Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Intelligent Anomaly Detection in Distributed System Using Deep Learning Techniques

Vineeta Shrivastava[1], Megha Kamble[2], Vaibhav Udgir[3]

[1]PhD Research Scholar, LNCT University, Bhopal, shrivastavavinita21@gmail.com
[2]Professor, LNCT University, Bhopal, meghak@lnctu.in
[3]Software Engineer, Think Future Technologies, Bhopal, vaibhavudgir@gmail.com

The rapid growth of IoT devices has increased attack points, making cybersecurity crucial. Intrusion Detection Systems (IDSs) help manage networks, alerting to malicious traffic. Research focuses on zero-day attacks, with deep learning techniques needed for effectiveness. In this paper we propose deep leaning to improve the intrusion detection system based on CGAN for class imbalancing, ReseNet based learning to train the model and Vgg16 and Vgg19 models to classify the attacks for binary classifiers. The experimental results show that the proposed model achieves 95% accuracy using Vgg16 model and 94.3% using Vgg19 model for binary classifier.
Keywords: Intrusion detection system, Conditional GAN, Deep neural networks, Labelled data, Network flow monitoring.

1. Introduction

Network science is advancing rapidly around the domain, making it incredibly easy to share information. However, this rapid development also brings many challenges to communication systems, making them vulnerable to numerous kinds of assaults. An Intrusion Detection System (IDS) is a tool that uses detection algorithms to identify potential cyberattacks on a host or network. Basically IDS are categorized into two categories namely signature-based intrusion detection system (SIDS) and anomaly-based intrusion detection system (AIDS). Signature-based IDS (SIDS): These systems detect attacks by looking for a known pattern or signature of an attack. Anomaly-based IDS (AIDS): These systems monitor traffic patterns and compare them to what is considered normal or typical for the network. If there is any deviation from the norm, it is flagged as a possible intrusion

There are several methods for implementing Signature-based Intrusion Detection Systems (SIDS) and Anomaly-based Intrusion Detection Systems (AIDS). The limitations of SIDS can be addressed by using AIDS, which has become a growing area of interest for

researchers. Statistical approach these use statistical data, like variance, standard deviation, mean, mode, to detect intrusions. Statistical IDS can be implemented using time-series, multivariate, and univariate models. Knowledge-based Approaches: These methods build models centered on protocols derived from social expertise. Techniques such as machine learning and deep learning like SVM, KNN, Decision tree, and linear regression these approaches leverage algorithms to learn and detect anomalies in network traffic. In developing knowledge-based Intrusion Detection Systems (IDS), tools such as expert systems, finite-state systems, and description languages are commonly used. Another popular method for creating anomaly-based IDS is machine learning, which are categorized into two groups one is supervised learning and other is unsupervised learning. The unsupervised learning approach uses unlabeled data to find patterns without prior knowledge of the outcomes, while supervised learning relies on labeled data to train the system with known results. Additionally, there is a hybrid approach known as hybrid based learning, which is a grouping of mostly not labeled data with a smaller amount of labeled data to improve training efficiency and accuracy.

Contribution of Paper

We proposed an IDS using a deep learning method to effectively categorize the attacks. To find the attacks in the IDS, we employed Vgg16 and Vgg19. As was covered in the preceding part, the current DL approach has drawbacks that are addressed by modifying it with a learning framework based on CGAN and ResNet. We experimented with several configurations of Vgg16 and Vgg19 by obtaining more functionality. With the vgg16 model, we did, however, get faster speeds. The latter section displays the comprehensive model strategy. To improve performance, we also improved the UNSW-NB15 dataset. The enhancement process involves choosing unique characteristics and eliminating superfluous capabilities.

Organization of Paper

The rest of the work is presented in following manner: The second phase describes the latest studies on this topic. Third phase gives the description about CGAN for network imbalancing. The work done is shown in phase 4. This portion also covers the study's primary impact and goes into considerable length on the suggested methodology. Phase five examines how the model performed based on a number of variables, such as conclusions and discussions. The study’s work comes to conclude in phase six.

2. Literature Review

Gamage et al.[1] provide a taxonomy of transfer learning models for anomaly recognition, revealing that auto encoders along with neural networks are not capable to outperform supervised feed-forward neural networks.

Kasim et al.[2] proposed an efficient deep learning approach, auto encoder support vector machine, combined with Canadian institute of intrusion detection system, effectively apprehensions fundamentally created Distributed Denel of Service network load, achieving 99.1% success in detection using Kali Linux.

Rani et al.[3] proposes a uniform detection method using supervised machine learning and Random Forest classifier, achieving 99.9% accuracy in intrusion detection using NSL-KDD

and KDDCUP99 datasets.

Bharti et al.[4] combined Intrusion Detection System with Machine Learning Based (Random Forest) achieved an impressive score of 99% in the CSE-CIC-IDS-2018.

Gao et al.[5] developed a deep learning based model namely feed forward neural network detects periodically unrelated attacks with accuracy of 99%, however the LSTM based algorithm detects correlated attacks with an F1 of 99.68±0.04%.

