MILLI[1], YAZAN ALAYA AL-KHASSAWNEH[2*]
[1]Computer Science Department,
Zarqa University,
Zarqa
JORDAN

[2]Computer Science Department,
Isra University
Amman,
JORDAN

Abstract: -This study investigates the effectiveness of deep learning models, namely Generative Adversarial Networks (GANs), Convolutional Neural Networks with three layers (CNN-3L), and Convolutional Neural Networks with four layers (CNN-4L), in the domain of multi-class categorization for intrusion detection. The CICFlowMeter-V3 dataset is utilized to thoroughly evaluate the performance of these models and gain insights into their capabilities. The primary approach involves training the models on the dataset and assessing their accuracy. The GAN achieves an overall accuracy of 93%, while CNN-3L demonstrates a commendable score of 99.71%. Remarkably, CNN-4L excels with a flawless accuracy of 100%. These results underscore the superior performance of CNN-3L and CNN-4L compared to GAN in the context of intrusion detection. Consequently, this study provides valuable insights into the potential of these models and suggests avenues for refining their architectures. The conclusions drawn from this research indicate that CNN-3L and CNN-4L hold promise for enhancing multi-class categorization in intrusion detection systems. It is recommended to further explore these models with diverse datasets to strengthen overall comprehension and practical applicability in this crucial field.

Key-Words: - Intrusion Detection, Convolutional Neural Networks, Generative Adversarial Networks, Classification, Machine Learning, Anomaly Detection, Network Monitoring.

Received: March 29, 2023. Revised: February 15, 2024. Accepted: March 3, 2024. Published: April 19, 2024.

1 Introduction

The integration of artificial intelligence (AI) into various aspects of our lives has brought about a new era, characterized by the rise of deep learning. Deep learning has made significant contributions to the strengthening of programs, systems, and applications, particularly in the field of security. Within the realm of neural network architectures, different designs serve different purposes, such as analysis, classification, detection, and generation tasks. This study focuses on the important role of Generative Adversarial Networks (GANs) in the domain of intrusion detection, which is a critical aspect of cybersecurity.

Recognizing that malicious attacks often begin with intrusive efforts, referred to as security incidents, this study emphasizes the importance of preventing unauthorized access to systems and their repositories. In today's context, network attacks have become increasingly common, necessitating the use of advanced Intrusion Detection Systems (IDS). These IDS play a crucial role in identifying and mitigating threats, including both known and new intrusion attempts, by examining endpoint configurations.

As the cyber threat landscape continues to evolve, researchers have explored various methodologies to categorize and counter infiltration attempts. Previous studies have utilized a range of techniques, including clustering, convolutional neural networks (CNNs), deep belief networks, support vector machines (SVM), and naive Bayes. Notably, researchers have also delved into the use of unsupervised learning and anomaly detection through clustering methods, highlighting the complex nature of the challenge.

First of all, any attack begins with intrusion efforts, which are regarded as a security incident or collection of security incidents that make up a security event in which an intrusive party seeks to gain unrestricted access to a system or system's exchequer. Despite the fact that any assault on your network may seem intrusive. Network attacks are turning into a regular annoyance, [1]. The intrusion detection system (IDS) can look at endpoint configurations and detect threats and unauthorized attempts to access secure devices.

According to [2], artificial intelligence has the potential to solve many social, economic, and environmental challenges, but only if AI-enabled devices are safe. Many of the artificial intelligence (AI) models developed in recent years can be attacked using advanced techniques. This problem has led to intensive adversarial AI research to develop machine learning and deep learning models that can withstand different types of attacks. To demonstrate how adversarial attacks are performed against AI applications, this article has provided a comprehensive overview of artificial intelligence. These topics include adversarial knowledge and skills, current methods for creating adversarial examples in practice, and current cyber defense models. Furthermore, the author presented a rigorous methodology to present a war strategy against machine learning and artificial intelligence, and from such attacks he considered various cyber defenses that can protect AI applications.

There have been various researches that used deep learning and machine learning to categorize the infiltration attempts. Some of them relied on the use of clustering techniques to identify the attempts. Because of this, some of it utilized CNNs and was set up with reinforcement learning. It provides the environment, which is the working area and uses the endpoints that are included in the dataset, to the system or agent, [3].

