

ATLIDN Deep Learning for DDoS Intrusion Detection in Cloud Environments

Posted on October 2, 2026

Abstract

The Distributed Denial of Service (DDoS) Intrusion Detection System (IDS) for cloud environments uses [deep learning](#) (DL) to detect intrusions. Advanced DL techniques are used by this system to automatically identify and counteract these malicious attacks that have the potential to interfere with cloud services. The IDS aims to effectively distinguish between typical and abnormal network traffic patterns linked to DDoS attacks by utilising the strength of DL algorithms. By offering an intelligent and automated defence mechanism against DDoS attacks, this research helps to get better the safeguard and dependability of cloud services, ultimately ensuring the efficient operation of cloud-based applications and services. Accumulate labelled IoT network traffic data, capturing both typical and attack situations. Cleanse and preprocessing the pre-processed data to remove noise and anomalies. Next, normalise the data to ascertain that features have comparable scales, promoting convergence during training. In order to extract the features, utilise DPI techniques to extract important data from network packets, such as IP addresses for the source and destination, packet size, and payload information. To identify temporal dependencies in network traffic, use time-series analysis. Use transformers or recurrent neural networks (RNNs) to model the flow of packets and their interdependencies over time. Pearson correlation coefficients are taken into account when choosing the features. To lessen redundancy, eliminate features with high correlation coefficients. It computes Mutual Information Computation. Choose features with high mutual information scores, which show their significance in class distinction. RFE should be applied to the remaining features to progressively remove less crucial features. To rank features according to their importance and remove the ones that aren't as important, use the classifier Random Forest. The chosen features are then presented. It is suggested to classify data using the "Attention-enhanced Transfer Learning Intrusion Detection Network" (ATLIDN). Create the ATLIDN architecture using the VGG16 and ResNet50 base models. Use a self-attention system to balance the contributions of the ResNet50 and VGG16 pathways in real time. the pre-trained ResNet50 and VGG16 models for intrusion detection should be fine-tuned. PYTHON is used for the implementation.

Keywords: *ATLIDN Architecture, Deep Learning, Intrusion Detection System, DDOS Attacks, Cloud Environment.*

Nomenclature

Abbreviation	Description
ANN	Artificial Neural Network
APO	Artificial Plant Optimization
BPNN	Back Propagation Neural Network
CC	Cloud computing
CNN	Convolutional Neural Network
DDoS	Distributed Denial of Service
DL	Deep Learning
DMN	Deep Maxout Network
DPDK	Data Plane Development Kit
DPI	Deep Packet Inspection
DSA	Deep stacked autoencoder
FC	Fully Connected
FCM	Fuzzy C Means
FPR	False Positive Rate
GHLBO	Gradient hybrid leader optimization

HLBO	Hybrid leader-based optimization
IDS	Intrusion Detection Systems
LSTM	Long Short-Term Memory
MCC	Mathews Correlation Coefficient
ML	Machine Learning
NPV	Normal Predictive Value
RBM	Restricted Boltzmann Machines
RF	Random Forest
RFE	Recursive Feature Elimination
RHS	Random Harmony Search
SDN	Software Defined Network
SVM	Support Vector Machine

Introduction

A computational model called CC enables claim network entrance to a shared pool of adaptable computing resources with a minimum of service provider involvement. CC has enjoyed tremendous popularity and acceptance over the past ten years. Numerous enticing CC features, such as on-demand self-service, resource pooling, rapid elasticity, pay-as-you-go pricing, etc., have drawn gradually businesses to participate in CC's accomplishment by affecting their data and uses to the cloud [1]. Due to of the CC's fundamental features, DDoS attacks against it are also increasing [2]. DDoS is a form of attack that uses distributed network resources on the Internet to disrupt network approachability and it is derived from DoS attack. DDoS uses a variety of techniques to stop the victim system from offering regular services. The Dos attack did not initially garner enough attention because it was launched by a single machine and was not destructive. However, as the Internet

expanded and network resources were continuously improved, DDoS started to use distributed attacks to continuously increase the influence of DoS attacks [3].

CC market has undergone an unprecedented development over the past five years as a result of the significant economic benefits. With an annual market growth rate of 24 percent, the global CC market now generates \$180 billion in revenue. However, the widespread use of CC also left the cloud systems extremely open to various kinds of [cyber-attacks](#). Therefore, both service providers and end users are becoming increasingly concerned about the security of the cloud environment. IDSs are one of many strategies for safeguarding cloud systems and are essential for quickly identifying and thwarting security attacks [4]. IDS was being used to keep track of a network's traffic in order to look for malicious activity [5]. If a malicious attack was discovered while the traffic was being monitored, the system administrator was immediately notified. IDSs have the ability to stop traffic coming from a system that is infected with a virus and to take action against such malignant action. There exist two different kinds of IDS: anomaly-based and signature-based systems. [6, 7]. IDS [8] has the highest priority when it comes to protecting users' cloud-based data and keeping consumers' trust [9].

DL [10] has demonstrated superior accuracy compared to other ML techniques on large datasets [11]. Large quantities of unsupervised data are used to extract intricate abstractions of important data descriptions. Manual feature engineering is replaced by hierarchical feature extraction and unsupervised or semi-supervised feature learning in DL. Due to DL's many benefits, it has been used in numerous studies to enhance category execution, involving the discovery and categorization of attacks opposed IDSs. [12, 13].

The foremost contribution of the research is as follows: –

To solve the important problem of intrusion detection in Cloud IoT devices by a novel combination of DL, transfer learning, and attention mechanisms.

To create an IDS that is reliable and effectual and is suited for cloud environments.

To gradually remove less crucial features, perform RFE on the remaining features.

This research report is structured as follows. Section II gives an outline of related work on DL-Based IDS for DDoS Attacks in Cloud Environment. Section III explains the Proposed Methodology. Section IV presents information about datasets and evaluation parameters. Section V presents our conclusions.

Literature Review

In 2021, Liu *et al.* [14] identified an IDS that combined DL and ML. To quickly classify typical events from attack events, the model uses the k-means and RF algorithms for binary classification. Distributed computing of these algorithms was carried out on the Spark platform. The events deemed abnormal were further classified into various attack types using DL algorithms such as CNN, LSTM and

others.

In 2020, Samriya *et al.* [15] have enhanced the general security of the CC environment by a new hybridization approach for the IDS. Additionally, this method aids in addressing a variety of cloud security challenges to support cloud security. In contrast to fuzzy based clustering, which was optimised using the spider-monkey optimisation algorithm, the method uses ANN with fuzzy bases for effective clustering of anomalies. The fuzzy clustering approach's iterative classification and selection process is bypassed by this hybridization method by inevitably modernization the fitness value.

In 2021, Mayuranathan *et al.* [16] have used IDS as a security measure that works at the network layer. Traditional IDS on cloud platforms has a low detection accuracy and a complex computational structure. Then presented an efficacious feature subset selection-based sorting model for the detection of DDoS attack while keeping these problems in mind. The best feature sets were chosen with limit discovery using the RHS optimisation model in order to detect DDoS attacks in IDS. Followed the selection of the features, a DL-based classifier model employing RBM was used to identify the DDoS.

In 2023, Balasubramaniam *et al.* [17] proposed GHLBO algorithm it makes DDoS attack detection simple to identify in an efficient manner. The goal of this improved algorithm was to efficiently train a DSA to recognise an attack. Here, oversampling was used to augment the data while DMN with an overlap coefficient was used to combine the features. Additionally, by combining the HLBO and gradient descent algorithms, the proposed GHLBO was created.

In 2022, Priyadarshini & Barik *et al.* [18] have suggested a new source-based DDoS defence technique that applied to equally cloud & fog environments to lessen DDoS attacks. To detect the unusual behaviour of DDoS attacks at the Network/Transport level, it deploys the DDoS defender module at the SDN controller using SDN. The suggested work offered a DL-based detection technique that used network traffic analysis mechanisms to filter and forward legitimate packets to the server while blocking infected packets that could otherwise lead to additional attacks.