Alsoufi et al.[6] provides an overview of deep learning based anomaly detection system revealed high rate of flase alarm and accuracy, recommending further research for robust IDS.

Emadi et al.[7] employs deep learning techniques like LSTM and transfer learning to develop an effective intrusion detection model for detecting network intrusions, comparing their results for optimal performance.

Lee et al.[8] Secure shell brute-force and distributed denial-of-service attacks in SDN are effectively prevented by the introduction of a deep learning-based intrusion detection and prevention system (DL-IDPS), achieving near 99% and 100% accuracy respectively.

Musa et al.[9] presents reviews various studies on effectual Intrusion detection utilizing neural network classifiers distinct and hybrid approaches, and collection evaluating seven datasets and discussing results for future guidance.

Gulghane et al.[10] suggests a cutting-edge deep learning method to improve IDS performance in the existing system. The effectiveness of the assessed datasets for networked attack detection assessment, specifically KDD Cup 99 datasets the NSL-KDD.

Kim et al.[11] propose Convolutional neural network based model, regularized UTF-8 eccentric programming, and pattern reorganization are used to precisely analyze Long pooling circulation features and malicious probability.

Rai et al.[12] Ensemble learning strategies like DRF, gradient boost, applied with H2O framework using python library, outperform traditional machine approach.

Rahman et al.[13] proposes an effective lightweight intrusion detection system IoT networks, offering competitive detection accuracy with advanced centralized system methods, but balancing accuracy and time performance.

Akter et al.[14] presents a neural network framework for detecting malicious server features, utilizing self-taught techniques and the NSL-KDD benchmark dataset for evaluation.

Ferrang et al.[15] performed a comparative analysis of deep learning techniques for IDS, namely, deep classification models and machine learning models..

Meryem et al.[16] The hybrid machine learning solution successfully reduced error rates and enhanced accuracy in identifying malicious behaviours using rule-based analysis, achieving an average of 99.7% accuracy.

Zhong et al.[17] in comparison to earlier individual learning model approaches, the hierarchal deep learning framework for IDS which uses behavioral and content-related variables to improve the rate of identification of intruding threats.

Dong et al.[18] suggested IDS framework using LSTM model uses comparative test on real world industry UNSW-NB15 dataset illustrates effective performance of suggested LSTM model to verify Intrusions in a network and achieves accuracy 88.11%.The limitation lies that it is not good for all types of attacks.

Kasongo and sum et al.[19] In order to develop an integrated malware detection system for VANETs that are exploit dispersed SDN and combine deeper neural networks with generative adversarial networks to detect attacks. Extensive testing results confirm that using CIDS is reliable and effective for VANET surveillance.. The proposed method achieves accuracy 87.10%.

Kasongo et al.[20] We suggest a type of RNN with transfer learning for IDS that combines the precision of deep learning methods with the benefits of a system with several agents technique. The experiments have shown that model achieves accuracy 88.42%.

Sallam et al.[21] proposed residual learning centered on IDS using dataset like USWNB-15 and achieves accuracy of 93.94%.The limitation of proposed work is cost of high energy consumption resources usage.

Table 1: Comparison analysis of Recent IDS Research

Ref	Technique used	Accuracy	Limitation
Kasim et al.[2] 2020	Deep learning	99.1	Not good for large data
Rani et al.[3] 2020	Supervised ML	99	high computation-cost
Bharti et al.[4] 2020	Machine Learning Based (Random Forest)	99	Requires a lot of training time
Gao et al.[5] 2020	feedforward neural network (FNN)	99.56 and F1 Score 99.68	Unable to handles large network data traffic
Lee et al.[8] 2020	LSTM	99	Requires a lot of training time
Meryem et al.[16] 2020	Support vector machine and deep learning algorithms	99.7	Requires a lot of training time
Dong et al.[18] 2019	LSTM	Accuracy 88.11%	Not good for large data
Kasongo & Sum et al. [19] 2020	Deep Neural Network	Accuracy 87.10%	Not good for all types of attacks
Kansongo et al. [20] 2023	Residual Neural Network	Accuracy 88.42%	Not good for all types of attacks
Sallam et al.[21] 2023	Residual Learning	Accuracy 93.94%	Cost of high energy consumption resources usage and high energy consumption

Mosaiyebzadeh et al.[22] presents a Internet intrusion detection system using deep learning and developed on a publicly accessible set of MQTT assaults, achieving an average accuracy of 97.09% and an F1-score of 98.33%.

Musa et al.[23] explores research articles on particular, mixture, and group ordering processes, compares outcomes metrics, failings, in IDS development, and suggests future research directions.

Rincy et al.[24] introduces an innovative hybrid IDS name NID shield, which classifies datasets based on attack types and individually predicts attack vulnerability. The UNSW-NB15 and NSL-KDD datasets demonstrate a high level and low FPR achieved by the CAPPER technique.

Manhas et al.[25] proposes an Intrusion Detection System (IDS) using machine learning techniques like linear regression, random forest, support vector machine, LSTM to detect malicious network activity.

