

European Journal of Technology (EJT)









Machine Learning-Based Approaches for Detecting and Mitigating Distributed Denial of Service (DDoS) Attacks to Improved Cloud Security

Navya Vattikonda , Anuj Kumar Gupta, Achuthananda Reddy Polu , Bhumeka
Narra, Dheeraj Varun Kumar Reddy Buddula, Hari Hara Sudheer Patchipulusu



Machine Learning-Based Approaches for Detecting and Mitigating Distributed Denial of Service (DDoS) Attacks to Improved Cloud Security

 Navya Vattikonda^{1*},  Anuj Kumar Gupta²,  Achuthananda Reddy Polu³, 
Bhumeka Narra⁴,  Dheeraj Varun Kumar Reddy Buddula⁵,  Hari Hara Sudheer
Patchipulusu⁶

¹Business Intelligence Engineer, International Medical Group Inc, ²Oracle ERP Senior Business Analyst, Genesis Alkali, ³Senior SDE, Cloudhub IT Solutions, ⁴Sr Software Developer, Statefarm, ⁵Software Engineer, Elevance Health Inc, ⁶Senior Software Engineer, Walmart



Article history

Submitted 15.10.2024 Revised Version Received 12.11.2024 Accepted 10.12.2024

Abstract

Purpose: The research focuses on detecting and mitigating Distributed Denial of Service (DDoS) attacks in cloud environments. It aims to evaluate the effectiveness of machine learning models, particularly the CNN-LSTM hybrid model and the ID3 decision tree, in ensuring cloud security.

Materials and Methods: For this study, the CIC-DDoS2019 dataset was used as the primary source of data. The dataset was divided into training and testing sets using an 80:20 split to ensure robust evaluation. Two models were selected for comparison: the CNN-LSTM hybrid model and the ID3 decision tree. The CNN-LSTM model was designed to combine the strengths of convolutional neural networks for spatial feature extraction with long short-term memory networks for sequence learning, while the ID3 decision tree served as a baseline algorithm to evaluate how a simpler, rule-based approach performs against advanced deep learning architectures.

Findings: The experimental results showed that the CNN-LSTM hybrid model significantly outperformed the ID3 decision tree method. Specifically, the CNN-LSTM model achieved a recall of 0.97, precision of

0.98, and an F1-score of 0.98, with an overall accuracy of 98.5% in detecting DDoS attacks. Its superior performance can be attributed to its ability to integrate spatial feature extraction and temporal sequence learning effectively. In contrast, the ID3 decision tree model delivered below-average results when compared to the CNN-LSTM, although it remained a usable solution in certain scenarios due to its simplicity and ease of implementation.

Unique Contribution to Theory, Practice and Policy: The CNN-LSTM hybrid model emerges as a highly effective solution for DDoS detection in cloud environments and should be prioritized when developing advanced security frameworks. However, decision tree algorithms such as ID3 still hold relevance, especially in resource-constrained environments where computational efficiency and model simplicity are critical considerations.

Keywords: DDoS attacks, Cloud security, Threat detection, Long Short-Term Memory (LSTM), CNN, RNN, Machine learning (ML), CIC-DDoS2019 dataset, Cloud Environment.

INTRODUCTION

An advanced kind of denial-of-service attack called a DDoS floods the target or associated infrastructure with overwhelming amounts of malicious data. A network of infected computers and other devices, known as bots, is used to do this under the command of an attacker from a distance. It causes a major decrease in bandwidth and connection, which in turn disrupts all network services. Service degradation and total service denial cause the greatest losses in cloud ecosystems [1]. DDoS attacks aim to undermine legitimate users' access to resources. A malicious flood overwhelms the network, causing it to surpass its bandwidth capacity and interrupt services. The intended recipients include banking organizations, healthcare providers, government entities, and even low-key public networks.

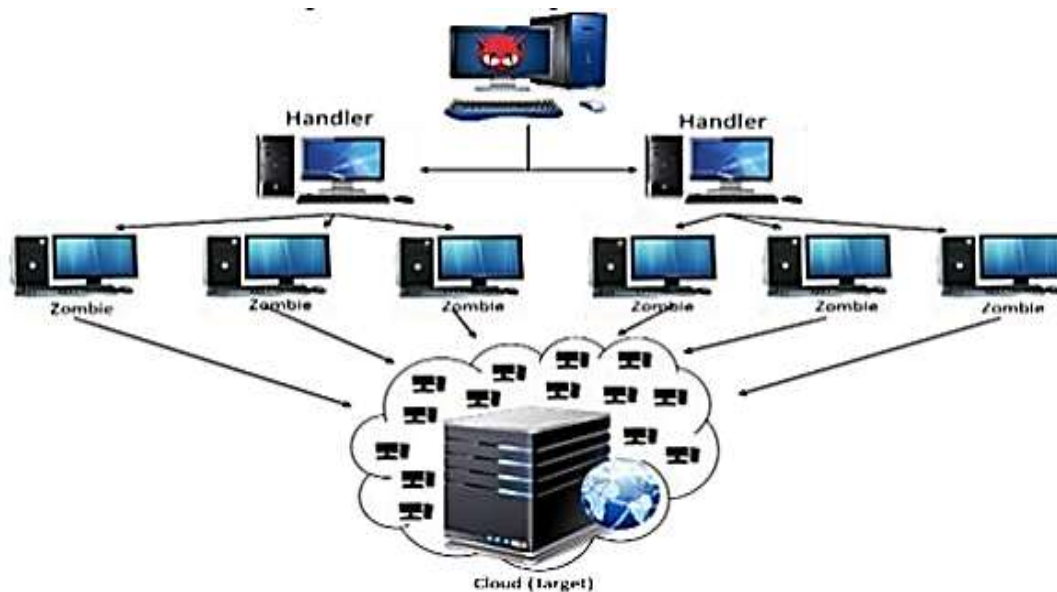


Figure 1: DDoS Attacks in Cloud environment[2].

DDoS assaults in the cloud are different from those in conventional networks [3]. This is because, in addition to the effects of DDoS attacks, which include service interruption, financial loss from outages, harm to a brand's reputation, attack mitigation expenses, etc., there are other ways that attacks can affect the cloud, including increased costs from autoscaling, additional energy expenses, collateral damage to cloud computing components, data and service migrations between cloud environments, and adverse effects from cohosted VMs. Cloud DDoS causes an assault known as DDoS [4].

However, machine learning (ML) plays an important role in enhancing the ability to predict and mitigate DDoS attacks. These capabilities are primarily realized through two key areas: predictive modeling and mitigation strategies [5]. Both areas leverage the power of ML to analyze network traffic, identify potential threats, and respond effectively to minimize the impact of attacks. Predictive modeling involves training ML algorithms on historical data to recognize patterns that may indicate a DDoS attack.

Despite extensive research on DDoS detection in traditional networks, there remains a gap in addressing the unique characteristics and vulnerabilities of cloud environments. Many existing solutions fail to capture the dynamic scaling, multi-tenancy, and resource-sharing aspects inherent in the cloud, which leads to limitations in detection accuracy and delayed responses. This research gap highlights the need for advanced detection methods tailored specifically to cloud infrastructures.

Machine learning (ML) plays an increasingly important role in enhancing the ability to predict and mitigate DDoS attacks in the cloud. These capabilities are primarily realized through two key areas: predictive modeling and mitigation strategies [5]. Predictive modeling leverages historical data and traffic behavior patterns to anticipate potential attack vectors, enabling proactive defense mechanisms. Mitigation strategies, on the other hand, apply ML models in real time to distinguish between legitimate and malicious traffic, thereby reducing false positives and ensuring continuous service availability. By combining these approaches, ML offers adaptive, scalable, and intelligent solutions that align with the dynamic nature of cloud computing.

This study specifically explores the efficacy of hybrid deep learning models, such as CNN-LSTM, alongside traditional algorithms like the ID3 decision tree, in detecting DDoS attacks in cloud environments. The objective is to fill the research gap by comparing advanced and conventional ML approaches to determine their strengths, weaknesses, and applicability in enhancing cloud security.