The studies used a variety of methods to detect attacks, including deep belief networks and hybrid based probabilistic NNs (Neural Networks), [4]. Some research publications also used SVM and naive Bayes to create the intrusion detection system. Based on the approach of object detection, [5], [6] advocated utilizing deep belief networks (DBN) for the detection process, [7], studied the use of CNN for feature mapping and the detection process, and [8], advised employing GANs with the deployment of several layers. [9], Utilized CNN for representation learning to aid the system's self-detection. [10], used deep learning techniques, such as DBN, for the implementation of an IDS (Intrusion detection system). [11], suggested to employ unsupervised learning by taking the dataset's output without the labeling data (y), then examining it with clustering methods for anomaly detection. In this study, we used three distinct deep learning approaches to categorize the system infiltration attempts.

In [12], the NSL-KDD dataset is investigated in order to rectify some of the issues discovered in the KDD cup99 data. The findings indicate that an NSL-KDD dataset is a useful tool for investigating and comparing various intrusion detection strategies. The time-consuming procedure of looking for intrusive patterns that use all 41 features in the dataset may degrade system performance. The dataset contains unnecessary data that does not directly assist the project at hand. When working with a dataset, the CFS Subset is used to reduce the number of dimensions that are important. On the dataset, various classification techniques were tested, both with and without feature reduction. Random Forest, on the other hand, achieves the best test accuracy when compared to the other algorithms. Random Forest, according to the results of this experiment, can greatly accelerate training and testing processes for intrusion detection, a critical application in the field of network security, even when dealing with a small feature set. It may be possible to improve the Random Forest algorithm, which is currently employed in many intrusion detection systems, in the near future.

Our approach falls under the category of multiclass classification. Then use the dataset we selected and the settings we specified to train each algorithm we developed. We will present the final finding, which compares the accuracy of the three methodologies to the models that were examined and tested in the aforementioned investigations.

2 Related Work

We'll talk about related studies that have already been conducted in the same area of research.

2.1 Deep Belief Network

First, the most crucial characteristics of the raw data had to be adjusted. Then, using nonlinear learning, the data was translated into low-dimensional data, also known as (low dimensions data). The PNN (probabilistic neural network) was chosen because the KDD CUP data set, which was published in 1999, may be classified using low-dimensional data. The benefits of DBN are demonstrated by its use to reduce the dimensionality of the data during the preprocessing stage and by the strong performance of learning representations. Additionally, the volume of data and shorter training times make it simpler to find the local optimum. The PSO technique is used to increase the number of DBN nodes in the hidden-layer in order to improve the performance of the DBN network with the necessity of expressing the features. The experimental findings in [13], demonstrated that combining deep learning with PSO can help with data dimensionality problems. Deep networks and shallow NIDSs were the methods that the authors of [14], examined for efficiency. On datasets like KDD 1999 identical, [13] and NIDS, such as classification preferences, the deep and shallow networks were trained and evaluated. When the results of deep and shallow networks were compared, it was found that deep networks were more effective at detecting attacks, [15].

2.2 SVM and Naïve Bayes

The activity is equal to the machine by using the returned annotations for random classifiers such as support vector machines and naive bayes, which are based on deep learning and are viewed as efforts at attacks based on experiments, such as black box attacks. The outcome showed that Deep Learning is capable of using SVM and Naive Bayes classifiers for text classification applications and can allocate them to tasks. Thus, great precision was attained, [16]. Due to the expansion of newly acquired original mitigation strategies with high ability and capacity, this situation created additional security challenges for the new attack chart with deep learning toward online machine learning algorithms to target on the highly accurate of attacks using deep learning.

2.3 CNNs

The question of whether Deep Learning systems can distinguish between various harmful attacks and lessen Android attack was investigated by the authors in [8]. The system was created using Convolutional neural networks, and the authors in [8], applied it to system call occurrences by selecting one sort of analysis, namely dynamic analysis. A recent dataset comprised of over 7000 real- world apps produced accuracy results for the model between 85% and 95%.