In 2022, Arunadevi & Sathya *et al.* [19] suggested the BPNN to recognise DDoS attacks in a cloud environment. APO algorithm was used to optimise the weights and bias of the connections in the proposed optimised BPNN. Utilising four datasets, the proposed APO-BPNN detection system was assessed.

In 2020, Jaber & Rehman *et al.* [20] suggested a novel IDS that combined an FCM algorithm with SVM. Implemented and contrasted with current systems was the suggested system. Experiments make use of the NSLKDD dataset. Based on performance evaluation and comparative analysis, the outcomes obtained with this new hybrid mechanism demonstrate that, compared to existing techniques, the proposed system could detect anomalies with a high detection accuracy and a low false alarm rate.

In 2020, Erhan & Anarim *et al.* [21] suggested novel DDoS detection framework based on the Matching Pursuit algorithm. Then simultaneously employed a number of network traffic characteristics

to effectively identify low-density DDoS attacks. The suggested approach makes use of a dictionary created from network traffic parameters using the KSVD algorithm. Dictionary creation using network traffic offers models for legitimate and malicious traffic, and increases the method's adaptability to network traffic.

In 2021, Varghese & Muniyal *et al.* [22] have suggested a novel framework to incorporate intelligence in the data layer using DPDK in the SDN architecture to address the performance issues of IDS and the design issues of SDN about DDoS attacks. This innovative framework was known as a DPDK-based DDoS Detection framework because DPDK offers quick packet processing and data plane monitoring. Additionally, the fast detection of DDoS attacks was provided by the statistical anomaly detection algorithm. The novel IDS framework's effectiveness and efficiency were guaranteed by the experimental D3 framework results. Table 1 shows Various authors reviews.

Research Gaps

Table 1: Research Gaps

Author	Technique/ Process	Research Gaps
Tama <i>et al.</i> [8]	An entirely novel technique for anomaly-based IDS is proposed, combining hybrid feature selection with a two-stage meta classifier.	In order to solve a multi-class classification problem using the proposed approach, which classifies incoming network traffic as either normal or some attack groups, it must be validated.
Singh & Ranga <i>et al.</i> [9]	an efficient ensemble-based ML approach using four classifiers for a network-based intrusion detection model	On encrypted network packets, didn't used the proposed IDS.
Balasubramaniam <i>et al.</i> [17]	GHLBO	The strategy is not incorporated advanced optimisation techniques for improved performance, and more performance metrics is not taken into account when evaluating performance.

Proposed Methodology

In order to improve the security of cloud devices, this research aims to create an advanced IDS for DDoS attacks that makes use of DL and transfer learning techniques along with attention mechanisms.

Step 1: Data Collection

Gather labelled IoT network traffic data, capturing both normal and attack scenarios.

Step 2: Pre-processing

Data Cleaning: Clean and preprocess the data to remove noise and anomalies.

Normalization: Normalize the data to ensure that features have similar scales, aiding convergence during training.

Step 3: Feature Extraction

DPI: Apply DPI techniques to extract relevant information from network packets, such as payload content, packet size, protocol information, and source/destination IP addresses.

Time-Series Analysis: Utilize time-series analysis to capture temporal dependencies in network traffic.

Sequence Modelling: Use RNN or transformers to model the sequence of packets and their dependencies over time.

Step 4: Feature Selection

Correlation Analysis: Calculate the Pearson correlation coefficients between features. Remove features with high correlation coefficients (above a specified threshold) to reduce redundancy.

Mutual Information Computation: Compute mutual information between each feature and the target labels (normal or attack). Select features with high mutual information scores, indicating their importance in distinguishing between classes.

RFE: Perform RFE on the remaining features to iteratively eliminate less important features. Use a classifier – Random Forest) to rank features based on their importance and eliminate the least important ones.

Step 5: Classification via "Attention-enhanced Transfer Learning Intrusion Detection Network" (ATLIDN)

Design the ATLIDN architecture that incorporates both ResNet50 and VGG16 base models.

Implement a self-attention mechanism to dynamically weigh the contributions of ResNet50 and VGG16 pathways.

Fine-tune the pre-trained ResNet50 and VGG16 models for intrusion detection. Figure 1 shows the view of the proposed work.

Input Layer: Accepts the selected features as input.

Feature Encoding Layer: FC (Dense) Layer: Processes input features and converts them into a more abstract representation.

Attention Mechanism Layer: Self-Attention Mechanism: Captures feature interactions and focuses on relevant patterns across different pathways.

Transfer Learning with Pre-trained ResNet50 and VGG16 Bases:

Pre-trained ResNet50 Base: Import a pre-trained ResNet50 model

Pre-trained VGG16 Base: Import a pre-trained VGG16 model

Classification Layer:

Combine Outputs: Combine the outputs of the ResNet50 and VGG16 pathways using attention-based weights.

Fully Connected (Dense) Layers: Process the combined output.

Softmax Activation: Provides probability distribution over classes (normal or attack).

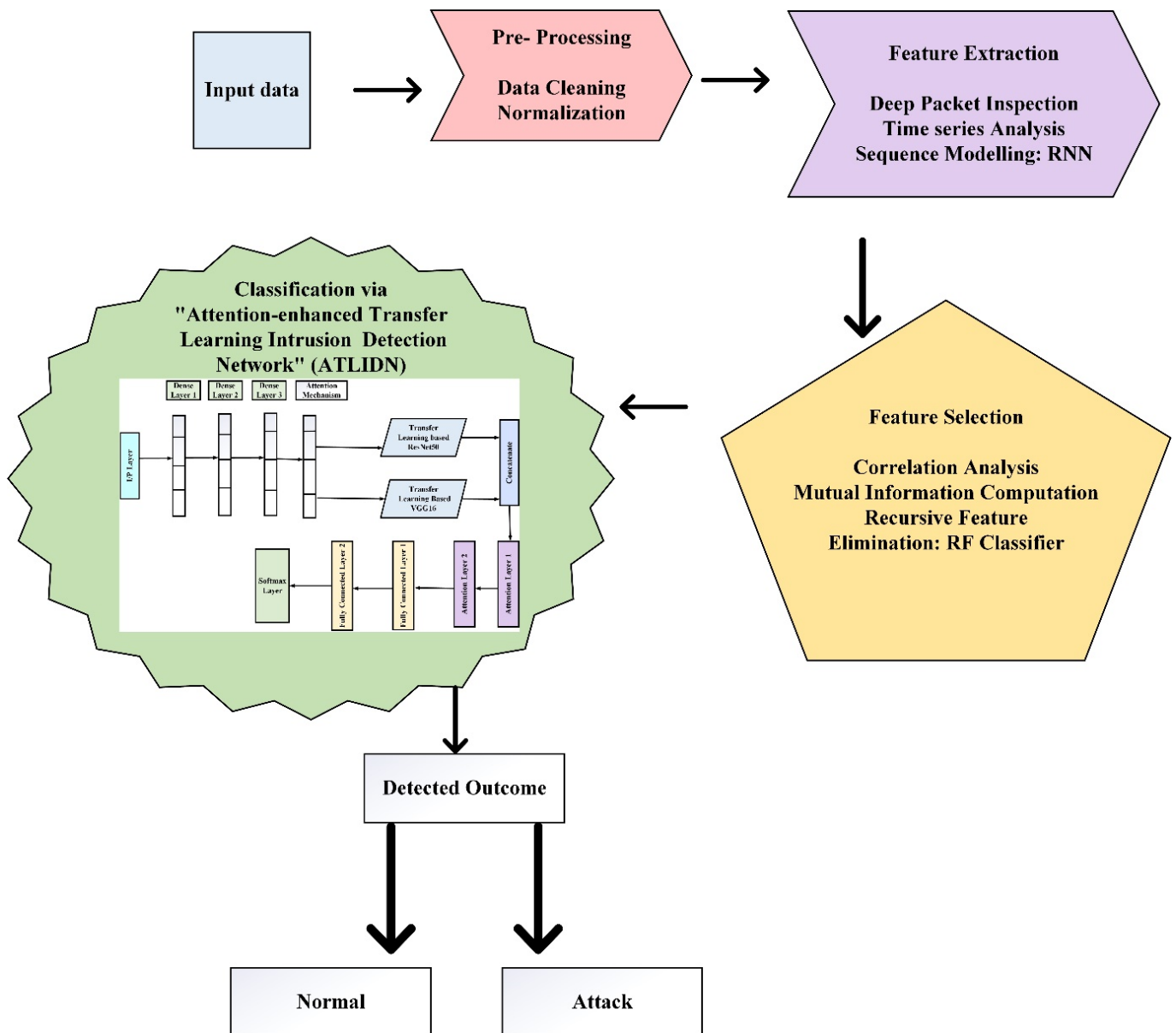


Figure 1: Overall Flow of the paper

3.1. Data Collection

Labelled IoT network traffic data collection involves recording a wide variety of events, such as typical good-natured activities and potential attack scenarios. To protect IoT ecosystems from new cyber threats, this dataset would be a valuable asset for developing and improving robust IDS.