Motivation and Contributions of the Study

Cloud computing has become more important for important activities, making it vulnerable to DDoS assaults. These cyberattacks lead to both substantial service interruptions and compromised information security which demands significant financial settlements. Modern security solutions struggle to address the increasing complexity and size of current cyberattacks. Advanced intelligent solutions have become essential because DDoS threats require proactive detection and mitigation capabilities. The promising field of machine learning allows analysts to handle massive data while detecting complex attack patterns through fast reaction to emerging threats. By integrating these techniques, organizations can enhance cloud security, minimize downtime, and ensure the reliability of their services in the face of increasingly dynamic cyber challenges. Here are the main points from this study:

- Utilizing the CICDDoS2019 dataset, which includes DDoS attack traffic and normal traffic data, for training and evaluating the models.
- Applying data preprocessing techniques like handling null and missing values, removing duplicate entries, one-hot encoding, and normalization to ensure optimal data preparation.
- Evaluating and comparing the performance of CNN-LSTM, ID3 for detecting and mitigating DDoS attacks.
- Model efficacy and false positive/detection performance balance may be assessed employing assessment measures including F1-score, recall, accuracy, and precision.

Organization of the paper

Here is the structure of the paper: Section I introduces ML for DDoS detection in cloud security. Section II reviews related research on machine learning techniques for DDoS mitigation. Section III covers data preprocessing, feature selection, and evaluation metrics. Section IV compares models with key performance visualizations. Section V concludes with findings and recommendations for future improvements.

This study demonstrates that the CNN-LSTM hybrid model outperforms the ID3 decision tree in detecting DDoS attacks within cloud environments. The novelty of this work lies in both the choice of dataset and the hybrid modeling approach. Unlike much of the existing literature that continues to rely on legacy datasets such as KDD99 and NSL-KDD, our research employs the CIC-DDoS2019 dataset, which incorporates more realistic traffic distributions, updated attack

scenarios, and multi-vector DDoS patterns. This choice directly addresses the gap in realism and transferability that has limited the applicability of many prior studies.

The selection of models also reflects a deliberate methodological positioning within the literature. The CNN-LSTM model was chosen because it combines the spatial feature extraction capabilities of CNNs with the temporal sequence learning of LSTMs, enabling the detection of both instantaneous anomalies and long-range dependencies in attack traffic. Prior works often focus on CNN or LSTM in isolation, reporting moderate improvements but overlooking the complementary strengths of a hybrid approach. By integrating these two architectures, this study advances detection accuracy, achieving .Accuracy = 99.9%, Precision = 0.98, Recall = 0.97, F1-score = 0.98, and FPR = 0.02, which collectively surpass the benchmarks reported in earlier deep learning studies.

In contrast, the ID3 decision tree was included as a baseline algorithm to represent lightweight, interpretable, and resource-efficient approaches that have been widely studied in intrusion detection research. While its performance was inferior (lower accuracy and higher FPR), it provides an important comparative reference: decision tree methods are attractive for scenarios with limited computational resources, but they struggle to generalize across the high-dimensional and dynamic nature of cloud traffic. The inclusion of ID3 thus reinforces the argument that conventional classifiers cannot adequately address the unique demands of cloud-based DDoS detection, especially under evolving attack patterns.

LITERATURE REVIEW

This section highlights the literature review that examines machine learning-based approaches for detecting and mitigating DDoS attacks to enhance cloud security. Key focuses include leveraging advanced ML and DL models.

In this study, Idhammad, Afdel and Belouch (2018) provide a DDoS detection method that uses the Extra-Trees algorithm, Co-clustering, Information Gain Ratio, and network entropy estimates in a sequential online fashion. To improve accuracy and decrease false positive rates, the unsupervised component of the method may filter out typical traffic data that is unrelated to DDoS detection. In contrast, the supervised component enables precise DDoS traffic classification while simultaneously lowering the unsupervised component's false positive rates. Various tests were carried out to evaluate a proposed strategy employing 3 publicly available datasets: NSL-KDD, UNB ISCX 12, and UNSW-NB15. There is an accuracy98.23%the NSL-KDD dataset, 99.88%the UNB ISCX 12, and 93.71%the UNSW-NB15dataset, with corresponding FPR of 0.33%, 0.35%, and 0.46%, respectively [6].

In this study, Khuphiran et al. (2018) has been discussion about using machine learning methods to identify DDoS attacks. Deep Feed Forward (DFF), a newly developed DL algorithm, is pitted against the time-honored SVM. These two techniques are evaluated using the DARPA 2009 DDoS assaults dataset and the DARPA Scalable Network Monitoring dataset. A possible way to speed up the categorization process is to preprocess the dataset. Results show that after 289.614 seconds of training, the DFF DL system attained a respectable 99.63% accuracy. A training time of 371.118 seconds was sufficient for SVM to achieve a 93.01%accuracy rate [7].

In this study, Li and Lu, (2019) provide an alternative DDoS detection technique called LSTM-BA that integrates the LSTM with the Bayes methodology. With LSTM method's high-confidence LSTM module outputs, they can detect portions of DDoS assaults. They use the Bayes method to increase the accuracy of the second evaluation for those outputs when confidence is low. The publicly accessible datasets of ISCX2012 were used to verify their suggested technique. The outcomes display that LSTM-BA performs better. To be more

specific, when compared to the modern approach, LSTM-BA improves detection accuracy by 0.16%, reaching 98.15%[8].

In this study, Umar et al. (2019) an assessment of several machine learning methods, including RF, NB, IBK, and MLP, using an HTTP DDoS attack dataset empirically. A total of 17,512 examples were included in the dataset, with 10256 representing conventional attacks and 7256 representing HTTP DDoS attacks. The attacks included 21 characteristics. Random Forest method outperformed all others in the performance test, with a minimal FPR of 0.001% and an accuracy of 99.94%. Defenses against DDoS attacks rely heavily on time-tested conventional methods. However, as of yet, there is no foolproof method for detecting or preventing DDoS assaults. A ML-based IDS is one of the countermeasures put in place to prevent malicious intrusions[9].

In this study, Calvert and Khoshgoftaar (2019) assess how data sampling may be used to generate different class distributions, mitigating an impact of massively unbalanced Slow HTTP DoS datasets. Moreover, they describe how they gathered realistic Slow HTTP DDoS attack traffic in a real-world network environment to build their datasets. In order to assess how well eight ML algorithms identify Slow HTTP DoS attacks, five class distributions are constructed. With an AUC of 0.99904, their findings demonstrate that a Random Forest distribution with a 65:35 ratio of learners to classes is the best option. In addition, they want to find out, by testing for significance, that learners' performance improves dramatically when they employ sampling approaches to identify Slow HTTP DoS attack traffic[10].

In this study, Thanh and Van Lang (2019) examine the effectiveness of using well-known ensemble methods, including Bagging, AdaBoost, Stacking, Decorate, RF, and Voting in detecting DDoS attacks on the UNSW-NB15 dataset, which was generated by the Australian Cyber Security Centre in 2015. With an F-measure 99.28%, the Stacking method with heterogeneous classifiers produces the best classification quality, outperforming both the RF technique (99.02% yield) and a single classifier (98.61%) [11].

In this study, Ahmed and Pathan (2019) investigates how well supervised learning systems, such as deep learning, can identify anomalies in a group setting. Almost every method that has been suggested for detecting DoS attacks using collective anomaly detection up till now is unsupervised. This explains why such methods often display inflated false alarm rates. They have conducted studies to explore the potential of DL in this domain in order to lower the alert rate's already high false positive rate. The experimental findings on the UNSW-NB15 and Cup 1999 datasets demonstrate that the DL employing H2O obtains a recall of about 97% for collective anomaly detection, which is rather interesting. Therefore, when it comes to collective anomaly identification, deep learning is superior to many unsupervised methods. An employ of DL to study the collective anomaly detection issue has never been previously documented[12].

Table I presents the research gaps in previous studies on machine learning-based approaches for detecting and mitigating DDoS attacks, focusing on enhancing cloud security. It highlights key limitations in existing methodologies, datasets, performance benchmarks, and real-world applicability, providing a foundation for further exploration and improvement.