3. CGAN with ResNet model

The traditional GAN method has a limitation called mode collapse, where it focuses too much on one class instead of representing the entire distribution. This issue arises when the real sample distribution has multiple modes.

To address this problem, we introduced a modified version of the traditional GAN called the conditional generative adversarial network (CGAN). In CGAN, we combine categorical data and noise with actual samples as inputs for both the generator (g) and discriminator (D), using a specific loss strategy. CGAN is successful in learning from the existing distribution samples collectively.Fig.1 shows the architecture of Conditional GAN.

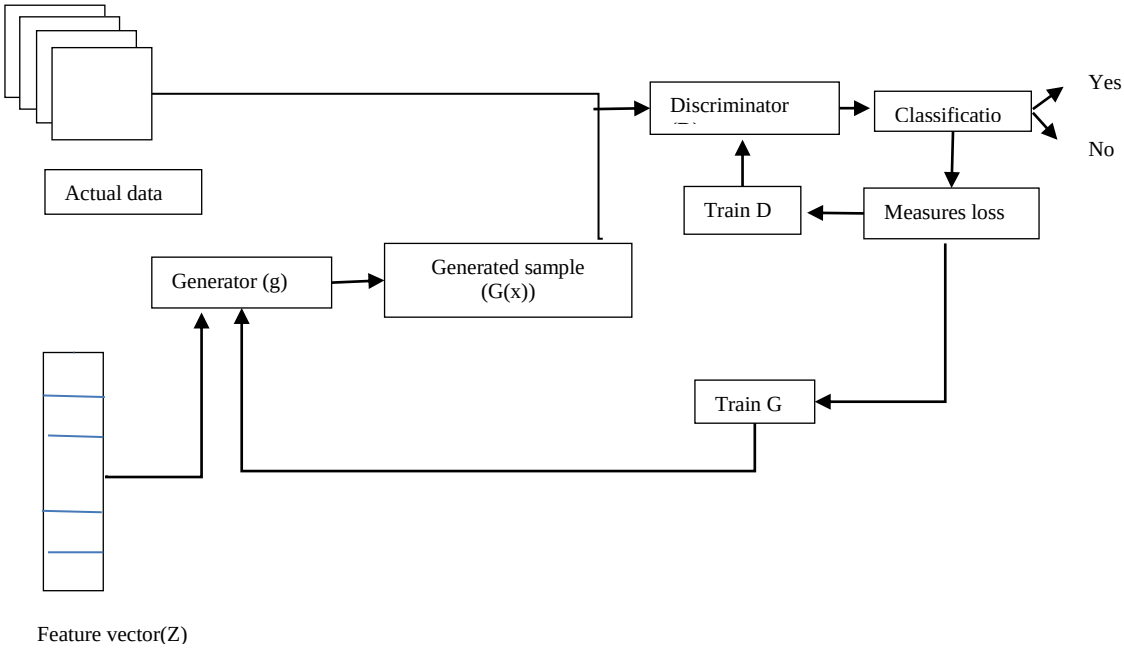


Fig. 1: The architecture of conditional generative adversarial network.

Taking into account its conditioned environment, that system is taught to discriminate properly across created and genuine inputs. The discriminant value can be represented formally as D: {x,c} → Probability of being real. When CGANs are trained, they simultaneously maximize the generator and discriminator, and the resultant desired function is the sum of the losses from the generator and the discriminator, expressed as:

$$L_{gen} = -\log(D(G(z,c),c)) \quad (1)$$

$$L_{disc} = -\log(D(x,c)) - \log(1 - D(G(z,c),c)) \quad (2)$$

$$L_{cGAN} = L_{gen} + L_{disc} \quad (3)$$

Realistic and conditionally accurate generated samples are generated during learning when both discriminator and generator change the settings in opposite ways to establish a Nash equilibrium.

4. Proposed Methodology

In proposed methodology, we adapted CGAN by combining the RestNet and Transfer learning to address the problems in previous research. Incorporate transfer learning by leveraging pre-trained weights from the ResNet backbone. This involves initializing your CGAN with weights from a ResNet model that has been trained on our UNSW-NB 15 dataset. Fine-tune ResNet backbone during the training of our CGAN to adapt it to the mark area. This fine-tuning process helps the network learn domain-specific features.

Generators (g)

The generator in CGAN takes random noise as input (z) along with conditional information (c), such as class labels or attributes.

Discriminator (d)

The discriminator network in the CGAN is responsible for distinguishing between actual descriptions from the dataset generated by the initiator.

ResNet Integration

Modify the generator network to have a ResNet backbone. This ResNet backbone can consist of several residual blocks with skip connections.

Conditional Input

Incorporate the conditional information (c) into both the generator and discriminator networks.

The work flow of the proposed prototypical given in fig. 2.The methodology worked in 2 stages .In stage in the process starts by applying data pre-process in which we extract features, character digitization and data normalization of from dataset. In the next stage the system will classify the intrusion detection by based on RestNet model using sliding window extraction and then classify the attacks.