3.2. Pre-processing

In this research work, pre-processing is processed using Data Cleaning and Normalization. Figure 2 shows the flow of Pre-processing.

Figure 2: Pre-processing phase

3.2.1. Data Cleaning

The procedure of fixing or removing data that is duplicated, corrupted, improperly formatted, or incomplete within a dataset is referred to as data scrubbing or data cleaning.

3.2.1. Normalization

The goal of data normalisation is to reduce or even eliminate duplicate data. One of the popular techniques for normalising the input features or variables is the Mix-Max scaler. All features are converted to a range between 0 and 1. Used a normalisation approach to reduce bias brought on by features that were measured at various scales and did not equally contribute to model fitting. Adopted feature-wise normalisation, such as Min-Max scaling, in order to standardise feature vectors, as shown in Eq. (1).

(1)

3.3. Feature Extraction

In this research work, DPI, Time series Analysis, Sequence Modelling is considered for the feature extraction. Figure 3 shows the phase of feature extraction.

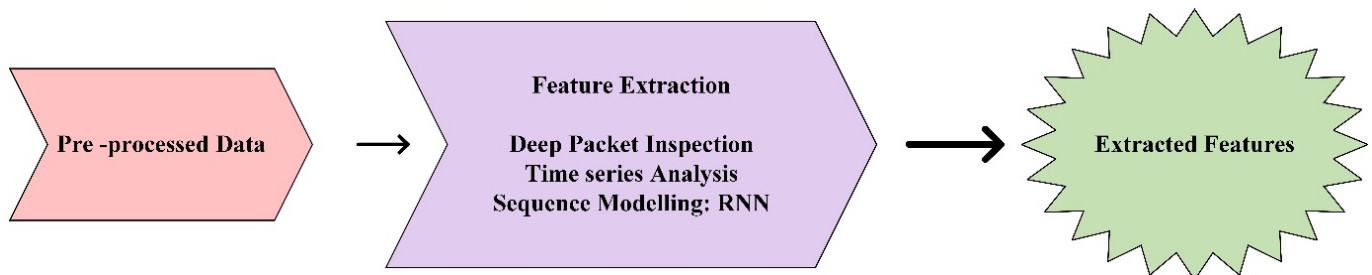


Figure 3: Feature Extraction Phase

3.3.1. DPI

DPI is based on pattern matching, which makes it possible to search for particular content within network traffic. DPI analyses packets using criteria set by network administrators or ISPs. DPI, in contrast to traditional packet filtering, keeps track of these packets' contents while also locating their source and identifying the service or application that sent them. It then makes a decision regarding how to handle this traffic in accordance with the established rules and uses filters to reroute non-business-critical traffic coming from particular websites or IP addresses. To deal with the growth in volume and complexity of data as well as security threats or anomalies, DPI software is used in

conjunction with firewalls and IDS as an addition to traditional security systems.

Payload Content: the actual information contained in a network packet, such as the text of an email or a file, or the HTML code of a website.

Packet Size: The amount of data contained in a network packet in bytes, which is crucial for efficiency analysis and anomaly detection.

Protocol Information and Source/Destination IP Addresses: IP addresses, which identify the sender and recipient locations and are essential for routing and security analysis in networking, specify the type of communication (e.g., TCP for reliable data, UDP for faster transmission).

3.3.2. Time-Series Analysis

Network traffic time-series analysis involves tracking data over time to spot patterns, trends, and anomalies. Temporal dependencies, such as regular communication patterns, peak usage periods, and sudden spikes in activity, it is found by analysing sequential network packet data, assisting in network optimisation, intrusion detection, and performance monitoring. This method enables proactive response to shifting conditions and aids in understanding how network behaviour changes over time.

3.3.3. Sequence Modelling

Sequence models are computer learning algorithms that input or output data sequences. Sequential data takes many different forms, including text streams, audio and video clips, time-series data, and others. A popular technique in sequence models is RNNs.

3.3.3.1. Recurrent Neural Network

Neural networks are extended to sequential data by recurrent neural networks (RNNs). They generate a series of hidden states from an input vector sequence that are calculated at time step as follows. The mathematical model is shown in Eq. (2).

(2)

is a recurrent weight matrix, is input-to-hidden weight matrix & is an arbitrary activation function. Bidirectional RNNs are possible if have access to the entire input sequence due to the use data from both the past and future time steps. The mathematical model is shown in Eq. (3) – Eq. (5).

(3)

(4)

(5)

where denotes the concatenation of and . A deeper architecture is created by stacking RNNs by using as the input to another RNN. The mathematical model is shown in Eq. (6).

(6)

A sigmoid function, like the hyperbolic tangent, serves as the activation function in standard RNNs. Due to the vanishing and exploding gradients, training such networks is known to be particularly challenging.

3.4. Feature Selection

3.4.1. Pearson correlation Analysis

It serves as a gauge for how closely two things would correlate. Using recommendation systems is a common strategy for discovering connections between things. The Pearson formula is used to calculate the correlation between two items. The Pearson formula is given in Eq. (7).

(7)

If means a strong +ive correlation, null correlation, is a strong -ive correlation.

3.4.2. Mutual Information Computation

The MI describes the transfer of a significant number of valuable details from the provided input source images to the final fused image. The mathematical model is shown in Eq. (8).

(8)

is the fused image and and describe two source input images. and stand for the input reference and fused output image's joint histograms. Whereas, , , and represents the matching histograms of , , and .

3.4.3. RFE

A feature selection technique to find the vital features in a data archive is called RFE. Once the desired number of features is obtained, the process involves repeatedly removing the least important components and creating a model with the remaining features.

3.4.3.1. Random Forest

It is a ML classifier as stated in supervised techniques that creates and constructs models using Decision Tree Classifiers. In general, trees pick up abnormal behaviour and overfit the trained model with bias and minor differences. It is applied to lower feature variance in a given dataset. Additionally, it aids in classifying the identical training and test datasets, emerging at the price of a slight bias increase. This approach is used by many businesses, including banking and online, to estimate goals. An ensemble methodology is used to categorise, make future predictions, and carry out particular tasks. When scientists attempt to categorise something, the Random Forest generates a class that nearly all trees have selected. K-fold cross-validation effects are provided by Random Forests. In their supporting data, Scikit-learn and Spark both provide specifics on the impurity factor equations. Users set their variance to serve as a stand-in for categorization and use both of the Gini impurity parameters by default. Mean square error is used by both parameters in regression to calculate variance reduction. Utilising Mean Absolute Error, variability reduction is calculated in Scikit-learn. The mathematical model is shown in Eq. (9) & Eq. (10).

(9)

(10)

The Gini impurity formula is represented by Eq. (1). Where stands for the probabilities of each potential class in the solution space, for purity, and for impurity of a specific node. In this case, Gini only pursues categorical targets.