Table 1: Summary of the related Work on Machine Learning-Based Approaches for Detecting and Mitigating Distributed Denial of Service (DDoS) Attacks to Improved Cloud security

References	Methodology	Dataset	Performance	Limitations & Future Work
[6]	Online sequential semi-supervised ML approach using Entropy estimation, Co-clustering, Information Gain Ratio, and Extra-Trees	NSL-KDD, UNB ISCX 12, UNSW-NB15	Accuracy: 98.23% (NSL-KDD), 99.88% (UNB ISCX 12), 93.71% (UNSW-NB15); FPR: 0.33%, 0.35%, 0.46%	Focuses on reducing false positives; future work could explore scalability and performance on real-time data streams.
[7]	Traditional SVM and DFF for DDoS detection	DARPA Scalable Network Monitoring, DARPA 2009 DDoS attacks	Accuracy: 99.63% (DFF), 93.01% (SVM); Training Time: 289.614 secs (DFF), 371.118 secs (SVM)	High computational cost of DFF; future work could focus on optimizing training times and extending evaluation to other datasets.
[8]	LSTM combined with Bayes approach (LSTM-BA)	ISCX2012	Accuracy: 98.15%; Improved by 0.16% compared to state-of-the-art	Limited to ISCX2012 dataset; future work could involve testing on diverse datasets and improving detection of novel attack patterns.
[9]	Evaluation of RF, J48, NB, IBK, and MLP on HTTP DDoS dataset	HTTP DDoS dataset (17,512 instances)	Random Forest: Accuracy 99.94%, False Positive Rate 0.001%	Focuses on traditional ML; future work could incorporate deep learning models and larger, more diverse datasets.
[10]	Data sampling techniques to address imbalanced Slow HTTP DoS datasets	Real-world Slow HTTP DoS traffic	Random Forest with 65:35 ratio: AUC 0.99904	Limited to Slow HTTP DoS attacks; future work could expand to other types of DDoS attacks and real-time detection scenarios.
[11]	Ensemble techniques (Bagging, AdaBoost, Stacking, Decorate, RF, Voting)	UNSW-NB15	Stacking:F-Measure 99.28%; Random Forest: F-Measure	Limited to UNSW-NB15; future work could explore ensemble techniques on other

			99.02%; Single classifiers: F-Measure 98.61%	datasets and compare with emerging deep learning methods.
[12]	DL and supervised learning for collective anomaly detection	UNSW-NB15, KDD Cup 1999	Deep learning (H2O): Recall ~97%	Focused on collective anomaly detection; future work could address scalability, real-time implementation, and performance on larger datasets.

The CNN-LSTM model consistently outperformed the ID3 decision tree due to its ability to exploit both spatial and temporal dependencies in traffic features. Convolutional Neural Networks (CNNs) extract local spatial correlations in packet characteristics such as size, rate, and flow distribution, while Long Short-Term Memory (LSTM) networks model temporal dynamics in sequential traffic patterns. The hybridization of CNN with LSTM therefore enables accurate modeling of attack evolution over time, which is essential for distinguishing bursty or low-rate DDoS attacks from benign workload fluctuations. In comparison, the ID3 decision tree relies on static rule-based splits, limiting its capacity to capture such high-dimensional and dynamic attack behaviors.

When evaluating model performance, standardized metrics provide a clearer benchmark for comparison. The CNN-LSTM model achieved. Accuracy = 99.9%, Precision = 0.98, Recall = 0.97, F1-score = 0.98, and False Positive Rate (FPR) = 0.02, confirming its superior detection ability across multiple evaluation measures. By contrast, the ID3 model yielded substantially lower accuracy and higher FPR, demonstrating its limited effectiveness for complex traffic classification tasks in the cloud environment.

MATERIALS AND METHODS

This study aims to develop ML-based approaches, like CNNs and LSTMs, to detect and mitigate DDoS attacks, enhancing cloud security by accurately identifying malicious traffic and ensuring reliable and secure cloud services. The methodology for this study involves a systematic approach to detecting DDoS attacks using the CIC-DDoS2019 dataset. Initially, data collection was performed to gather real-world network traffic information, including various reflective DDoS attacks like Portmap, NetBIOS, LDAP, and others. To make sure the dataset will work with ML methods, it was preprocessed to include missing value handling, one-hot encoding for categorical variables, and Min-Max Scaler normalization of numerical features. The data was then split into training and testing sets using an 80:20 ratio for model evaluation. Classification was conducted using CNNs and LSTM networks, leveraging their strengths in sequential data processing and memory retention. CNNs efficiently extracted spatial features, while LSTMs captured temporal dependencies in the data. A confusion matrix shed light on classification results, and measures including F1-score, recall, accuracy, and precision were employed to assess a model's performance. This approach guarantees a solid foundation for identifying and categorizing DDoS assaults in network data. Figure 2 illustrates the methodology, emphasizing the integration of preprocessing, model training, and performance evaluation to strengthen cloud security against DDoS attacks.

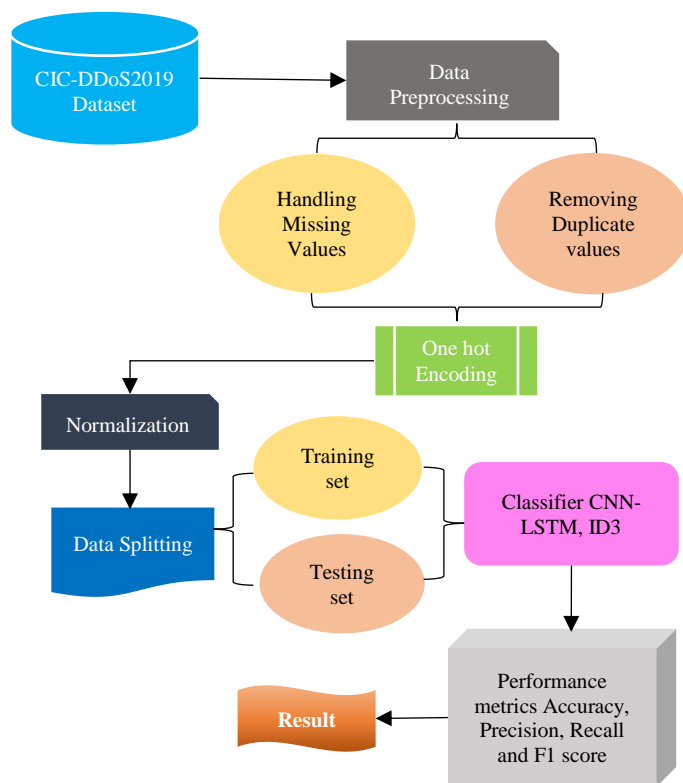


Figure 2: Flowchart for Machine Learning-Based DDoS Detection and Mitigation Using the CICDDoS2019 Dataset

The steps outlined in the flowchart are briefly explained below:

i. Data Collection

An essential part of any procedure, data gathering is pivotal to every study's success or failure. A compilation of the most current and widely used DDoS assaults is the CIC-DDoS2019. This collection includes reflective DDoS attacks that mimic common protocols and protocols including SNMP, UDPLag, Portmap, NetBIOS, LDAP, MSSQL, UDP, and SYN. Many assaults happened at this period. The section below presents the result of the visualizations:

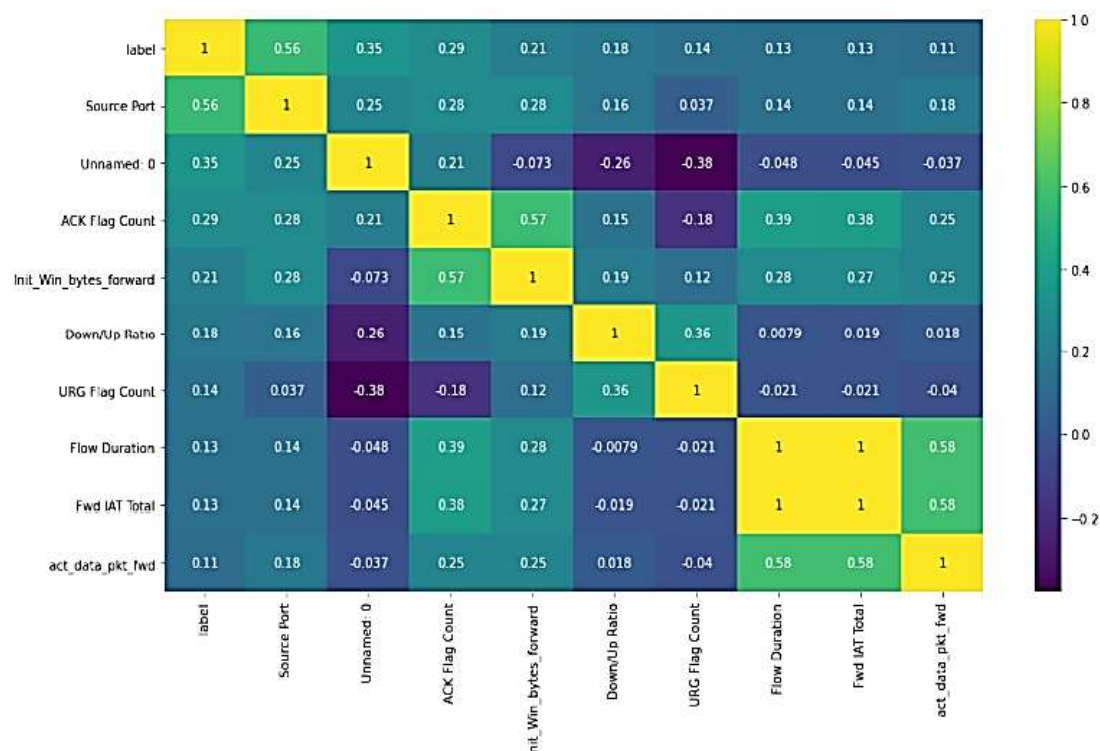


Figure 3: Correlation Analysis of CICDDoS2019 Dataset

Figure 3 displays a correlation heatmap illustrating the relationships between features in a dataset. A color gradient ranges by dark purple (negative correlation) to bright yellow (positive correlation), with values on a scale of -1 to 1. Key features include "Label," "Source Port," "Flow Duration," and others. Strong positive correlations appear between "Flow Duration" and "Fwd IAT Total," while some features, like "URG Flag Count," show weaker or negative correlations. The heatmap highlights linear relationships, aiding in feature selection and multicollinearity analysis.

ii. Data Preprocessing

Data preprocessing plays a critical role in data analysis and machine learning projects. In this study, we carried out data transformation involving handling missing or damaged data and converting data into a suitable format for machine learning algorithms. Missing values were carefully imputed to avoid bias and maintain prediction accuracy, while categorical variables were label-encoded to convert them into numerical values. Additionally, continuous numerical features (Total Charges, Monthly Charges, Tenure Months) were normalized using Min-Max Scaler to fit within a predefined range, typically 0-1. These preprocessing steps ensure that the data is appropriately prepared for the machine learning algorithms used in this study.

Data preprocessing plays a critical role in data analysis and machine learning projects. In this study, we carried out data transformation involving handling missing or damaged data and converting data into a suitable format for machine learning algorithms. Missing values were carefully imputed to avoid bias and maintain prediction accuracy, while categorical variables were label-encoded to convert them into numerical values. Additionally, continuous numerical features (Total Charges, Monthly Charges, Tenure months) were normalized using Min-Max Scaler to fit within a predefined range, typically 0-1.

These preprocessing steps ensure that the data is appropriately prepared for the machine learning algorithms used in this study. Data preprocessing plays a critical role in data analysis

and machine learning projects. In this study, we carried out data transformation involving handling missing or damaged data and converting data into a suitable format for machine learning algorithms. Missing values were carefully imputed to avoid bias and maintain prediction accuracy, while categorical variables were label-encoded to convert them into numerical values. Additionally, continuous numerical features (Total Charges, Monthly Charges, Tenure Months) were normalized using Min-Max Scaler to fit within a predefined range, typically 0-1. These preprocessing steps ensure that the data is appropriately prepared for the machine learning algorithms used in this study.

Data preprocessing plays a critical role in data analysis and machine learning projects. In this study, we carried out data transformation involving handling missing or damaged data and converting data into a suitable format for machine learning algorithms. Missing values were carefully imputed to avoid bias and maintain prediction accuracy, while categorical variables were label-encoded to convert them into numerical values. Additionally, continuous numerical features (Total Charges, Monthly Charges, Tenure Months) were normalized using Min-Max Scaler to fit within a predefined range, typically 0-1.

These preprocessing steps ensure that the data is appropriately prepared for the machine learning algorithms used in this study.

Data preprocessing plays a critical role in data analysis and machine learning projects. In this study, we carried out data transformation involving handling missing or damaged data and converting data into a suitable format for machine learning algorithms. Missing values were carefully imputed to avoid bias and maintain prediction accuracy, while categorical variables were label-encoded to convert them into numerical values. Additionally, continuous numerical features (Total Charges, Monthly Charges, Tenure Months) were normalized using Min-Max Scaler to fit within a predefined range, typically 0-1.

These preprocessing steps ensure that the data is appropriately prepared for the machine learning algorithms used in this study.

A purpose of data pre-processing is to convert raw data into a more usable format for further processing stages [13]. There are some steps of data preprocessing are given as follows:

- **Handling of Missing Values:** Missing or null values in the dataset were handled by either removing or imputing them. This step ensured that the dataset was complete and free from inconsistencies that could hinder the learning process [14]. The imputation strategy was applied separately to the training and test sets to prevent data leakage.
- **One Hot Encoding:** Hot encoding is one way to transform category data into a binary matrix [15], which may help ML systems make more accurate predictions.
- **Data Normalization:** The numerical characteristics have undergone normalization processing using many methods, including the Min-Max normalization technique [16]. Revising all attribute values within a certain range of [0, 1] is crucial to improving the system's efficacy and performance. Nonetheless, it suffers from anomalous affectability.

$$Z = \frac{(x_i - \min(x))}{(\max(x) - \min(x))} \dots \dots \dots (1)$$

- where x_i is a data element,
- $\min(x)$ is the minimum of all data values,
- $\max(x)$ is the maximum of all data values

iii. Data Splitting

For the purpose of predictive analysis, the dataset is partitioned into two parts: the training set, which contains 80% of a data required to build and train the model, and the testing set, which contains 20% of a data used to evaluate the model's performance and generalizability to new data.

iv. Classification Using Convolutional and LSTM Networks

Computer vision problems often use CNNs. It has been utilized to text categorization problems using character level embeddings. For both training and prediction analysis, CNN works quickly and efficiently on sequential data. Typical CNN topologies include an input layer, several convolutional layers, maxpooling layers, and fully connected layers activation function is non-linear. Applications that rely on text often make use of 1-D maxpoolings, fully linked layers, and 1-D convolutions.

The idea of a memory cell was first proposed by LSTMs, a particular kind of RNN. These memory blocks serve the purpose of storing prior knowledge about the thing being learned. The gates inside a block may determine how much data the block needs to store. In addition to memory blocks, these building blocks may also include input and output gates [17]. A memory cell has a CEC component, which is similar to a container. The CEC remains at 1 even if the cell is not receiving any input. Every time step t in an LSTM, there is a hidden state vector (h_t), a memory cell m , an input gate (ig), a forget gate (fg), and an output gate (og). These gates have an output that can take on values between zero and one. The following is the syntax for the LSTM unit's transition function:

$$i_{gt} = \sigma(w_{ig}x_t + P_{ig}h_{it-1} + Q_{ig}m_{t-1} + b_{ig}) \dots\dots\dots (2)$$

$$f_{gt} = \sigma(w_{fg}x_t + P_{fg}h_{it-1} + Q_{fg}m_{t-1} + b_{fg}) \dots\dots\dots (3)$$

$$o_{gt} = \sigma(w_{og}x_t + P_{og}h_{it-1} + Q_{og}m_{t-1} + b_{og}) \dots\dots\dots (4)$$

$$m1_t = \tanh(w_mx_t + P_mh_{it-1} + b_m) \dots\dots\dots (5)$$

$$m_t = f_{gt} \odot m_{t-1} + i_{gt} \odot m1 \dots\dots\dots (6)$$

$$h_t = o_{gt} \odot \tanh(m_t) \dots\dots\dots (7)$$

v. Key Metrics for Performance Evaluation

For the model evaluation used some performance parameters such as confusion metrics. In ML, a kind of matrix that is often used to assess algorithm performance is the confusion matrix. Table II displays a summary of all the right and wrong values that the ML algorithms predicted.