2.4 GANs

The Generative Adversarial Network (GAN) comprises of two feed-forward neural networks, one of which is the Generator (G), and the other of which is the Adversary (D), which estimates the quality of the former and creates samples that are similar to the real ones. Each of the two networks is primarily a deep neural network (DNN), [9], with varying numbers of layers connected in such a way that the output of the devices in each layer serves as the input for the devices immediately above it. The class difference in using supervised classification, a method was provided to address the credit card fraud detection problem in an enhanced set was developed as a training group that carries additional periods of an opportunity class in comparison to the authentic set, [14]. Synthetic examples were produced using a tweaked generative adverse network (GAN). Given the GAN discriminator factor, this approach can discriminate between real samples and synthetic ones. The suggested framework made progress while maintaining accuracy, providing a limited rise. The authors of [17], cautioned against using the Generative Adversarial Networks (GAN) model, an uncontrolled deep learning technique that creates new digital data that is identical to the current data, to correct the imbalance in the data. Additionally, it suggested a Random Forest model to select detection performance following the correction of data imbalances using a GAN. The test results showed that the version suggested in this research performed better overall than the version labeled without resolving the imbalance of facts. Additionally, compared to all other models, it was found that the performance of the proposed version was the best overall. The authors of [11], suggested a method for an online device that performs unsupervised feature reduction using a one-hidden layer RBM. The additional RBM that creates the deep thought network receives the resultant weights from this RBM. Authors in [11], employed a deep mastering structure using the DARPA KDDCUP' 1999 dataset to evaluate the effectiveness of a Logistic Regression classifier with multi-class SoftMax activation function that was fed the previously pre-trained weights. Their design performs better in terms of detection accuracy and speed than earlier deep mastery systems. The structure was successfully detected, with a 97.9% accuracy ratio on the entire 10% KDDCUP' ninety-Nine test dataset. They were also able to lower a low fake awful fee of 2.47% by improving the simulation's educational methodology. Even if the KDDCUP' ninety-Nine dataset has flaws that are well acknowledged, it nonetheless offers machine learning strategies for anticipating assault attempts with a manageable challenge.

3 Methodology

The research employed a methodology that aimed to rigorously assess the effectiveness of three different models for multi-class intrusion detection. These models included Generative Adversarial Networks (GANs), Convolutional Neural Networks with three layers (CNN-3L), and Convolutional Neural Networks with four layers (CNN-4L). The evaluation of these models was conducted using the CSE-CIC-IDS2018 dataset, which was chosen for its inclusion of both malicious and benign network traffic, encompassing various attack scenarios and profile classes.

The methodology began by thoroughly examining and preprocessing the dataset. This involved analyzing the dataset in CSV format, reducing the number of features, and balancing the samples. Following this preprocessing stage, the three models were trained using the preprocessed data. The GANs architecture employed a generator-discriminator framework, while the CNN models utilized convolutional layers, pooling layers, and fully-connected layers.

During the training process, the models were exposed to different attack categories, ensuring a comprehensive evaluation of their performance. The outcomes of this evaluation, measured in terms of accuracy, revealed that CNN-3L and CNN-4L outperformed the other models. This highlights their potential for advancing the categorization of multiple classes in intrusion detection systems.

This section of the research provides a detailed overview of the methodology's systematic approach to model training, validation, and testing. It establishes a strong foundation for the subsequent analysis and discussion of the results.

3.1 The Dataset

The CSE-CIC-IDS2018 dataset is the one we used. This dataset, which we use, consists of both malicious traffic produced by various network attacks and benign (good) network traffic. A good way to include thorough descriptions of intrusions and summary distribution algorithms for applications, protocols, or low-level network approaches is the CSE-CIC-IDS2018 proposed by UNB-The Canadian Institute for Cyber Security, [18], a technique of notions to generate datasets under scientific conditions. Marketers or human operators may utilize these profiles to produce activities and make decisions at the community level. The dataset comprises seven distinct assault scenarios because of the concept nature of the created profile: brute-force, botnet, dos, dos, web attacks, and community infiltration through stealth. The victimized firm contains five departments, 420 machines, and 30 servers, whereas the attacking infrastructure consists of 50 machines. The dataset includes 80 functions that were extracted from the community site visitors using the CICFlowMeter-V3 tool, together with the machine logs of every machine and the collected community site visitors. Two excellent profile classes are included in the dataset: (a) B-profile, which may be viewed via HTTPS, HTTP, SMTP, POP3, IMAP, SSH, and FTP. M-Profiles Make an effort to clearly explain an assault scenario. In the best-case scenario, individuals can interpret those profiles and ultimately express them. We can identify three types of online attacks in this study: brute-force password guessing, cross-site scripting, and SQL injection (Brute Force-Web). The dataset was used as a CSV file.

3.2 Reading the Dataset

The dataset we have is made up of csv files with rows and columns that represent the items and features that each item has, respectively. The dataset was read using the Pandas library to display the types of attacks it has, including 104,000 benign files, brute force-web 362, brute force-XSS 151, and SQL injection 53. The dataset was then read using the Python programming language to display each data sample's row and column information.

3.3 Data Pre-Processing

Every sample in the dataset has 80 features; therefore, to raise the dimensionality during system Training, it is crucial to increase the number of features by reducing the number of features and select the most effective features in accordance with prior research on the same dataset, [19]. The number of characteristics has been decreased from 80 to14. The Dataset has been divided into 4 groups, each of which has the identical features (14 features total), namely benign, brute-force-XSS, brute-force-web, and SQL Injection. Organized it into a single directory. Additionally, we needed to balance the dataset's data samples for each group. In Figure 1, the data samples of each class are depicted.