3.5. Classification via "Attention-enhanced Transfer Learning Intrusion Detection Network" (ATLIDN)

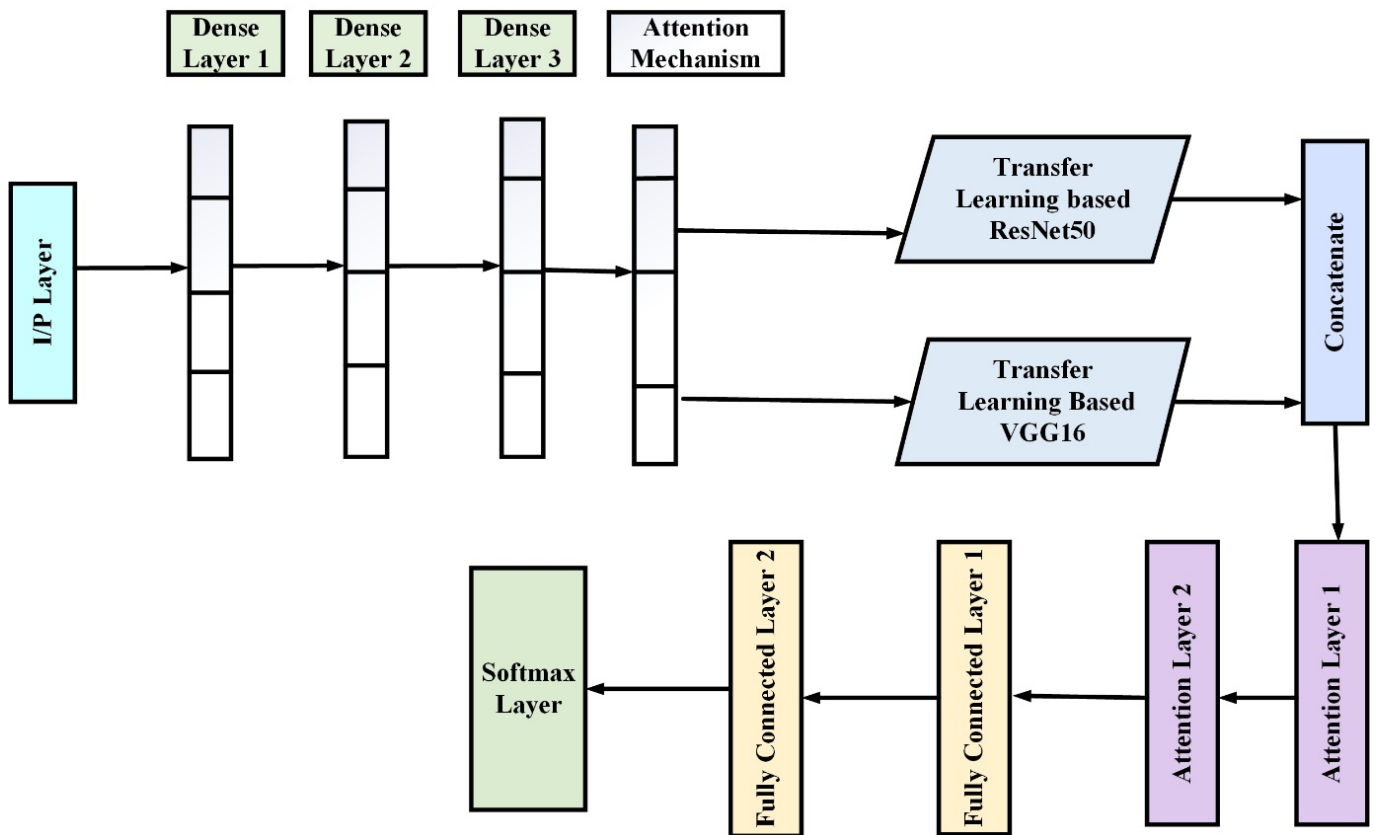


Figure 4: Architecture of ATLIDN

Advantages of Combined ATLIDN Architecture

Dynamically combines ResNet50 and VGG16 pathway features

This architecture skilfully combines the advantages of ResNet's skip connections and VGG's deep convolutional layers, enabling it to adaptably harness the most useful features from either pathway depending on the complexity of the input data.

Established on data, adjusts to the advantages of each architecture

The architecture adapts to data characteristics optimally by dynamically choosing between ResNet50 & VGG16 features, ensuring that intricate details are captured by VGG16-like depth and important spatial information is preserved by ResNet50-like skip connections.

Enhanced ability to occupy a sort of patterns and links in the data

The architecture is better able to seize a kind of originals and relationships in the data by uniting ResNet50 and VGG16 pathways, which accommodate both low-level features and high-level semantic information.

Improved generalisation and overall performance

The architecture achieves improved generalisation capabilities and performance metrics, producing more accurate results across a variety of datasets and tasks. This is made possible by the synergistic fusion of ResNet50 and VGG16 attributes. Figure 4 shows the architecture of ATLIDN.

3.5.1. ResNet 50

The essential concept is posterior ResNet50 is the forerunner of a relation that moves in a singularity-quickly manner and omits at least one layer. The method uses skip connections, which connect skip connections on 2 or 3 layers that have batch normalisation and ReLU between the design. The network is given the option to match the residual mapping rather than have the chance to find the underlying mapping. Consider as an example of an underlying mapping that can be fitted by a number of stacked layers, with standing for the inputs to the first of these layers. If it is assumed that numerous nonlinear layers can eventually imprecise complex functions, this is analogous to assuming that they asymptotically estimate a residual function with definition . Thus, the initial function is changed to . Despite the fact that both should be able to asymptotically estimate the necessary functions, the learning curves for each form may be different. The boon of using a skip link is that regularisation would remove any layer that has a negative impact on the design's performance. As a result, very deep neural networks is trained without being constrained by vanishing gradients, unlike conventional CNN models. Parametric gates are employed in these skip connections in a manner similar to that of LSTM networks. These gates regulate the overall volume of data that passes through the skip connection. ResNet50 addresses the gradient vanishing and feature map vanishing issues when it trains an excessive number of deep CNNs. The ideal mapping that the capability applications want to develop is unaffected by the identity links between non-adjacent layers, which is how ResNet50 operates. Due to gradients' access to a second shortcut channel made possible by the identity connection, back propagation is more fluid. ResNet50, a well-known deep residual network, produces a good balance amid network depth and training efficacy. Architecture of ResNet50 is exhibited in Figure 5.