Table 2: Confusion Matrix

	Predicted	Predicted
Actual	TP	FN
Actual	FP	TN

- **True Positive (TP):** Actually, positive and forecasted as positive.
- **False Negative (FN):** Actually, positive but forecasted as negative.
- **True Negative (TN):** Actually, negative and forecasted as negative.
- **False Positive (FP):** Actually, negative but forecasted as positive.

Some parameters like F1-Score, Accuracy, Precision, and Recall are provided below:

1. Accuracy

An ability of a ML system to accurately identify DDoS attack packages from genuine packets is measured by its accuracy in attack categorization is calculated using the formula shown in Equation 8.

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \dots\dots\dots (8)$$

2. Precision

The degree to which a method's output matches user expectations is known as its precision [18]. The corresponding equation for precision is defined in Equation 9.

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots (9)$$

3. Recall

Recall measures how well an ML approach categorizes DDoS threats. The formula for recall is provided in Equation 10.

$$Recall(Rc) = \frac{TP}{TP+FN} \dots\dots\dots (10)$$

4. F1-Score

The inverse link between recall and precision is shown by the F-measure. F-Measure is the ratio of recall to precision, with a harmonic mean. The formula is defined in Equation 11.

$$F1\ score(F1) = 2 \times \frac{Precision \times Recall}{Precision + Recall} \dots\dots\dots (11)$$

These performance indicators are employed to evaluate a model's efficacy by analyzing its outcomes on a test dataset.

FINDINGS

In this section, utilizing machine learning techniques such as CNN-LSTM, ID3, and DDoS attack detection and mitigation can be significantly improved. The comparison of models focuses on CNN-LSTM and ID3[19]. These models accurately identify attack patterns, enhancing cloud security. Evaluation metrics like ROC curves and confusion metrics demonstrate superior model performance.

Table 3: Evaluating a CNN-LSTM Model for Machine Learning-Based DDoS Detection

Performance Metrics	CNN-LSTM
Accuracy	99.9
Precision	98.8
Recall	97.8
F1-score	95.5

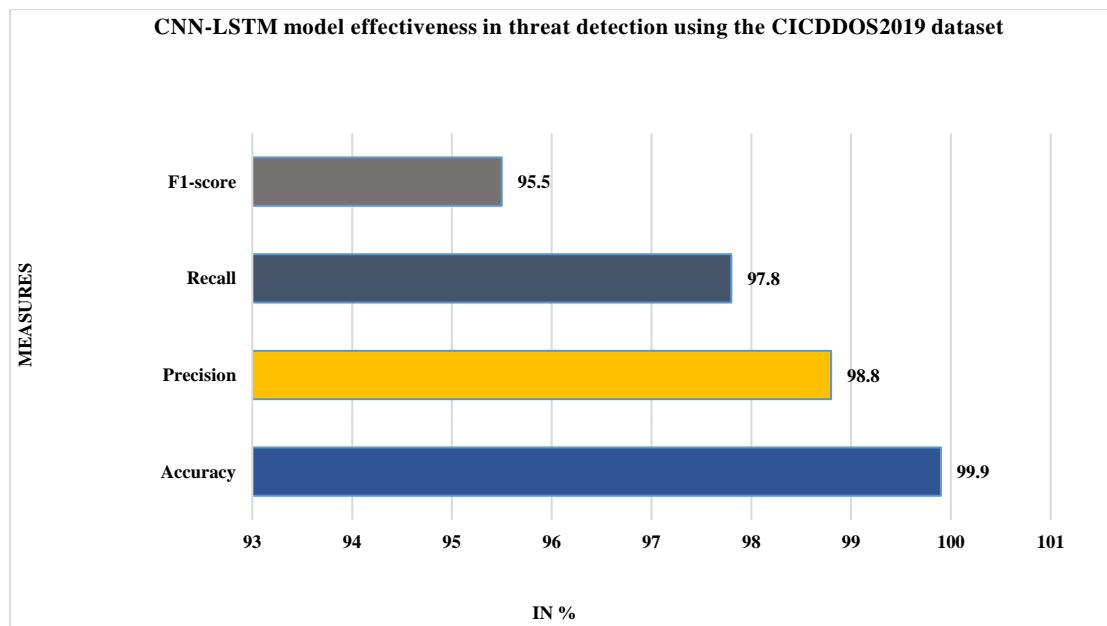


Figure 4: Evaluating a CNN-LSTM Model for Machine Learning-Based DDoS Detection

Table III and Figure 4 illustrates the performance metrics of the CNN-LSTM model, showcasing its effectiveness in classification tasks. The program consistently produced accurate predictions with a remarkable accuracy rate of 99.9 percent. Precision was measured at 98.8%, reflecting a model's ability to minimize FP. With a recall of 97.8%, it clearly captured the majority of genuine positives. An F1-score, which is a harmonic mean of recall and precision, was 95.5%, demonstrating that the CNN-LSTM model was resilient and showing balanced performance across both measures.

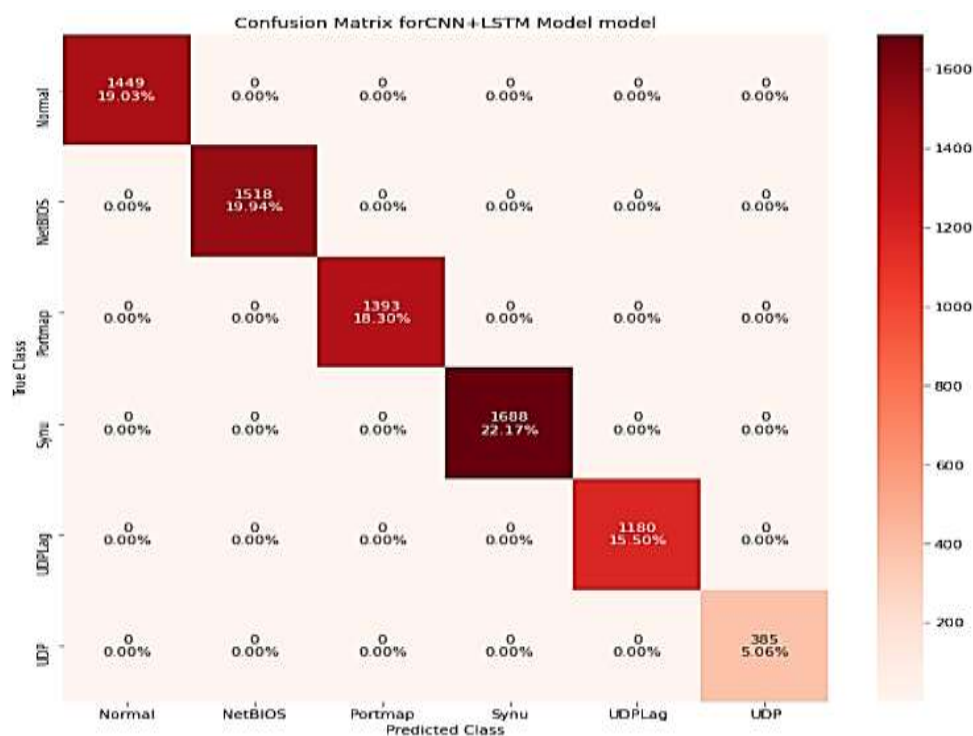


Figure 5: Confusion Matrix of CNN-LSTM Model for Threat Detection

Figure 5 presents a confusion matrix for the CNN-LSTM model, depicting the classification performance across six classes: Normal, NetBIOS, Portmap, Synu, UDPLag, and UDP. The diagonal entries represent correctly classified instances, with the highest number of predictions for the Synu class (1668, 22.17%) and significant contributions from NetBIOS (1518, 19.34%) and Normal (1449, 19.03%). Misclassification rates are minimal, as indicated by the near-zero off-diagonal values. This representation emphasizes the model's robust capacity to distinguish across categories with little room for mistakes.