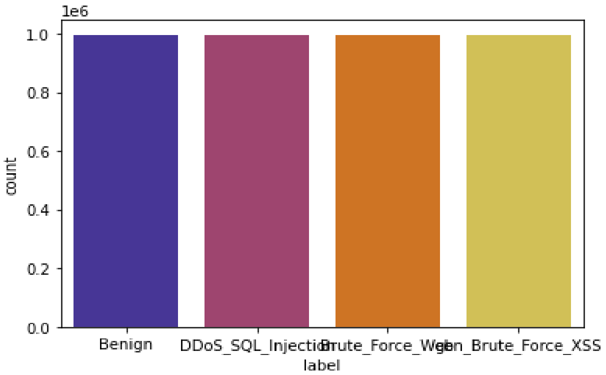

Fig. 1: Data samples of each class

3.4 The Model

This study utilizes a trio of models to examine intrusion detection, ensuring a strong and fair evaluation. The selected dataset serves as a consistent basis for comparing the models, promoting reliability in assessments. Within this framework, two Convolutional Neural Network (CNN) models are developed, each employing different techniques, while a Generative Adversarial Network (GANs) takes the lead as the initial detection model. The GANs architecture operates on a generator-discriminator paradigm, enhancing pattern recognition within network traffic. Subsequently, the second detection model adopts a traditional CNN design, highlighting the effectiveness of convolutional layers in extracting features. The third model introduces a sequential CNN, increasing network depth for advanced abstraction and representation learning. This deliberate model selection provides a comprehensive exploration of both conventional and innovative strategies, offering nuanced insights into their effectiveness and limitations in the field of intrusion detection. The subsequent discussion will delve into the intricacies of each model, clarifying their unique attributes and providing a detailed analysis of their respective performances.

3.4.1 GANs

Boltzmann machines use Markov chains to approximate patterns from the statistics distribution (superb samples), which we try to model, rather than patterns from our generative models, which assign samples z from the earlier p (z) to the statistics area. These generative models use supervised learning to approximate an intractable fee feature. To the greatest extent possible to deceive the discriminator into thinking that the samples it produces are drawn from the statistics distribution. GANs are a clever technique to educate a generative model because the generator is trained by updating the real dataset.

1-The generator: Generative models that can just produce facts rather than providing an estimate of the density function. After all, when applied to photographs, such models seem to only provide more images, but the generative version is successful in distributing the dataset's high dimensionality of functions and metrics. Training and sampling from generative models is a good way to assess how well we can represent and manage high-dimensional random distributions. In particular, GANs are fairly good at doing semi-supervised learning, and they can be trained with missing data and can make predictions on inputs with missing data. The concept of the most probability is to describe a version that supplies an estimate of a chance distribution, parameterized by parameters θ. The generative fashions make use of maximum probability estimation. Its purpose is to give the version a parameter so that it can be.

2-The Discriminator:

The discriminator determines whether the facts are true or false, learns how to employ traditional supervised learning techniques, and divides inputs into two classes (actual or faux). The discriminator can be duped by the generator, [20]. The discriminator, like the police, aims to allow legitimate cash while catching counterfeit cash, whereas the generator is like a forger trying to create fake money. In order to win this game, the counterfeiter must figure out how to produce currency that cannot be distinguished from real money, and the generator community must figure out how to provide samples that can be drawn from the same distribution as the training data. Formally, GANs are a creation to dependent containing latent variables, and they are a dependent probabilistic model, [21]. Modern image generation and manipulation systems depend heavily on GANs, and they have the potential to support a wide range of additional applications in the future.

In our research article, we deployed a system to identify network intrusion attempts using the GANs architecture. We implemented the generator and the discriminator in distinct classes, class generator and class discriminator, using Keras from the Tensor flow package. Figure 2 displays the discriminator and generator components. Four fully connected layers (FC layers) with ReLu activation function make up the generator. In Figure 3, the layers of the generator are presented. Four completely linked layers make up the discriminator, with the first two fully connected layers being followed by dropout layers (reason of using dropout layers). In Figure 4, the discriminator is shown, illustrating its components and structure.