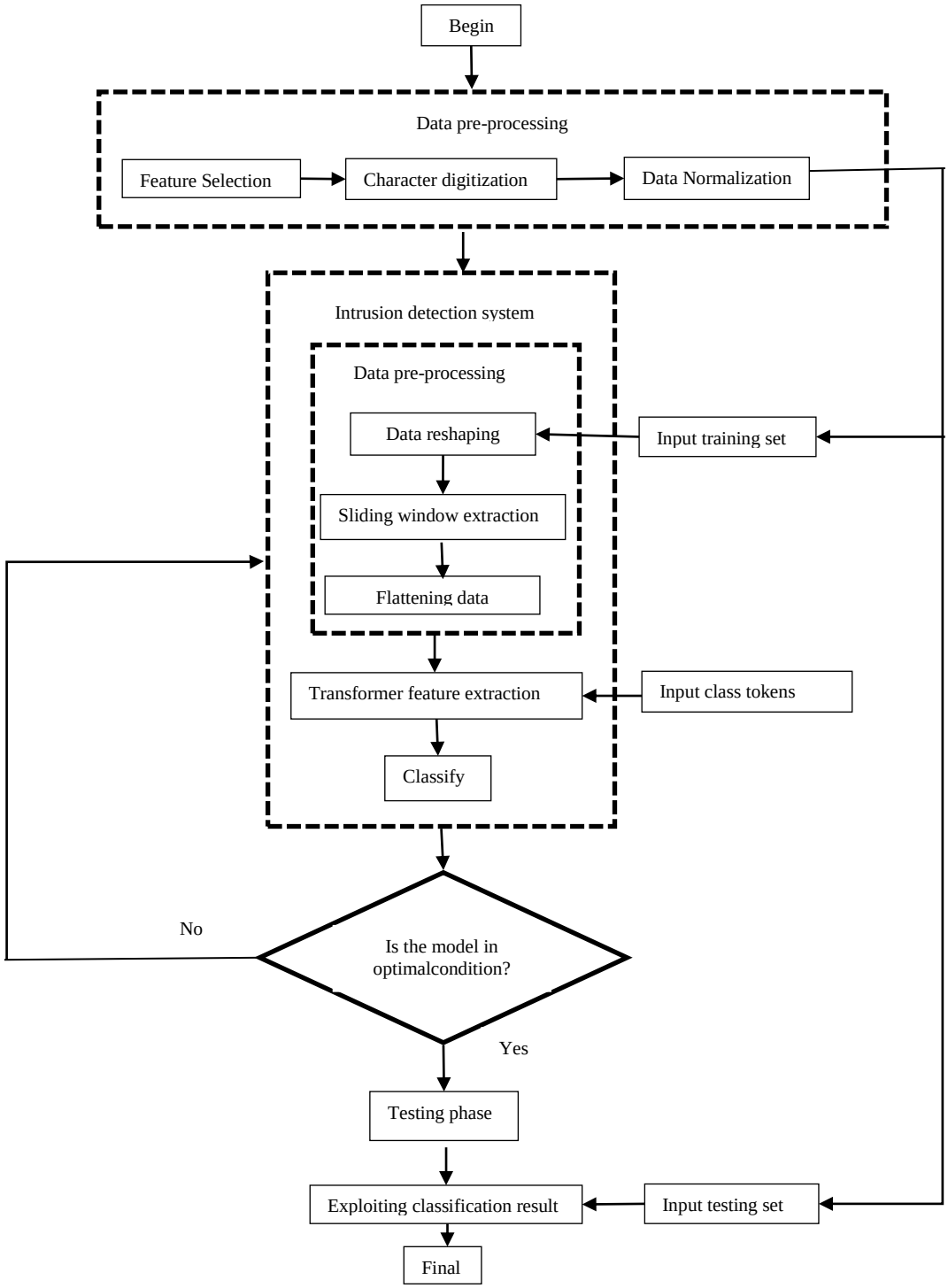


Fig. 2: Flow chart of proposed model

4.1 ResNet based learning

Simplified Residual Network

ResNet is a Convolutional neural network built on a huge scale using residual blocks. Its size is seven times larger than VGG-16 and twenty times larger than AlexNet. Owing to this residual impact, the network's depth can be greater than that of regular networks, hence preventing the deep network's gradient from disappearing and making training more challenging. ResNet's efficiency boosts somewhat rather than decreases as the quantity of cells rises. Figure 5 depicts the residual block's layout.

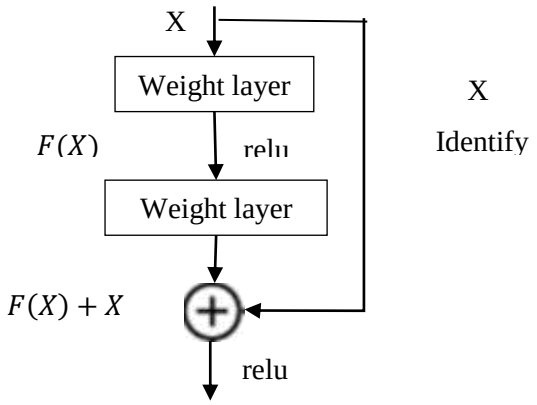


Fig. 3: The structure of residual block.