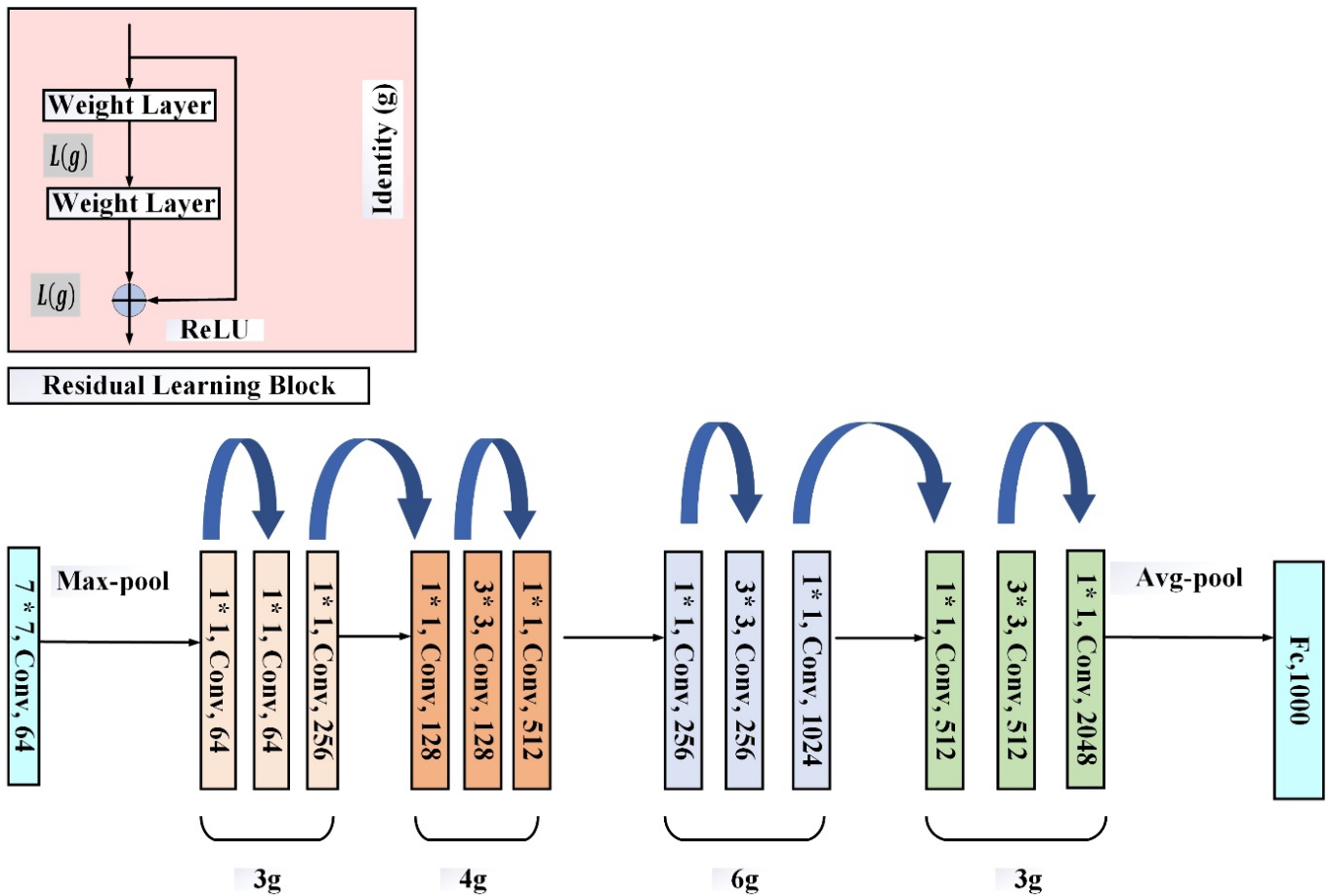


Figure 5: ResNet50 Architecture

3.5.2. VGG 16

With the help of the ImageNet database, VGG-16 was experienced. VGG-16 network has undergone extensive training, so even with small image data sets, it provides excellent accuracy. A small receptive field of 3*3 is available to the VGG-16, which has 16 convolutional layers. There are 5 such layers in total, each with a Max pooling layer of size 2*2. After final layer of Max pooling, there are 3 FC layers. Three fully interconnected layers are added after this. As a final layer, it employs the softmax classifier. All hidden layers are subject to ReLu activation. Architecture of VGG-16 is shown in Figure 6.

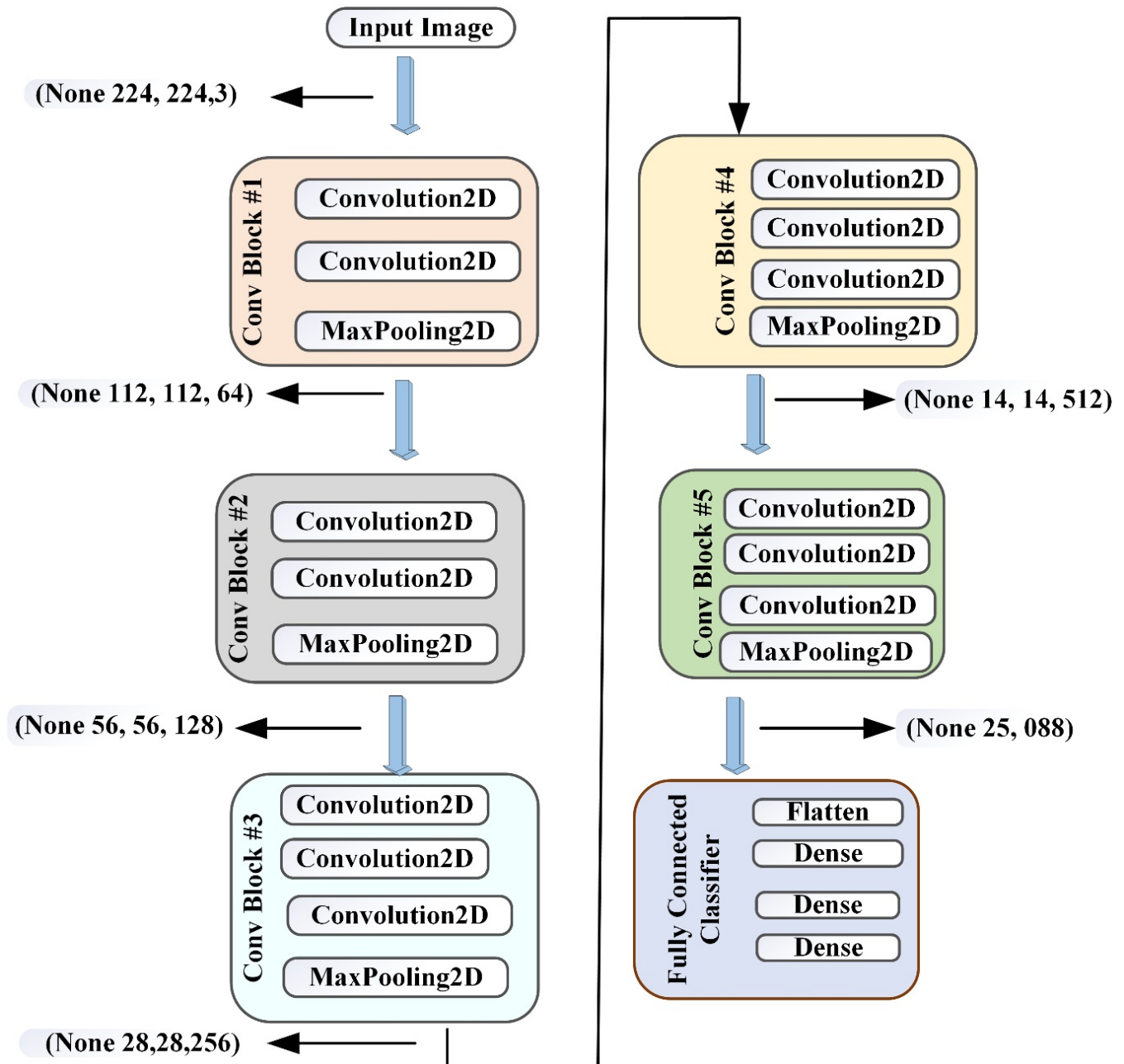


Figure 6: Architecture of VGG16

Results and Discussion

The suggested model has been executed by PYTHON. The intended model has been analysed in terms of Accuracy, FPR, FNR, Precision, F-Measure, Sensitivity, Specificity, and MCC. The outcomes of the suggested and existing methodologies are shown in Table 2.

4.1. Performance Metrics

i) Accuracy

A classification model's performance is assessed by the metric of accuracy. According to Eq. (11), accuracy is the prediction model's percentage for both the number of values that were correctly predicted and the overall number of predicted values.

(11)

ii) Precision

The model's precision measures how accurate it is, or how many of the positive predictions have turned out true. The mathematical expression is shown in Eq. (12)

(12)

iii) Sensitivity

True positive rate is another name for it. It is used to calculate the percentage of positives that are accurately classified as such. The mathematical expression is shown in Eq. (13).

(13)

iv) Specificity

True negative rate is another name for it. It is used to calculate the percentage of negatives that are accurately classified as such. The mathematical expression is shown in Eq. (14).

(14)

v) F-Measure

F1 Score seeks to strike a balance in recall and precision. The mathematical expression of F-Measure is shown in Eq. (15)

(15)

vi) MCC

MCC is a statistical metric used to assess the effectiveness of classification models. The mathematical expression is shown in Eq. (16).

(16)

vii) NPV

According to the actual instances of normal features, this is the proportion of normal feature instances that were correctly detected. The mathematical expression is shown in Eq. (17).

(17)

viii) FPR

The FPR metric shows the ratio of the total number of misclassified normal instances to the total number of normal instances. The mathematical expression is shown in Eq. (18).