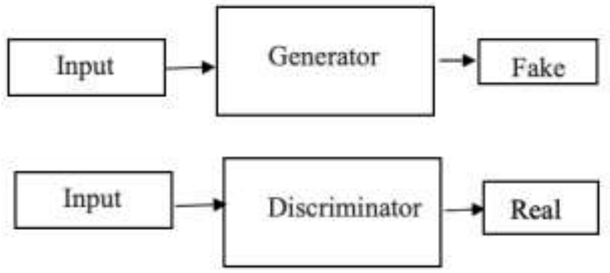

Fig. 2: discriminator and generator

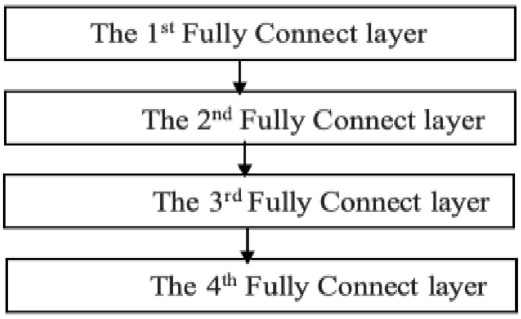
The 1st Fully Connect layer
The 2nd Fully Connect layer
The 3rd Fully Connect layer
The 4th Fully Connect layer

Fig. 3: Generator Layer

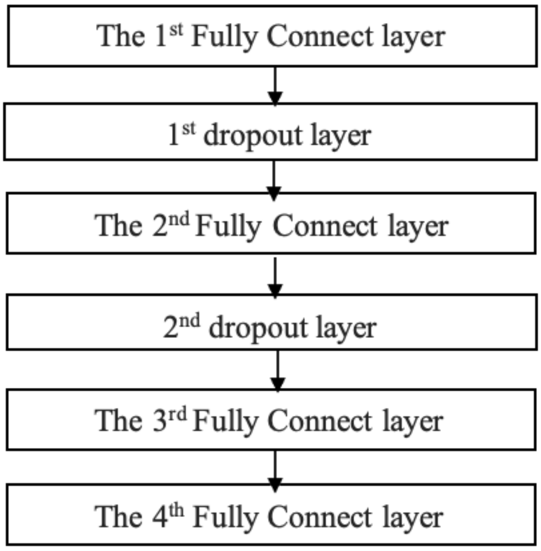
The 1st Fully Connect layer
1st dropout layer
The 2nd Fully Connect layer
2nd dropout layer
The 3rd Fully Connect layer
The 4th Fully Connect layer

Fig. 4: Discriminator

3.4.2 CNN

Due to CNN's strong performance, a wide range of packages built entirely around the CNN version have been presented. Convolutional neural networks stand out from other neural networks thanks to their superior performance with picture, speech, or audio signal inputs.

Fully-linked (FC) layer, Convolutional layer, and pooling layer, [22]. The sequential fashions are a great fit for our kingdom because we have a multi-elegance class project, as it is indicated in [23]. The first layer of a convolutional network is the convolutional layer. The fully-linked layer is the final layer, even though convolutional layers may be supplemented by utilizing further convolutional layers or pooling layers. The CNN will become more complicated with each layer as it determines additional portions of the image. Layer Convolution The convolutional layer, the middle building component of a CNN, [24], is where the majority of computation takes place. It requires a few things, including enter data, a filter, and a function map.

Let's assume that entry may be a shaded image, which is composed of a matrix of sometimes 3D, sometimes 2D, and sometimes 1D pixels. It depends on the project and the preprocessing to determine how the final output should be and what the best preparation method is. Down sampling, sometimes referred to as pooling layers, carries out dimensionality reduction and lowers the amount of parameters in the input. The pooling operation sweeps a filter across the entire input similarly to the convolutional layer, with the exception that this filter lacks weights. Instead, the kernel populates the output array by applying an aggregation function to the values in the receptive field. There are various pooling layer types, and they change depending on the workloads. The term "fully connected layer" accurately captures what this layer is. As stated previously, with partially related layers, the pixel values of the input photograph aren't immediately associated to the output layer. However, every node inside the output layer connects instantly to a node inside the previous layer with inside the fully-related layer. Based on the information acquired from the previous layers and their unique filters, this layer performs the classification task. While FC layers frequently use a softmax activation characteristic to classify, we inputs correctly, generating a probability from zero to one, while convolutional and pooling layers typically utilize ReLu functions. We will advise CNN models, instruct them, and evaluate the outcomes. Also, the second iteration of CNN consists of a dropout layer in between a convolutional layer and a pooling layer with dense layers. Softmax is used as the final layer in every CNN version. We suggested the use of certain pooling layers after preprocessing the dataset, which had one convolutional layer and a few dense layers. On the dataset we recommended, we trained the three models using 70% of it for training and 30% for testing. First, we trained the generator using GAN on the dataset's benign samples, then we trained it using Web brute force attacks, then we trained using XSS brute force, and last, we trained the generator using the dataset's SQL injections section.