Figure 3 shows that X represents a residual block's input and F(X) represents the block's result prior to the following activation function. Taken another way, F(X)=W_2 σ(W_1 (X)) where σ represents the rectified linear unit's (ReLU) activation function, W_1 and W_2 signify the first and subsequent layer weights, and σ (F(X)+X) is the residue block's result.

4.2 Vgg16 Architecture

The VGG16 architecture is a type of deep learning model used for image recognition. It's like a highly skilled visual system that can look at an image and tell what's in it.

Working of Vgg16 is given below:

Layers: VGG16 is made up of a series of layers. Think of these layers as a series of steps where each step refines the image a little more.

Convolutional Layers: The first set of layers are convolutional layers. These act like a set of filters that look at small parts of the image to find patterns, like edges or textures. VGG16 has 13 of these layers.

ReLU Activation: After each convolutional layer, there’s a ReLU activation function. This is like a decision-maker that keeps only the important information and discards the rest.

Pooling Layers: These layers come after some of the convolutional layers and are like zooming out a bit on the image to see the bigger picture. They decrease the dimensions of the pixel of an image but keep the essential material. VGG16 has 5 pooling layers.

Fully Connected Layers: Near the end, there are fully connected layers. Imagine these layers as a complex decision-making system that uses all the information gathered to determine what the image is. VGG16 has 3 of these layers.

Output Layer: The final layer serves as an output layer gives the final decision on what the image contains.VGG16 is often used as a pretrained model, meaning it has previously been proficient on a large dataset of images (like the ImageNet dataset), so it has learned a lot about recognizing different objects. Below fig. 4 shows architecture of Vgg16.

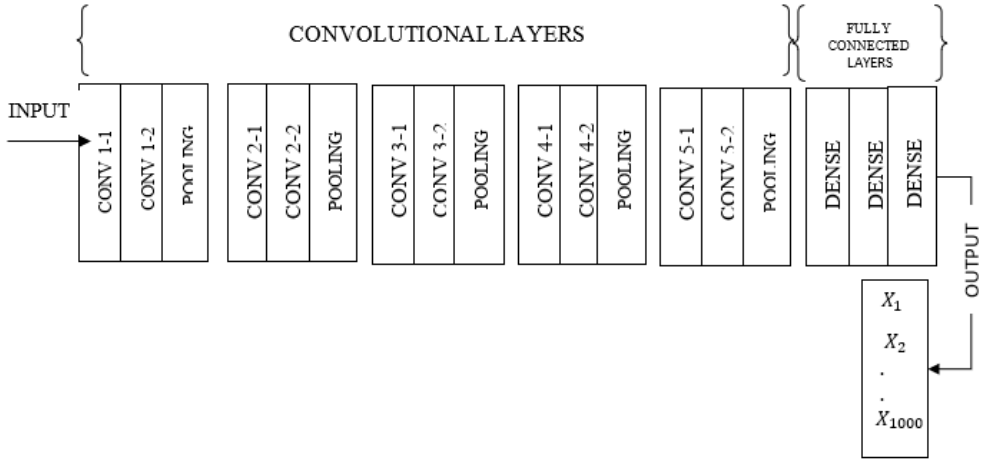


Fig. 4: Arcitecture of Vgg16

4.3 Vgg19 Architecture

VGG19 is characterized by its deep and uniform architecture, with minor 3x3 conv kernels and max-pooling sheets interspersed to decrease dimensionality. The uniformity and simplicity of VGG19 make it a powerful and easy-to-implement architecture for image classification tasks. Despite its depth, the use of small filters allows it to capture intricate details in images while maintaining manageable computational complexity. This architecture has been highly influential in the development of more advanced neural networks and continues to be used as a baseline in many computer vision applications. Fig. 5 shows the proposed architecture of vgg19.

It is one of the most influential architectures in the field of computer vision and is identified for its effortlessness and depth, assembly it effective for a wide variety of image classification tasks.

The VGG19 network uses an image with a fixed size of 224 x 224 x 3 as input. It consists of 16 convolutional layers, each with a series of covx sheet and a max-pooling layer. The system uses small receptive fields and the ReLu activation function for non-linearity. A max-pooling layer reduces spatial dimensions and provides translation invariance. The final layer compress three densely connected layers each with 4096 networks, for high-level reasoning and classification.