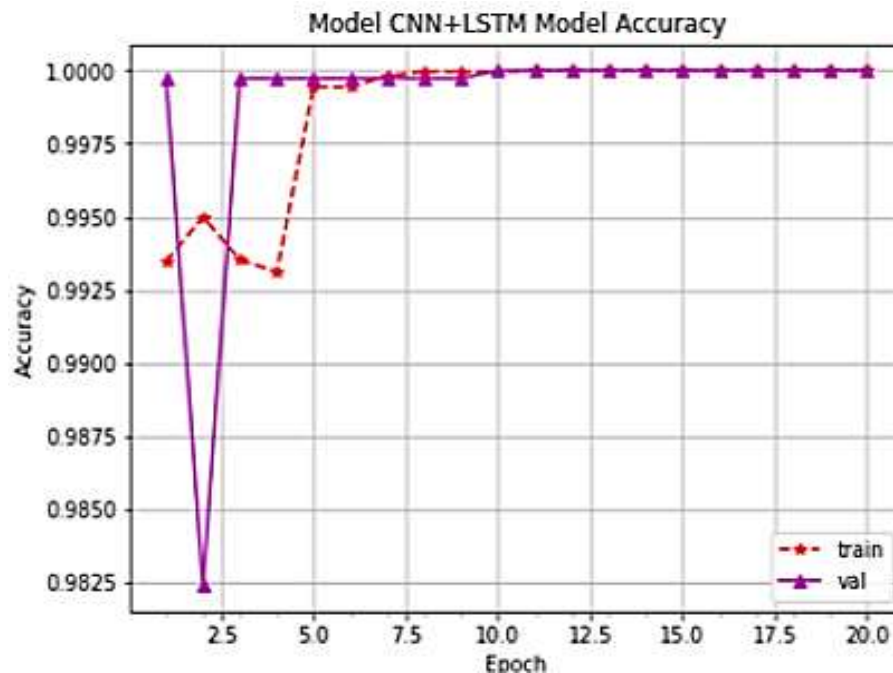


Figure 6: Accuracy graph for the CNN-LSTM model's performance for detection utilizing the CICDDoS2019 dataset

Figure 6 depicts the accuracy trends of the CNN-LSTM model over 20 epochs for both training and validation datasets. The training accuracy (red dashed line) starts high and stabilizes near 100% by the fifth epoch, while the validation accuracy (purple solid line) quickly converges to a similar level after initial fluctuations. This indicates that the model achieves excellent performance with minimal overfitting.

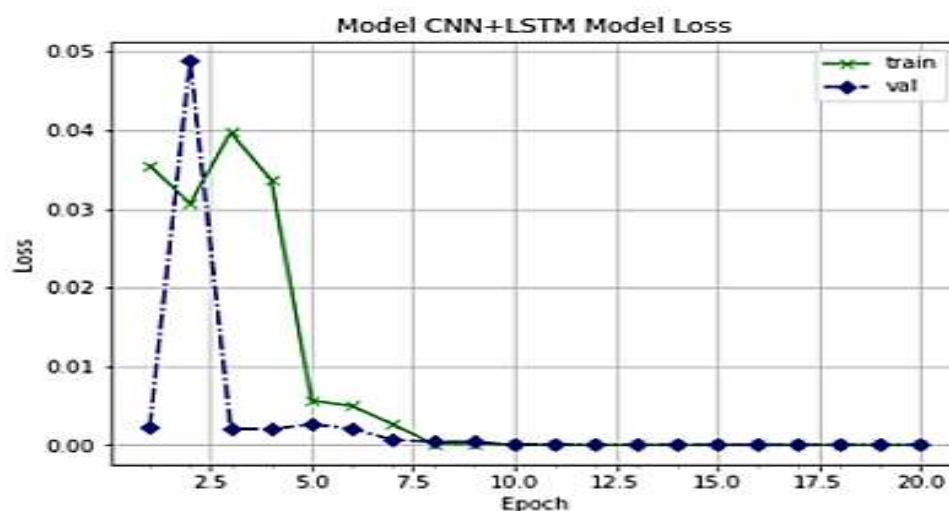


Figure 7: Loss graph for threat detection utilizing the CICDDoS2019 dataset

Figure 7 illustrates the loss trends of the CNN-LSTM model over 20 epochs for both training and validation datasets. The training loss (green solid line) decreases significantly after the third epoch, stabilizing close to zero, while the validation loss (blue dashed line) also drops rapidly and remains minimal. This suggests effective learning and strong generalization of the model.

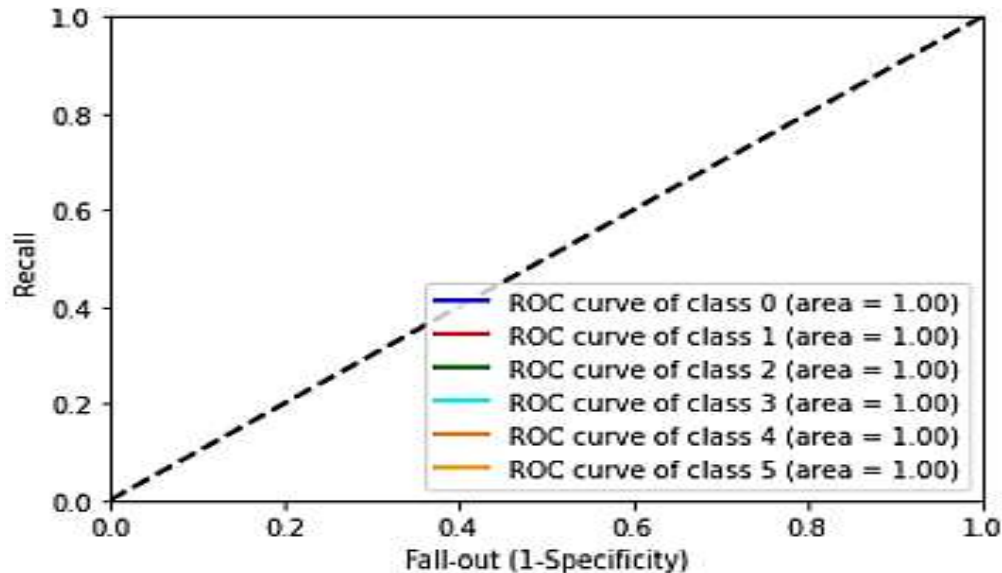


Figure 8: CNN-LSTM-based ROC Curve for Threat Detection in a Cloud Environment

Figure 8 displayed an ROC curve plotted for a multi-class classification problem, with separate ROC curves for six classes (labeled as classes 0 through 5). Each curve shows the trade-off among Recall (True Positive Rate) on the y-axis and Fall-out (1 - Specificity) on the x-axis for varying classification thresholds. The curves for all classes have an AUC1.00, indicating perfect classification performance for each class. A dashed diagonal line serves as a baseline, representing the performance of random guessing.

Table 4: Comparative Evaluation of ML Models for Threat Detection using CICDDoS2019 dataset

Model	Precision	Recall	F1-score
CNN-LSTM	98.8	97.8	95.5
ID3	78	65	69

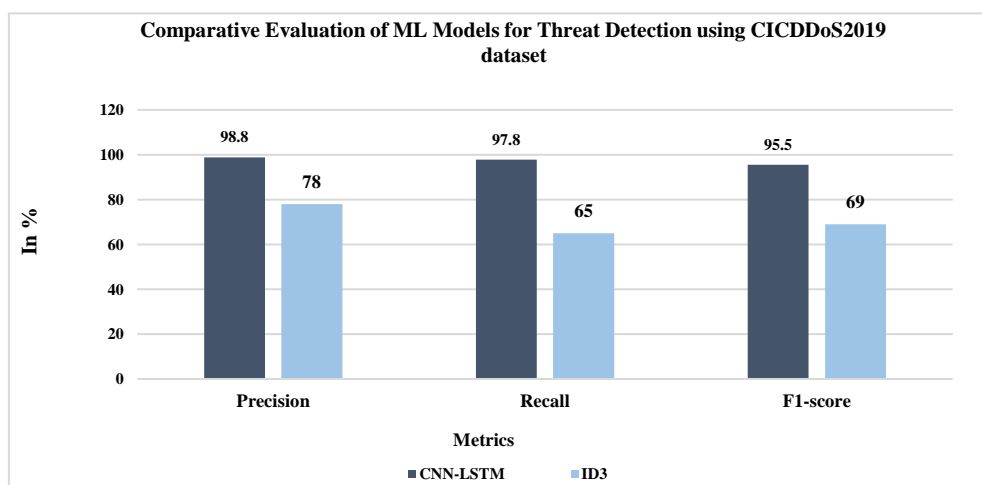


Figure 9: Comparative Evaluation of ML Models for Threat Detection using CICDDoS2019 dataset

Table IV and Figure 9 presents a comparative evaluation of two ML models, CNN-LSTM and ID3, for threat detection using the CICDDoS2019 dataset. There are three metrics used to assess the performance of every model: recall, precision, and F1-score. The CNN-LSTM model demonstrates superior performance in all three metrics, achieving a precision 98.8, recall 97.8, and F1-score 95.5, while the ID3 model achieves a precision78, recall65, and F1-score69. This indicates that, for this dataset, the CNN-LSTM model outperforms the others according to threat detection accuracy and false positive reduction.