We trained the entire dataset on the first CNN model, which has three layers; we did the same for the second CNN model, which has four layers; the training phase took place over the course of 2500 epochs; the input shape was 32 x 512; and the log steps for the GAN were about 128. After the 20th iteration during training, the GAN showed 97.66% for the innocuous files, 98.24% for the XSS brute force category, 98.34% for the brute force web category, and 89.06% for SQL injection. The first

CNN showed 99.71%, and the most recent CNN model has 100% accuracy. We trained the models using the Adam optimizer. The results of our training process are summarized in Table 1.

Table 1. GAN accuracy according each class category

Dataset category	Accuracy
Benign	97.66%
XSS brute force	98.24%
Brute force web	98.34%
SQL injection	89.06%

The approximate accuracy for the GAN is 95.82% for all the categories that exist in the dataset. In Figure 5, the accuracy of the CNN3L model is visualized after the training process.

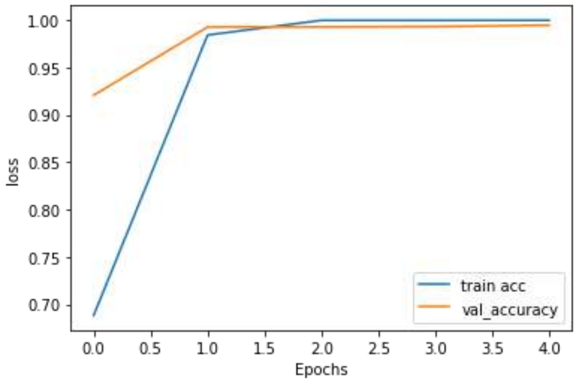

Fig. 5: Accuracy for the CNN3L model after training

The model's accuracy was 99%, as shown in Table 2.

Table. 2 Training and validation accuracy and loss for the CNN3L model

	Accuracy	Loss
Training	99.68%	0.0283
Validation	99.69%	0.0193

In the CNN technique, we attempted to categorize the intrusion attempt category by evaluating the precision and F1 score when the samples of the categories are equal. The confusion matrix is a summary of the performance of classification algorithms, [25]. Table 3 and Table 4 exhibit the confusion matrices for CNN3L and CNN4, respectively. While Figure 6 and Figure 7 visualize the confusion matrices for CNN3L and CNN4, respectively.

Table 3. Confusion matrix for the CNN3L

	Precision	Recall	F1 score	support
Benign	0.99	1.00	0.99	2504
Xss brute force	1.00	0.99	0.99	2468
Brute force web	1.00	1.00	1.00	2537
SQL injection	1.00	1.00	1.00	2491

Table 4. Confusion matrix of CNN4

	Precision	Recall	F1 score	support
Benign	1.00	1.00	1.00	2504
Xss brute force	1.00	1.00	1.00	2468
Brute force web	1.00	1.00	1.00	2537
SQL injection	1.00	1.00	1.00	2491

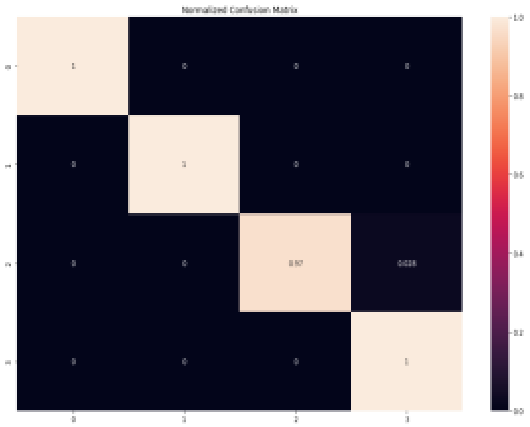

Fig. 6: Confusion matrix for the CNN3L

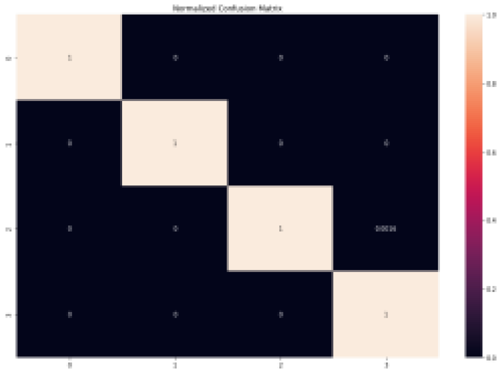

Fig. 7: Confusion matrix of CNN4L model

With the offered dataset, we proposed three deep learning methods in our research. Using the dataset we already had, we trained the algorithms on it. We trained the GAN four times, each time using a different section of the dataset. Its testing accuracy was roughly 93%, compared to 100% for the CNN-4L and 99.71% for the CNN-3L. The models fared better than the earlier studies, particularly the two CNN models. In Table 5, the training and validation accuracy and loss for the CNN4L model are outlined.