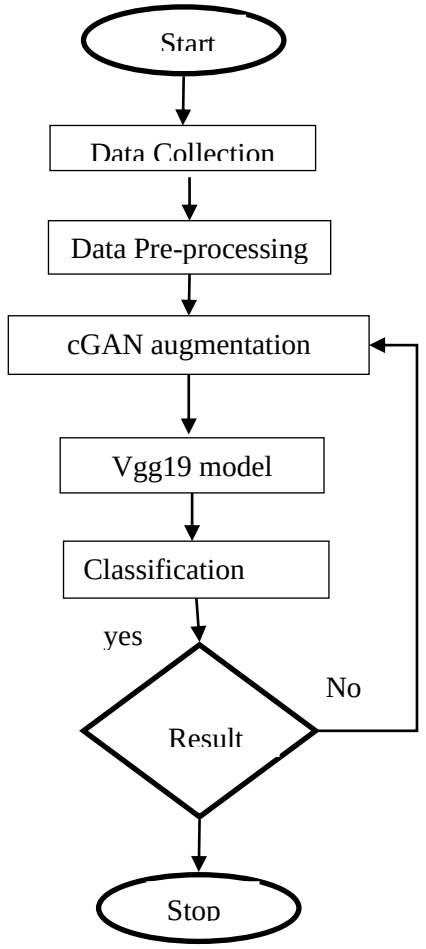


Fig. 5: Flow chart of Proposed Vgg19 model

5. Result and Discussion

In this study, we assessed the performance using the UNSW-NB15 dataset. This dataset was created by the Cyber Range Lab and includes both normal network behaviour and anomalous packets. The dataset comprises 100GB of network traffic captured in Pcap files, containing nine attack types along with normal network packets. To analyze this data, we extracted 49 features labelled by class using tools like Argus and Bro-IDS.

5.1 Performance Evaluation Measures

To evaluate the performance, following parameters are used:

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (4)$$

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{5}$$

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \tag{6}$$

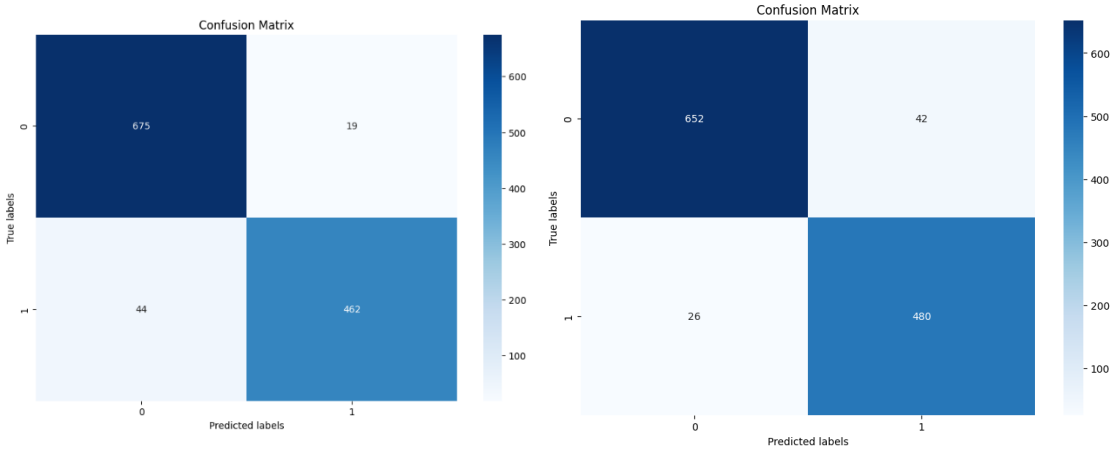
$$\text{F1} - \text{Score} = \frac{2 * \text{Precision} * \text{Recall}}{(\text{Precision} + \text{Recall})} \tag{7}$$

5.2 Result Analysis

Table 2. Performance Evaluation of Learning Models

	Vgg16			Vgg19		
Category	"Precision"	"Recall"	"F1-Score"	"Precision"	"Recall"	"F1-Score"
Attack	93.9	97.3	95.5	96.2	93.9	95
Normal	96	91.3	93.6	92	94.9	93.3

Table 2 compares the performance of two learning models i.e., Vgg16 and Vgg19 using metrics for the "Attack" and "Normal" categories. In the "Attack" category, Vgg19 has slightly greater accuracy, although Vgg16 shows better recall and has high F1-score. This indicates overall ability to detect actual attack situations. Vgg16 has higher precision in the "Normal" category whereas Vgg19 outperforms better in recall and their F1-scores are closely identical. Overall both models perform well. Vgg16 shows better recall and Vgg19 showing advantages in accuracy and infers that the decision between them is based on whether better precision or recall is more desired for the application's needs. Fig. 6 presents the confusion matrix for both models.



(a) VGG16 (b) Vgg19

Fig. 6: Confusion Matrix

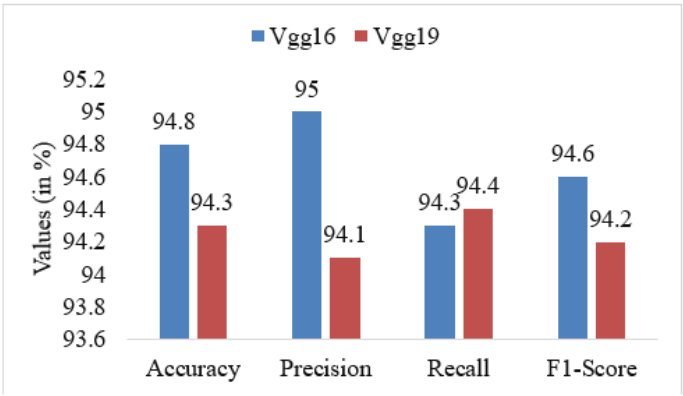


Fig. 7: Comparison of Learning Models

The fig. 7 shows the performance of the Vgg16 and Vgg19 models and following observations are inferred:

- Vgg16 gets a greater accuracy of 94.8% than Vgg19 (94.3%).
- Vgg16 also has a greater precision of 95% compared to 94.1% for Vgg19.
- The recall rates are nearly identical, with Vgg16 at 94.3% and Vgg19 slightly higher at 94.4%.
- Vgg16 once again leads with 94.6%, while Vgg19 has 94.2%.