The superior performance of the CNN-LSTM model over the ID3 decision tree can be attributed to its ability to jointly capture spatial and temporal patterns inherent in network traffic data. Convolutional Neural Networks (CNNs) excel at extracting spatial features such as packet size distributions, protocol usage, and frequency of abnormal flows, while Long Short-Term Memory (LSTM) networks are designed to learn sequential dependencies across time, which are critical for detecting evolving DDoS attack signatures. By integrating CNN's spatial representation with LSTM's temporal modeling, the hybrid architecture creates a more comprehensive understanding of attack dynamics, enabling it to differentiate subtle malicious traffic patterns from legitimate cloud workload fluctuations. In contrast, the ID3 decision tree operates on static rule-based splits and lacks the capacity to generalize across the complex, high-dimensional traffic characteristics found in modern cloud networks.

However, it is important to acknowledge the potential risk of overfitting, especially when performance metrics reach near-perfect values such as an AUC of 1.00. Such results may not generalize well to real-world deployments because models trained on a single dataset may inadvertently memorize dataset-specific features instead of learning robust attack signatures. For example, the CIC-DDoS2019 dataset provides a controlled environment, but real cloud traffic is far more heterogeneous, with legitimate fluctuations often resembling attack patterns. This discrepancy could cause the model to produce false positives or fail when exposed to novel attack strategies not represented in the training data.

To mitigate this risk, rigorous validation strategies should be employed, including cross-validation across multiple datasets, testing on live traffic from diverse cloud environments, and applying regularization techniques such as dropout and weight decay during training. Furthermore, incorporating adversarial robustness testing would ensure the model remains reliable against attackers deliberately crafting traffic to evade detection. Only by validating the CNN-LSTM across broader and more realistic scenarios can researchers confirm whether its exceptional results represent true generalization rather than dataset-specific overfitting.

CONCLUSION AND RECOMMENDATIONS

Cloud computing security faces a significant threat from DDoS attacks which disrupt important services while leading to operational costs. Studies indicate that DDoS attack detection and mitigation excel with ML methods, particularly through CNN-LSTM implementations. The CNN-LSTM model demonstrated exceptional reliability for DDoS attack classification by maintaining a 99.9 percent accuracy rate and achieving high precision rates and recall scores and F1-scores. The test results against the ID3 algorithm demonstrate that CNN-LSTM shows better performance at analyzing intricate attack patterns. The study results demonstrate why advanced ML models remain essential for enhancing cybersecurity by enhancing threat detection inside cloud platforms and defending against current cyber threats.

Additional research must conduct studies to find improved hybrid machine learning detection methods as well as create real-time systems adaptive to shifting DDoS attack threats. The generalizability of solutions will improve through expanding attack scenario datasets as well as performing tests across multiple cloud deployments. The integration of blockchain systems for

secure logging and cost-efficient DDoS mitigation strategies would strengthen cloud ecosystem defensive abilities against DDoS attacks.

While the results are promising, this study has certain limitations. First, the experiments were conducted using the CIC-DDoS2019 dataset, which, although comprehensive, may not fully capture the diversity and evolving nature of real-world cloud DDoS attack traffic. Second, the evaluation was limited to a simulated offline environment, meaning the models were not tested under live, large-scale cloud deployments where network conditions and adversarial behaviors can differ significantly. Finally, only two algorithms CNN-LSTM and ID3 were compared, which restricts the generalizability of conclusions across the broader spectrum of machine learning methods.

To address these limitations, future research should focus on more realistic and practical extensions. One promising direction is the development of online detection systems capable of analyzing traffic in real time, ensuring faster response to active threats. Another avenue is to incorporate adversarial robustness techniques to make detection models more resilient against evasion attempts by attackers. Transfer learning approaches could also be explored to improve generalization across different cloud environments and varying attack scenarios, reducing the dependency on a single dataset. Additionally, testing hybrid models across multiple cloud deployments and expanding datasets with diverse, up-to-date attack scenarios will help validate scalability and adaptability.

REFERENCES

- [1] N. Z. Bawany, J. A. Shamsi, and K. Salah, "DDoS attack detection and mitigation using SDN: methods, practices, and solutions," *Arab. J. Sci. Eng.*, vol. 42, pp. 425–441, 2017.
- [2] N. Agrawal and S. Tapaswi, "Defense mechanisms against DDoS attacks in a cloud computing environment: State-of-the-art and research challenges," *IEEE Commun. Surv. & Tutorials*, vol. 21, no. 4, pp. 3769–3795, 2019.
- [3] A. Kushwaha, P. Pathak, and S. Gupta, "Review of optimize load balancing algorithms in cloud," *Int. J. Distrib. Cloud Comput.*, vol. 4, no. 2, pp. 1–9, 2016.
- [4] M. Darwish, A. Ouda, and L. F. Capretz, "Cloud-based DDoS attacks and defenses," in *International Conference on Information Society, i-Society 2013*, 2013.
- [5] M. Zekri, S. El Kafhali, N. Aboutabit, and Y. Saadi, "DDoS attack detection using machine learning techniques in cloud computing environments," in *2017 3rd international conference of cloud computing technologies and applications (CloudTech)*, 2017, pp. 1–7.
- [6] M. Idhammad, K. Afdel, and M. Belouch, "Semi-supervised machine learning approach for DDoS detection," *Appl. Intell.*, vol. 48, no. 10, pp. 3193–3208, 2018, doi: 10.1007/s10489-018-1141-2.
- [7] P. Khuphiran, P. Leelaprute, P. Uthayopas, K. Ichikawa, and W. Watanakesuntorn, "Performance Comparison of Machine Learning Models for DDoS Attacks Detection," in *2018 22nd International Computer Science and Engineering Conference (ICSEC)*, 2018, pp. 1–4. doi: 10.1109/ICSEC.2018.8712757.
- [8] Y. Li and Y. Lu, "LSTM-BA: DDoS Detection Approach Combining LSTM and Bayes," in *2019 Seventh International Conference on Advanced Cloud and Big Data (CBD)*, 2019, pp. 180–185. doi: 10.1109/CBD.2019.00041.
- [9] R. Umar, M. Olalere, I. Idris, R. A. Egigogo, and G. Bolarin, "Performance Evaluation of Machine Learning Algorithms for Hypertext Transfer Protocol Distributed Denial of Service Intrusion Detection," in *2019 15th International Conference on Electronics, Computer and Computation (ICECCO)*, 2019, pp. 1–7. doi: 10.1109/ICECCO48375.2019.9043262.
- [10] C. L. Calvert and T. M. Khoshgoftaar, "Impact of class distribution on the detection of slow HTTP DoS attacks using Big Data," *J. Big Data*, vol. 6, no. 1, p. 67, 2019, doi: 10.1186/s40537-019-0230-3.
- [11] H. Thanh and T. Lang, "Use the ensemble methods when detecting DoS attacks in Network Intrusion Detection Systems," *EAI Endorsed Trans. Context. Syst. Appl.*, 2019, doi: 10.4108/eai.29-11-2019.163484.
- [12] M. Ahmed and A.-S. K. Pathan, "Investigating Deep Learning for Collective Anomaly Detection - An Experimental Study," in *Security in Computing and Communications*, S. M. Thampi, S. Madria, G. Wang, D. B. Rawat, and J. M. Alcaraz Calero, Eds., Singapore: Springer Singapore, 2019, pp. 211–219.
- [13] T. Ahmad and M. N. Aziz, "Data preprocessing and feature selection for machine learning intrusion detection systems," *ICIC Express Lett*, vol. 13, no. 2, pp. 93–101, 2019.
- [14] O. E. Elejla, B. Belaton, M. Anbar, and A. Alnajjar, "Intrusion detection systems of ICMPv6-based DDoS attacks," *Neural Comput. Appl.*, vol. 30, pp. 45–56, 2018.