(18)

ix) FNR

FNR is another metric that show the ratio of the total number of attack instances to the total number of misclassified attack instances. The mathematical expression is shown in Eq. (19).

(19)

Table 2: Comparative analysis of the proposed methodology and the existing method.

	CNN	ANN	Bi-LSTM	Proposed
Accuracy	0.958259	0.979544	0.94454	0.986824
Precision	0.683198	0.84413	0.580415	0.900786
Sensitivity	0.688566	0.847375	0.584587	0.900033
Specificity	0.977372	0.988911	0.97005	0.992975
F-Measure	0.685871	0.84575	0.582494	0.90041
MCC	0.663521	0.834797	0.552795	0.893355
NPV	0.977916	0.989181	0.970545	0.992916
FPR	0.022628	0.011089	0.02995	0.007025

FNR	0.311434	0.152625	0.415413	0.099967
-----	----------	----------	----------	----------

CNN, ANN, Bi-LSTM, and a Proposed model is compared in the table 2 for performance metrics. The ANN's accuracy score of 0.979544 indicates how well it identify patterns in data. Proposed achieves 0.986824, which is higher of others. The ANN's precision score of 0.84413 demonstrated its strong ability to categorise positive instances correctly. The highest precision, 0.900786, was shown by the proposed model. The most accurate method for correctly identifying positive instances, ANN, showed the highest sensitivity at 0.847375. A sensitivity of 0.900033 was achieved by the proposed model, which also performed well. The ability of ANN to accurately identify negative instances was demonstrated by the highest specificity it achieved (0.988911). Additionally, the proposed model performed remarkably well, with a specificity of 0.992975. The classification accuracy of ANN was highest, with an F-Measure of 0.84575, balancing recall and precision. An F-Measure of 0.90041 indicates that the proposed model also performed remarkably well. The MCC for ANN was the highest, at 0.834797, indicating strong overall performance in binary classification tasks. With an MCC of 0.893355, the proposed model also performed well. The highest NPV was achieved by ANN, which demonstrates its superior ability to recognise true negatives with a precision of 0.989181. Additionally, with an NPV of 0.992916, the proposed model performed remarkably well. The proposed model had the lowest FPR at 0.007025, indicating a superior ability to reduce false positives. As a result, it is especially suitable for applications where avoiding false alarms is crucial. The ANN had the lowest FNR, which was 0.152625, indicating that it is efficient at capturing positive instances. The Proposed model, however, also performed well with a low FNR of 0.099967, suggesting its potential for tasks where reducing false negatives is important.

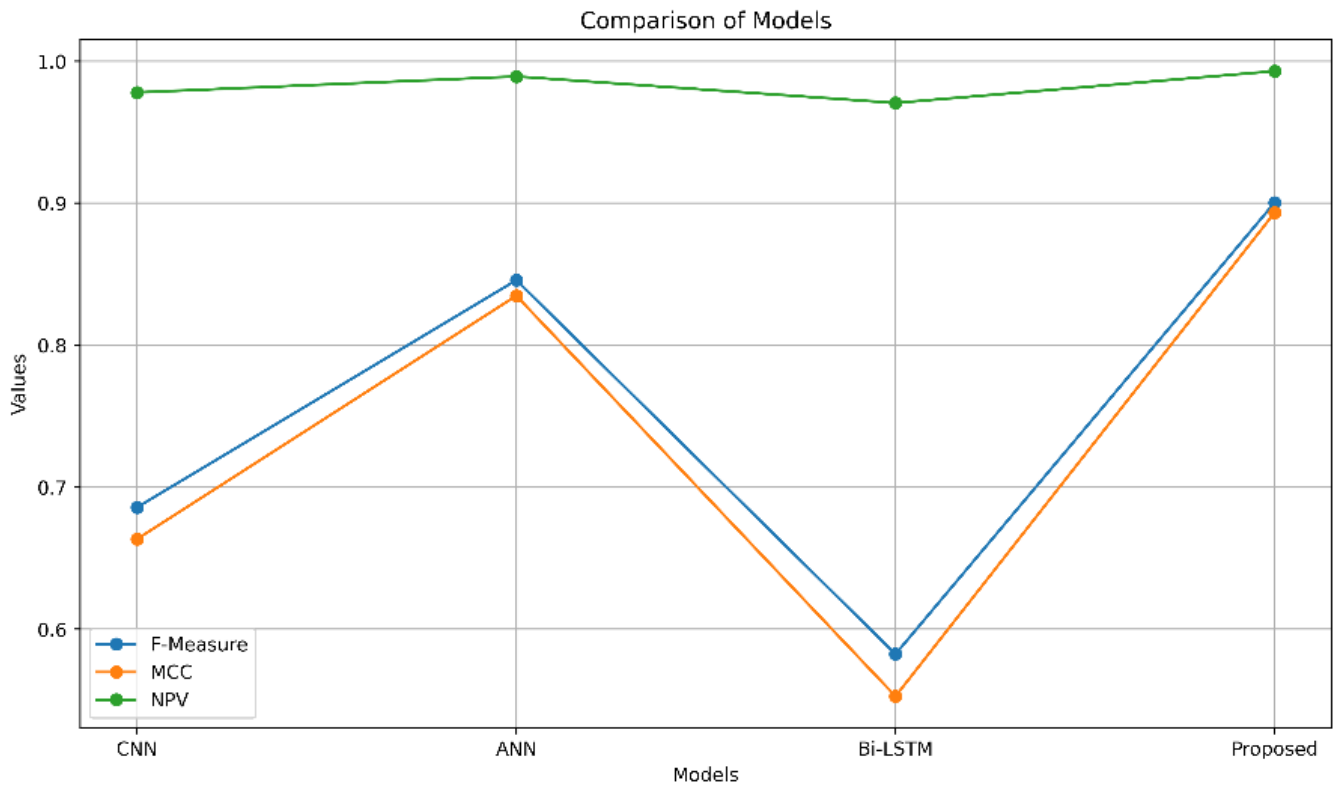


Figure 7: Graphical Representation of F-Measure, MCC & NPV.

F-Measure, MCC, and NPV are just a few of the crucial evaluation metrics where the proposed model performs well it is shown in Figure 7. This points to its potency in achieving a stable precision and recall, robustness in binary classification, and an exceptional capacity to correctly identify true negatives. This performance is essential for a variety of applications, especially those that demand a careful balance between false positives and false negatives as well as a high level of confidence in negative predictions.