Overall, Vgg16 performs somewhat better indicating that it may be the more effective model overall, particularly in cases where precision and overall accuracy are critical. Vgg19, however, has a comparable performance, particularly in recall, indicating that it is almost as robust.

Table 3. Comparative State-of-art for Binary Classification

	Dataset	Data Imbalance Handling	Learning	Accuracy
Dong et al. (2019)	UNSW-NB15	-	LSTM	88.11%
Kasongo and Sun (2020)	UNSW-NB15	-	DNN	87.10%
Kasongo (2023)	UNSW-NB15	-	Recurrent Neural Networks	88.42%
Sallam et al. (2023)	UNSW-NB15	-	Residual Learning	93.94%
Proposed	UNSW-NB15	cGAN	Vgg16	95%

Table 3 reviews several investigations utilizing the data set from UNSW for classification in binary, highlighting various techniques and their outcomes. Dong et al. [18] achieved a preciseness of 88.11% by using LSTM for training and Information Gain for feature selection without addressing data imbalance. Kasongo and Sun [19] used Extra Trees and DNN, reaching 87.10% accuracy, also without data imbalance strategies. Kasongo [20] separately applied XGBoost and Recurrent Neural Networks, slightly improving accuracy to 88.42%. Sallam et al. [21] achieved a notable accuracy of 93.94% using Residual Learning without certain techniques for choice of features or imbalanced data. The proposed

methodology in the study stands out by addressing data imbalance with CGANs and employing VGG16 for learning, achieving the highest accuracy of 95%. This suggests that addressing data imbalance significantly enhances model performance.

6. Conclusion

This work proposed IDS centered on vgg16 and vgg19 model to classify attacks on binary classifiers. To handle data imbalancing we have used conditional Generative Adversarial Networks by learning the mode using ResNet. The proposed method is distinguished for tackling class imbalancing by using CGANs and utilizing VGG16 for learning. This approach achieved the highest accuracy of 95%, indicating that addressing data imbalance can greatly improve model performance. This study will be expanded later on to include more datasets as well as further sophisticated methods for handling data imbalances caused by minority assaults.

References:

[1] S. Gamage and J. Samarabandu, "Deep learning methods in network intrusi detection: A survey and an objective comparison," J. Netw. Comput. Appl., vol. 169,p. 102767, 2020, doi: https://doi.org/10.1016/j.jnca.2020.102767.

[2] Ö. KASIM, "An efficient and robust deep learning based network anomaly detection against distributed denial of service attacks," Comput. Networks, vol. 180, p. 107390, 2020, doi: https://doi.org/10.1016/j.comnet.2020.107390.

[3] D. Rani and N. C. Kaushal, "Supervised Machine Learning Based Network Intrusion Detection System for Internet of Things," in 2020 11th International Conference onComputing, Communication and Networking Technologies (ICCCNT), 2020, pp. 1–7. doi: 10.1109/ICCCNT49239.2020.9225340.

[4] M. P. Bharati and S. Tamane, "NIDS-Network Intrusion Detection System Based on Deep and Machine Learning Frameworks with CICIDS2018 using Cloud Computing,"in 2020 International Conference on Smart Innovations in Design, Environment,Management, Planning and Computing (ICSIDEMPC), 2020, pp. 27–30. doi:10.1109/ICSIDEMPC49020.2020.9299584.

[5] J. Gao et al., "Omni SCADA Intrusion Detection Using Deep Learning Algorithms,"IEEE Internet Things J., vol. 8, no. 2, pp. 951–961, 2021, doi:10.1109/JIOT.2020.3009180.

[6] M. A. Alsoufi, S. Razak, M. M. Siraj, A. Ali, M. Nasser, and S. Abdo, "Anomaly Intrusion Detection Systems in IoT Using Deep Learning Techniques: A Survey," in Innovative Systems for Intelligent Health Informatics, 2021, pp. 659–675.

[7] S. Al-Emadi, A. Al-Mohannadi, and F. Al-Senaid, "Using Deep Learning Techniques for Network Intrusion Detection," in 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), 2020, pp. 171–176. doi:10.1109/ICIoT48696.2020.9089524.

[8] T.-H. Lee, L.-H. Chang, and C.-W. Syu, "Deep Learning Enabled Intrusion Detection and Prevention System over SDN Networks," in 2020 IEEE International Conference on Communications Workshops (ICC Workshops), 2020, pp. 1–6. doi: 10.1109/ICCWorkshops49005.2020.9145085.