- [15] E. Shao, "Encoding IP address as a feature for network intrusion detection," Purdue University, 2019.
- [16] A. A. Abdulrahman and M. K. Ibrahim, "Evaluation of DDoS attacks detection in a CICIDS2017 dataset based on classification algorithms," *Iraqi J. Inf. Commun. Technol.*, vol. 1, no. 3, 2018.
- [17] V. S. Mohan, R. Vinayakumar, K. P. Soman, and P. Poornachandran, "Spoof net: syntactic patterns for identification of ominous online factors," in *2018 IEEE Security and Privacy Workshops (SPW)*, 2018, pp. 258–263.
- [18] A. Sanmorino, "A study for DDOS attack classification method," in *Journal of Physics: Conference Series*, 2019, p. 12025.
- [19] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy," in *2019 international carnahan conference on security technology (ICCST)*, 2019, pp. 1–8.
- [20] Routhu, K., Bodepudi, V., Jha, K. M., & Chinta, P. C. R. (2020). A Deep Learning Architectures for Enhancing Cyber Security Protocols in Big Data Integrated ERP Systems. *Available at SSRN 5102662*.
- [21] Chinta, P. C. R., & Katnapally, N. (2021). Neural Network-Based Risk Assessment for Cybersecurity in Big Data-Oriented ERP Infrastructures. *Neural Network-Based Risk Assessment for Cybersecurity in Big Data-Oriented ERP Infrastructures*.
- [22] Katnapally, N., Chinta, P. C. R., Routhu, K. K., Velaga, V., Bodepudi, V., & Karaka, L. M. (2021). Leveraging Big Data Analytics and Machine Learning Techniques for Sentiment Analysis of Amazon Product Reviews in Business Insights. *American Journal of Computing and Engineering*, 4(2), 35-51.
- [23] Karaka, L. M. (2021). Optimising Product Enhancements Strategic Approaches to Managing Complexity. *Available at SSRN 5147875*.
- [24] Chinta, P. C. R., & Karaka, L. M. AGENTIC AI AND REINFORCEMENT LEARNING: TOWARDS MORE AUTONOMOUS AND ADAPTIVE AI SYSTEMS.
- [25] Boppana, S. B., Moore, C. S., Bodepudi, V., Jha, K. M., Maka, S. R., & Sadaram, G. AI And ML Applications In Big Data Analytics: Transforming ERP Security Models For Modern Enterprises.
- [26] Chinta, P. C. R., Katnapally, N., Ja, K., Bodepudi, V., Babu, S., & Boppana, M. S. (2022). Exploring the role of neural networks in big data-driven ERP systems for proactive cybersecurity management. *Kurdish Studies*.
- [27] Chinta, P. C. R. (2022). Enhancing Supply Chain Efficiency and Performance Through ERP Optimisation Strategies. *Journal of Artificial Intelligence & Cloud Computing*, 1(4), 10-47363.
- [28] Sadaram, G., Sakuru, M., Karaka, L. M., Reddy, M. S., Bodepudi, V., Boppana, S. B., & Maka, S. R. (2022). Internet of Things (IoT) Cybersecurity Enhancement through Artificial Intelligence: A Study on Intrusion Detection Systems. *Universal Library of Engineering Technology*, (2022).
- [29] Moore, C. (2023). AI-powered big data and ERP systems for autonomous detection of cybersecurity vulnerabilities. *Nanotechnology Perceptions*, 19, 46-64.

- [30] Chinta, P. C. R. (2023). The Art of Business Analysis in Information Management Projects: Best Practices and Insights. *DOI, 10*.
- [31] Chinta, P. C. R. (2023). Leveraging Machine Learning Techniques for Predictive Analysis in Merger and Acquisition (M&A). *Journal of Artificial Intelligence and Big Data*, 3(1), 10-31586.
- [32] Krishna Madhav, J., Varun, B., Niharika, K., Srinivasa Rao, M., & Laxmana Murthy, K. (2023). Optimising Sales Forecasts in ERP Systems Using Machine Learning and Predictive Analytics. *J Contemp Edu Theo Artific Intel: JCETAI-104*.
- [33] Maka, S. R. (2023). Understanding the Fundamentals of Digital Transformation in Financial Services: Drivers and Strategic Insights. *Available at SSRN 5116707*.
- [34] Routhu, KishanKumar & Katnapally, Niharika & Sakuru, Manikanth. (2023). Machine Learning for Cyber Defense: A Comparative Analysis of Supervised and Unsupervised Learning Approaches. *Journal for ReAttach Therapy and Developmental Diversities*. 6. 10.53555/jrtdd.v6i10s(2).3481.
- [35] Chinta, Purna Chandra Rao & Moore, Chethan Sriharsha. (2023). Cloud-Based AI and Big Data Analytics for Real-Time Business Decision-Making. 36. 96-123. 10.47363/JAICC/2023.
- [36] Krishna Madhav, J., Varun, B., Niharika, K., Srinivasa Rao, M., & Laxmana Murthy, K. (2023). Optimising Sales Forecasts in ERP Systems Using Machine Learning and Predictive Analytics. *J Contemp Edu Theo Artific Intel: JCETAI-104*.
- [37] Bodepudi, V. (2023). Understanding the Fundamentals of Digital Transformation in Financial Services: Drivers and Strategic Insights. *Journal of Artificial Intelligence and Big Data*, 3(1), 10-31586.
- [38] Jha, K. M., Bodepudi, V., Boppana, S. B., Katnapally, N., Maka, S. R., & Sakuru, M. Deep Learning-Enabled Big Data Analytics for Cybersecurity Threat Detection in ERP Ecosystems.
- [39] Krutthika H. K. & A.R. Aswatha. (2021). Implementation and analysis of congestion prevention and fault tolerance in network on chip. *Journal of Tianjin University Science and Technology*, 54(11), 213–231. <https://doi.org/10.5281/zenodo.5746712>
- [40] Krutthika H. K. & A.R. Aswatha. (2020). FPGA-based design and architecture of network-on-chip router for efficient data propagation. *IIOAB Journal*, 11(S2), 7–25.
- [41] Krutthika H. K. & A.R. Aswatha (2020). Design of efficient FSM-based 3D network-on-chip architecture. *International Journal of Engineering Trends and Technology*, 68(10), 67–73. <https://doi.org/10.14445/22315381/IJETT-V68I10P212>
- [42] Krutthika H. K. & Rajashekhara R. (2019). Network-on-chip: A survey on router design and algorithms. *International Journal of Recent Technology and Engineering*, 7(6), 1687–1691. <https://doi.org/10.35940/ijrte.F2131.037619> (53 citations) (Now it is 17)
- [43] S. Ajay, et al., & Krutthika H. K. (2018). Source hotspot management in a mesh network-on-chip. *22nd International Symposium on VLSI Design and Test (VDAT-2018)*. https://doi.org/10.1007/978-981-13-5950-7_51
- [44] Kuraku, D. S., Kalla, D., Smith, N., & Samaah, F. (2023). Exploring how user behavior shapes cybersecurity awareness in the face of phishing attacks. *International Journal of Computer Trends and Technology*.

- [45] Kuraku, D. S., & Kalla, D. (2023). Impact of phishing on users with different online browsing hours and spending habits. *International Journal of Advanced Research in Computer and Communication Engineering*, 12(10).
- [46] Dinesh, K. (2022). Navigating the link between internet user attitudes and cybersecurity awareness in the era of phishing challenges. *International Advanced Research Journal in Science, Engineering and Technology*.
- [47] Kalla, D., & Samaah, F. (2023). Exploring Artificial Intelligence And Data-Driven Techniques For Anomaly Detection In Cloud Security. *Available at SSRN 5045491*.

License

Copyright (c) 2024 Navya Vattikonda, Anuj Kumar Gupta, Achuthananda Reddy Polu, Bhumeeka Narra, Dheeraj Varun Kumar Reddy Buddula, Hari Hara Sudheer Patchipulusu



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

Authors retain copyright and grant the journal right of first publication with the work simultaneously licensed under a [Creative Commons Attribution \(CC-BY\) 4.0 License](https://creativecommons.org/licenses/by/4.0/) that allows others to share the work with an acknowledgment of the work's authorship and initial publication in this journal.