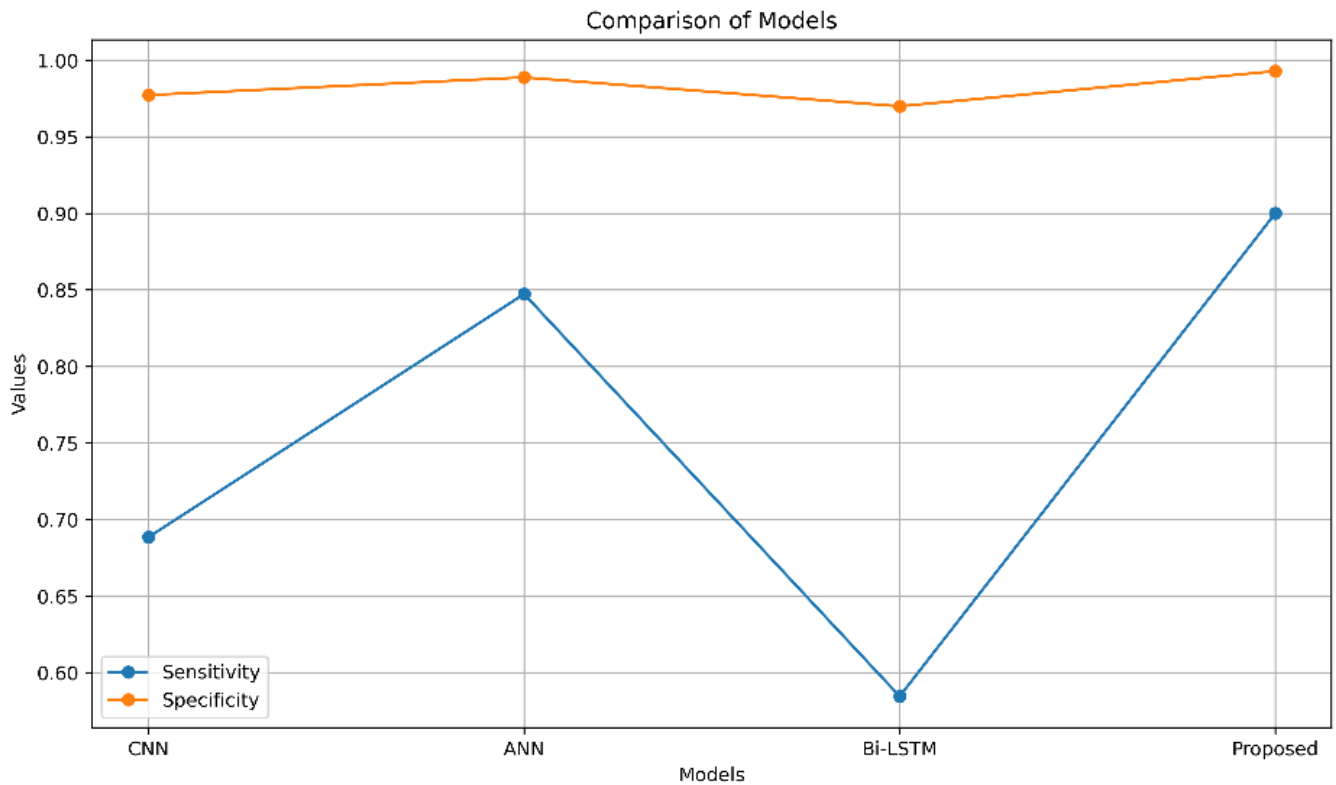


Figure 8: Graphical Representation of Sensitivity & Sensitivity.

The Sensitivity and Specificity scores for the four models show in Figure 8 that they perform differently in terms of correctly identifying positive instances and negative instances. The Proposed model receives high marks for Sensitivity and Specificity, making it a well-rounded option for tasks requiring accurate classification across both positive and negative classes. The ANN model performs exceptionally well in Sensitivity, effectively capturing positive cases.

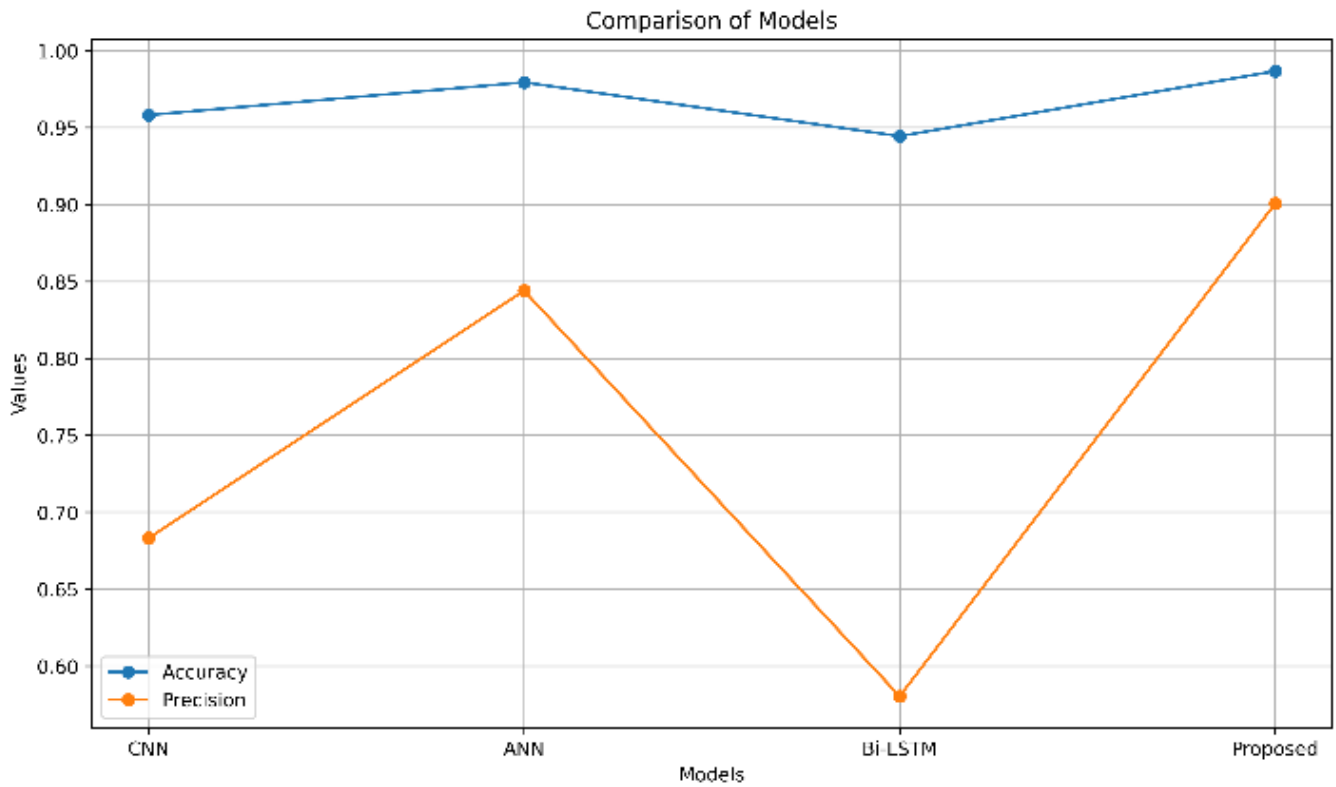


Figure 8: Graphical Representation of Accuracy & Precision

The Proposed model performs better than the Competition in Accuracy and Precision, demonstrating its efficacy in achieving high levels of overall correctness and a strong capacity to classify positive instances correctly it is shown graphically in figure 8.