Table 5. Training and validation accuracy and loss for the CNN4L model

	Accuracy	**Loss**
Training	99.9%	0.0047
Validation	100%	0.0012

The model showed 100% accuracy. In Figure 8, the accuracy of the CNN4L model is visualized after the training process.

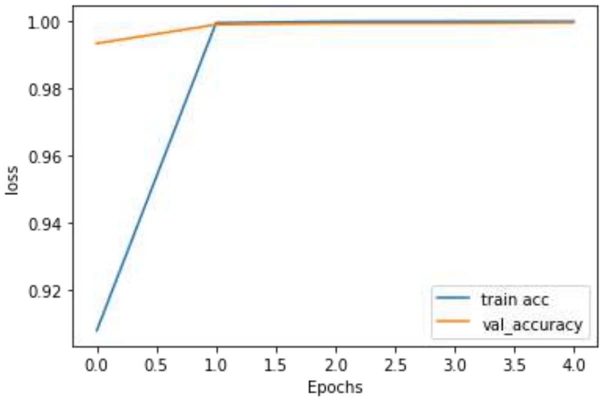

Fig. 8: Accuracy of CNN4L model after training

Table 6 compares the models we proposed, CNN3L, CNN4L, and GAN with a certain number of layers for the generator and discriminator, to the models that have been explored and developed in earlier works.

Table 5. Comparison between the models

The model	Accuracy
SVM	97.44%
Naïve Bayes	97.81%
CNN	85~95%
GAN	99.19%
DBN	03.25%
CNN3L	99%
CNN4L	100%
GAN in our study	95%
RBM	97.9%
Clustering	65.7%

4 Conclusion

To summarize, this study provides valuable insights into the performance of three different models—Generative Adversarial Networks (GANs), Convolutional Neural Networks (CNN), and Sequential CNN—in the field of intrusion detection using the CICFlowMeter-V3 dataset. The observed accuracies, ranging from 93% for GAN to a perfect 100% for Sequential CNN, highlight the significant potential of these deep learning models in enhancing intrusion detection systems. However, it is important to acknowledge the limitations of this study. The reliance on a specific dataset raises concerns about the generalizability of the findings, necessitating future investigations with diverse datasets to validate and improve the models' robustness across various scenarios.

Furthermore, this study primarily focuses on the technical aspects of model performance, leaving room for future research to explore the practical challenges associated with implementing these models in real-world settings. Proposed improvements include refining model architectures, exploring ensemble methods to leverage the strengths of multiple models, and adopting advanced techniques to effectively combat evolving cyber threats.

Moving forward, the research should go beyond technical proficiency and incorporate explainable AI techniques to enhance model interpretability. This is crucial for establishing greater trust in intrusion detection systems, considering the critical nature of the security domain. This study sets the foundation for further exploration, and future endeavors should prioritize addressing the identified limitations to steer the field towards more practical, adaptable, and reliable intrusion detection solutions.

References:

[1] Asiri, M., Saxena, N., Gjomemo, R., & Burnap, P. (2023). Understanding indicators of compromise against cyber-attacks in industrial control systems: a security perspective. *ACM Transactions on Cyber-Physical Systems*, 7(2), 1-33.

[2] Al-Khassawneh, Y. A. (2023). A review of artificial intelligence in security and privacy: Research advances, applications, opportunities, and challenges. *Indones. J. Sci. Technol*, 8, 79-96.

[3] Surakhi, O., Garcia, A., Jamoos, M., & Alkhanafseh, M. (2022). The Intrusion Detection System by Deep Learning Methods: Issues and Challenges. *International Arab*

Journal of Information Technology, 19(3 A), 501-513.

[4] Yan, R., Xiao, X., Hu, G., Peng, S., & Jiang, Y. (2018). New deep learning method to detect code injection attacks on hybrid applications. *Journal of Systems and Software*, 137, 67-77.

[5] Zhao, G., Zhang, C., & Zheng, L. (2017, July). Intrusion detection using deep belief network and probabilistic neural network. *In 2017 IEEE international conference on computational science and engineering (CSE) and IEEE international conference on embedded and ubiquitous computing (EUC),* Vol. 1, pp. 639-642.