[9] U. S. Musa, M. Chhabra, A. Ali, and M. Kaur, "Intrusion Detection System usingMachine Learning Techniques: A Review," in 2020 International Conference on SmartElectronics and Communication (ICOSEC), 2020, pp. 149–155. doi:10.1109/ICOSEC49089.2020.9215333.

[10] S. Gulghane, V. Shingate, S. Bondgulwar, G. Awari, and P. Sagar, "A Survey on Intrusion Detection System Using Machine Learning Algorithms," in Innovative Data Communication

Technologies and Application, 2020, pp. 670–675.
[11] A. Kim, M. Park, and D. H. Lee, "AI-IDS: Application of Deep Learning to Real-Time Web Intrusion Detection," IEEE Access, vol. 8, pp. 70245–70261, 2020, doi:10.1109/ACCESS.2020.2986882.
[12] A. Rai, "Optimizing a New Intrusion Detection System Using Ensemble Methods and Deep Neural Network," in 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), 2020, pp. 527–532. doi:10.1109/ICOEI48184.2020.9143028.
[13] M. A. Rahman, T. Asyhari, L. S. Leong, G. Satrya, M. Tao, and M. Zolkipli, "Scalable Machine Learning-Based Intrusion Detection System for IoT-Enabled Smart Cities,"Sustain. Cities Soc., vol. 61, p. 102324, 2020, doi: 10.1016/j.scs.2020.102324.
[14] M. Akter, G. Das Dip, M. S. Mira, M. Abdul Hamid, and M. F. Mridha, "Construing Attacks of Internet of Things (IoT) and A Prehensile Intrusion Detection System for Anomaly Detection Using Deep Learning Approach," in International Conference on Innovative Computing and Communications, 2020, pp. 427–438.
[15] 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., vol. 50, p. 102419, 2020, doi:https://doi.org/10.1016/j.jisa.2019.102419.[16] A. Meryem and B. E. L. Ouahidi, "Hybrid intrusion detection system using machine learning," Netw. Secur., vol. 2020, no. 5, pp. 8–19, 2020, doi:https://doi.org/10.1016/S1353-4858(20)30056-8.
[17] W. Zhong, N. Yu, and C. Ai, "Applying big data based deep learning system to intrusion detection," Big Data Min. Anal., vol. 3, no. 3, pp. 181–195, 2020, doi:10.26599/BDMA.2020.9020003.
[18] Dong, Y. Wu, J. Song, R. Lu, T. Li, and L. Zhao, "DeepFed: Federated Deep Learning for Intrusion Detection in Industrial Cyber–Physical Systems," IEEE Trans. Ind.Informatics, vol. 17, no. 8, pp. 5615–5624, 2019, doi: 10.1109/TII.2020.3023430.
[19] Kasongo and Sum, L. Zhou, W. Zhang, X. Du, and M. Guizani, "Collaborative Intrusion Detection for VANETs: A Deep Learning-Based Distributed SDN Approach," IEEE Trans. Intell. Transp. Syst., vol. 22, no. 7, pp. 4519–4530, 2020, doi:10.1109/TITS.2020.3027390.
[20] Kansongo and F. B. Ktata, "A deep learning-based multi-agent system for intrusion detection," SN Appl. Sci., vol. 2, no. 4, p. 675, 2023, doi: 10.1007/s42452-020-2414-z.
[21] Sallam, A. Rajendran, D. Pelusi, and V. Ponnusamy, "Deep Belief Network enhanced intrusion detection system to prevent security breach in the Internet of Things,"Internet of Things, vol. 14, p. 100112, 2021, doi: https://doi.org/10.1016/j.iot.2019.100112.
[22] F. Mosaiyebzadeh, L. G. Araujo Rodriguez, D. Macêdo Batista, and R. Hirata, "A Network Intrusion Detection System using Deep Learning against MQTT Attacks in IoT," in 2021 IEEE Latin-American Conference on Communications (LATINCOM), 2021, pp. 1–6. doi: 10.1109/LATINCOM53176.2021.9647850.
[23] U. S. Musa, S. Chakraborty, M. M. Abdullahi, and T. Maini, "A Review on Intrusion Detection System using Machine Learning Techniques," in 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), 2021,pp. 541–549. doi: 10.1109/ICCCIS51004.2021.9397121.
[24] T. Rincy N and R. Gupta, "Design and Development of an Efficient Network Intrusion Detection System Using Machine Learning Techniques," Wirel. Commun. Mob. Comput., vol. 2021, p. 9974270, 2021, doi: 10.1155/2021/9974270.
[25] J. Manhas and S. Kotwal, "Implementation of Intrusion Detection System for Internet of Things Using Machine Learning Techniques," in Multimedia Security: Algorithm Development, Analysis and Applications, K. J. Giri, S. A. Parah, R. Bashir, and K.Muhammad, Eds. Singapore: Springer Singapore, 2021, pp. 217–237. doi:10.1007/978-981-15-8711-5_11.