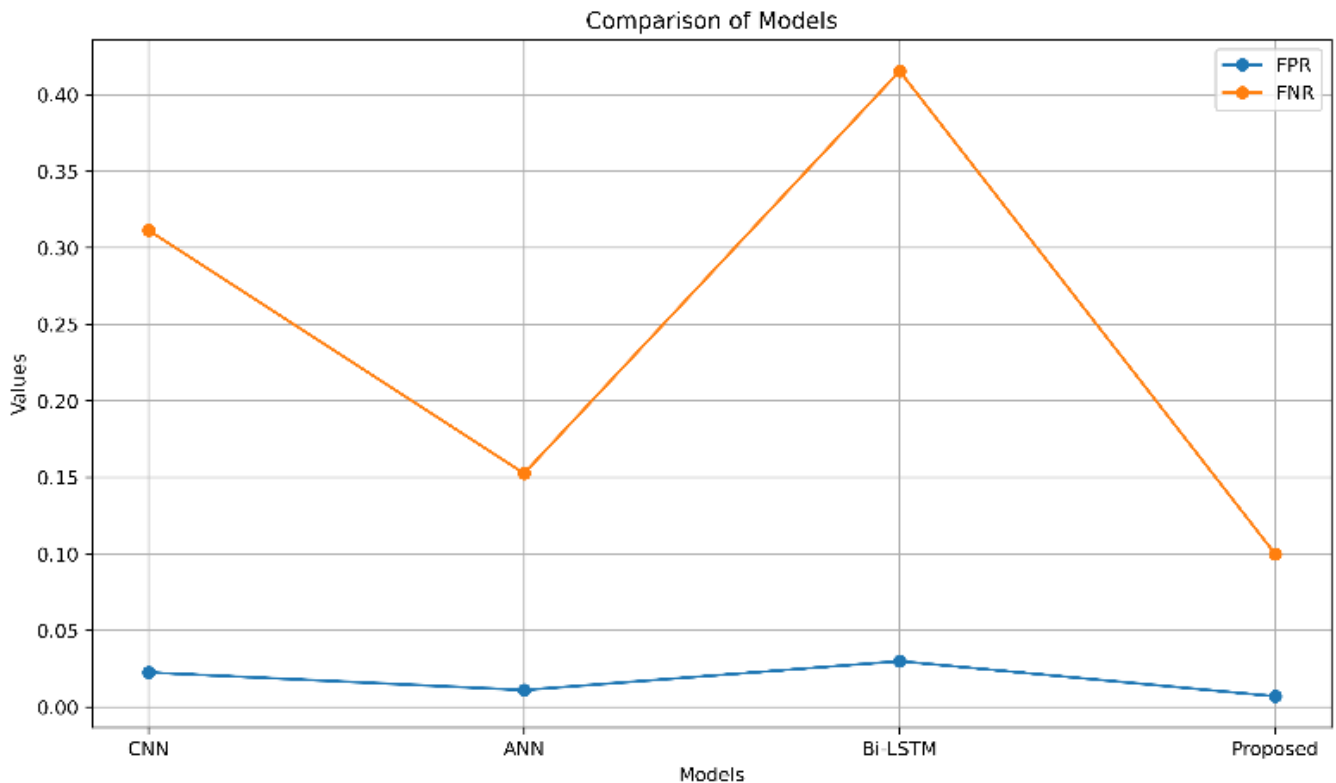


Figure 9: Graphical Representation of FPR & FNR

The Proposed model stands out by achieving both a remarkable low FPR and FNR it is shown in Figure 9, making it a strong option for applications where minimising both types of errors is crucial.

Conclusion

For cloud environments, the DDoS attack IDS employs DL to find intrusions. This system automatically detects and blocks such malicious attacks that have the potential to disrupt cloud services. It does so by utilising cutting-edge DL techniques. Using the power of DL algorithms, the IDS aims to effectively distinguish between normal and abnormal network traffic patterns related to DDoS attacks. This research contributes to improving the security and dependability of cloud services, ensuring the effective operation of cloud-based applications and services by providing an intelligent and automated defence mechanism against DDoS attacks. gather labelled data on IoT network traffic that includes both typical and attack scenarios. The pre-processed data must be cleaned and pre-processed to get rid of noise and anomalies. The data should then be normalised to ensure that the scales of the features are comparable, promoting convergence during training. Use DPI techniques to extract crucial information from network packets, such as IP addresses for the source and destination, packet size, and payload details, in order to extract the features. Use time-series analysis to spot temporal dependencies in network traffic. To model the flow of packets and their interdependencies over time, use transformers or RNNs. When selecting the features, Pearson correlation coefficients are

taken into consideration. Eliminate features with high correlation coefficients to reduce redundancy. Mutual Information Computation was performed. Select features that have a high mutual information score, as these features were important for class distinction. To gradually remove less important features, RFE was applied to the remaining features. Use the Random Forest classifier to rank features according to their importance and eliminate the less important ones. Next, the selected features are displayed. It was advised to use the ATLIDN to categorise data. Utilise the VGG16 and ResNet50 base models to create the ATLIDN architecture. Balance the real-time contributions of the ResNet50 and VGG16 pathways using a self-attention mechanism. For intrusion detection, the pre-trained ResNet50 and VGG16 models were adjusted. The implementation was completed in PYTHON.

References

- [1] Bhushan, K. and Gupta, B.B., 2019. Distributed denial of service (DDoS) attack mitigation in software defined network (SDN)-based cloud computing environment. *Journal of Ambient Intelligence and Humanized Computing*, 10, pp.1985-1997.
- [2] Mahdavi Hezavehi, S. and Rahmani, R., 2020. An anomaly-based framework for mitigating effects of DDoS attacks using a third party auditor in cloud computing environments. *Cluster Computing*, 23(4), pp.2609-2627.
- [3] Wang, X., Guo, N., Gao, F. and Feng, J., 2019. Distributed denial of service attack defence simulation based on honeynet technology. *Journal of Ambient Intelligence and Humanized Computing*, pp.1-16.
- [4] Vu, L., Nguyen, Q.U., Nguyen, D.N., Hoang, D.T. and Dutkiewicz, E., 2022. Deep generative learning models for cloud intrusion detection systems. *IEEE Transactions on Cybernetics*, 53(1), pp.565-577.
- [5] Hu, Q., Yu, S.Y. and Asghar, M.R., 2020. Analysing performance issues of open-source intrusion detection systems in high-speed networks. *Journal of Information Security and Applications*, 51, p.102426.
- [6] Alsharabi, N., Alqunun, M. and Murshed, B.A.H., 2023. Detecting Unusual Activities in Local Network Using Snort and Wireshark Tools. *Journal of Advances in Information Technology*, 14(4).
- [7] Selvaraj, N.P., Paulraj, S., Ramadass, P., Kaluri, R., Shorfuzzaman, M., Alsufyani, A. and Uddin, M., 2022. Exposure of botnets in cloud environment by expending trust model with CANFES classification approach. *Electronics*, 11(15), p.2350.
- [8] Tama, B.A., Comuzzi, M. and Rhee, K.H., 2019. TSE-IDS: A two-stage classifier ensemble for intelligent anomaly-based intrusion detection system. *IEEE access*, 7, pp.94497-94507.
- [9] Singh, P. and Ranga, V., 2021. Attack and intrusion detection in cloud computing using an

ensemble learning approach. *International Journal of Information Technology*, 13, pp.565-571.

[10] Saba, T., Rehman, A., Sadad, T., Kolivand, H. and Bahaj, S.A., 2022. Anomaly-based intrusion detection system for IoT networks through deep learning model. *Computers and Electrical Engineering*, 99, p.107810.

[11] Sun, P., Liu, P., Li, Q., Liu, C., Lu, X., Hao, R. and Chen, J., 2020. DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system. *Security and communication networks*, 2020, pp.1-11.

[12] Kunang, Y.N., Nurmaini, S., Stiawan, D. and Suprpto, B.Y., 2021. Attack classification of an intrusion detection system using deep learning and hyperparameter optimization. *Journal of Information Security and Applications*, 58, p.102804.

[13] Mighan, S.N. and Kahani, M., 2021. A novel scalable intrusion detection system based on deep learning. *International Journal of Information Security*, 20, pp.387-403.

[14] Liu, C., Gu, Z. and Wang, J., 2021. A hybrid intrusion detection system based on scalable K-means+ random forest and deep learning. *Ieee Access*, 9, pp.75729-75740.

[15] Samriya, J.K. and Kumar, N., 2020, October. A novel intrusion detection system using hybrid clustering-optimization approach in cloud computing. In *Materials Today: Proceedings* (Vol. 2, No. 1, pp. 23-54).

[16] Mayuranathan, M., Murugan, M. and Dhanakoti, V., 2021. Best features based intrusion detection system by RBM model for detecting DDoS in cloud environment. *Journal of Ambient Intelligence and Humanized Computing*, 12, pp.3609-3619.

[17] Balasubramaniam, S., Vijesh Joe, C., Sivakumar, T.A., Prasanth, A., Satheesh Kumar, K., Kavitha, V. and Dhanaraj, R.K., 2023. Optimization Enabled Deep Learning-Based DDoS Attack Detection in Cloud Computing. *International Journal of Intelligent Systems*, 2023.

[18] Priyadarshini, R. and Barik, R.K., 2022. A deep learning based intelligent framework to mitigate DDoS attack in fog environment. *Journal of King Saud University-Computer and Information Sciences*, 34(3), pp.825-831.

[19] Arunadevi, M. and Sathya, V., 2022. Optimized back propagation neural network for DDoS attack detection in the cloud environment.

[20] Jaber, A.N. and Rehman, S.U., 2020. FCM-SVM based intrusion detection system for cloud computing environment. *Cluster Computing*, 23, pp.3221-3231.

[21] Erhan, D. and Anarim, E., 2020. Hybrid DDoS detection framework using matching pursuit algorithm. *IEEE Access*, 8, pp.118912-118923.

[22] Varghese, J.E. and Muniyal, B., 2021. An Efficient IDS framework for DDoS attacks in SDN environment. IEEE Access, 9, pp.69680-69699.