[6] Vinayakumar, R., Soman, K. P., & Poornachandran, P. (2017, September). Evaluating effectiveness of shallow and deep networks to intrusion detection system. *In 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 1282-1289.

[7] Al-Milli, N., & Almobaideen, W. (2019, April). Hybrid neural network to impute missing data for IoT applications. *In 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT),* pp. 121-125.

[8] Martinelli, F., Marulli, F., & Mercaldo, F. (2017). Evaluating convolutional neural network for effective mobile malware detection. *Procedia computer science*, 112, 2372-2381.

[9] Fiore, U., De Santis, A., Perla, F., Zanetti, P., & Palmieri, F. (2019). Using generative adversarial networks for improving classification effectiveness in credit card fraud detection. *Information Sciences*, 479, 448-455.

[10] Alrawashdeh, K., & Purdy, C. (2016, December). Toward an online anomaly intrusion detection system based on deep learning. *In 2016 15th IEEE international conference on machine learning and applications (ICMLA)*, pp. 195-200.

[11] Portnoy, L. (2000). Intrusion detection with unlabeled data using clustering (Doctoral dissertation, Columbia University).

[12] Y. A. Al-Khassawneh, "An investigation of the Intrusion detection system for the NSL-KDD dataset using machine-learning algorithms," *2023 IEEE International Conference on Electro Information Technology (eIT)*, Romeoville, IL, USA, 2023, pp. 518-523, doi: 10.1109/eIT57321.2023.10187360.

[13] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S.,& Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11), 139-144

[14] Yadav, S., & Subramanian, S. (2016, March). Detection of Application Layer DDoS attack by feature learning using Stacked AutoEncoder. *In 2016 international conference on computational techniques in information and communication technologies (ICCTICT),* pp. 361-366.

[15] Ola Surakhi,Antonio García,Mohammed Jamoos,Mohammad Alkhanafseh, "The Intrusion Detection System by Deep Learning Methods: Issues and Challenges", *The International Arab Journal of Information Technology (IAJIT)* ,Vol. 19, Number 3A, pp. 501 - 513, Special Issue 2022, doi: 10.34028/iajit/19/3A/10.

[16] Shi, Y., Sagduyu, Y., & Grushin, A. (2017, April). How to steal a machine learning classifier with deep learning. *In 2017 IEEE International symposium on technologies for homeland security (HST),* pp. 1-5.

[17] Rao, Y. N., & Suresh Babu, K. (2023). An imbalanced generative adversarial network-based approach for network intrusion detection in an imbalanced dataset. *Sensors*, 23(1), 550.

[18] Dunmore, A., Jang-Jaccard, J., Sabrina, F., & Kwak, J. (2023). A Comprehensive Survey of Generative Adversarial Networks (GANs) in *Cybersecurity Intrusion Detection. IEEE Access.*

[19] Al-Milli, N., Hudaib, A., & Obeid, N. (2021). Population diversity control of genetic algorithm using a novel injection method for bankruptcy prediction problem. *Mathematics*, *9*(8), 823.

[20] Poongodi, M., & Hamdi, M. (2023). Intrusion detection system using distributed multilevel discriminator in GAN for IoT system. *Transactions on Emerging Telecommunications Technologies*, vol. 34 (11), e4815, https://doi.org/10.1002/ett.4815.

[21] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., & Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11), 139-144.

[22] Hamandi, H. R. (2022). Modeling and Enhancing Deep Learning Accuracy in Computer Vision Applications. Wayne State University, 29254756.

[23] Even-Zohar, Y., & Roth, D. (2001). A sequential model for multi-class classification. arXiv preprint cs/0106044, [Online]. https://aclanthology.org/W01-0502.pdf (Accessed Date: February 2, 2023).

[24] Ullah, F., Ullah, S., Srivastava, G., & Lin, J. C. W. (2023). IDS-INT: Intrusion detection system using transformer-based transfer learning for imbalanced network traffic. *Digital Communications and Networks*, https://doi.org/10.1016/j.dcan.2023.03.008.

[25] Nti, I. K., Narko-Boateng, O., Adekoya, A. F., & Somanathan, A. R. (2022). Stacknet Based Decision Fusion Classifier for Network Intrusion Detection. *International Arab Journal of Information Technology*, 19(3 A), 478-490.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The authors equally contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

Zarqa University, 2000 Zarqa 13110 Jordan.

Conflict of Interest

The authors have no conflicts of interest to declare

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)