International Journal of AI, BigData, Computational and Management Studies
Noble Scholar Research Group | Volume 3, Issue 4, PP. 49-59, 2022
ISSN: 3050-9416 | https://doi.org/10.63282/3050-9416.IJAIBDCMS-V3I4P106

Empowering Cloud Security with Artificial Intelligence: Detecting Threats Using Advanced Machine learning Technologies

Varun Bitkuri[1], Raghuvaran Kendyala[2], Jagan Kurma[3], Jaya Vardhani Mamidala[4], Sunil Jacob Enokkaren[5], Avinash Attipalli[6]
[1]Stratford University ,Software Engineer.
[2]University of Illinois at Springfield, Department of Computer Science.
[3]Christian Brothers University, Computer Information Systems.
[4]University of Central Missouri, Department of Computer Science.
[5]ADP, Solution Architect.
[6]University of Bridgeport, Department of Computer Science.

Abstract: The high growth rate of cloud computing where protection of digital assets due to the nature of the threat is of paramount importance especially in reducing cyber threats such as Distributed Denial-of-Service (DDoS) attacks. In the paper, I have suggested an intelligent framework of threat detection based on the Random Forest (RF) model to improve cloud security. This model was trained and also tested on the CIC-DDoS2019 dataset, with the capabilities of ensemble learning used to attain high levels of classification. In the experiment, the RF model attained an 99.97% accuracy with precision, recall of 99.97%, and an F1-score of 99.98 %. The better performance of the proposed approach is proven by making a comparative analysis with other models, Gradient Boosting (96.7%), Logistic Regression (95.0%), Support Vector Machine (94.32%). The model robustness is also confirmed by ROC curves, confusion matrix analysis and training-validation trends. Such results define Random Forest as an exceptionally efficient and predictable framework when it comes to countering changing cyber risks in cloud environments, with promise of both scalability and real-time effectiveness when implemented into practice.

Keywords: Cloud Security, DDoS Detection, Random Forest, Machine Learning, CICDDoS2019 Dataset, Threat Detection, Cybersecurity.

1. Introduction

The game has changed with cloud computing offering businesses and organizations an infrastructure that is adaptable, affordable and expandable. The cloud infrastructure has noted major increase in productivity and operational agility whereby organizations can implement high-quality applications, services provision through Internet with very low investment cost up-front [1] but a disturbing factor is that alongside the mentioned benefits is the aspect of cloud data security and services security. The cloud landscape has a large amount of sensitive data, and as a result of shared and, in many cases, more complicated infrastructure, there is a high level of opportunity at the disposal of a potential cybercriminal [2][3]. Traditional security controls are always rule-driven and usually reactive and thus it would not be able to keep up with the pace and the level of intricacy you can find in a cloud environment [4][5].

These new threats, which are characterized as being highly sophisticated (i.e., hard to detect using traditional tools like firewalls, antivirus software, and simple encryption techniques although, as is shown in the context of current attacks and their countermeasures, these basic tools have yet to become obsolete), such as zero-days, advanced persistent threats (APTs), and insider threats are basically unrecognizable with traditional tools, due to the fact they were previously undetectable using legacy security measures such as firewalls and anti-virus software, along with basic forms of encryption [6]. a security measure that Traffic volume of cloud and event logging is also too voluminous even to be tracked by human and too dynamic to be picked by rule-based systems in real-time [7]. To overcome such limitations, numerous cloud security products are currently including Artificial Intelligence (AI) and ML functionalities in their products.

The application of AI and ML has become an important resource within cloud security. Their inclusion in cybersecurity architectures is a momentous change [8], that allows systems to be more responsive, intelligent and autonomous in noting threats and mitigating their effects, and are fantastic at operating large data sets, spotting anomalies and deriving threats with much more accuracy and quickness than before [9]. Using behaviour analysis, adaptive learning, anomaly detection and real-time threat detection, these technologies enhance cloud security [10]. AI and ML are capable of effectively increasing the resilience of clouds as it allow creating automatic threat detectors and response systems. The combination of cloud computing and the most advanced machine learning technologies is a hard-to-beat approach to securing digital assets in a rapidly hostile cyber environment. This comes in help especially in detecting previously unknown or zero-day attacks before they cause any serious damage.

1.1. Motivation and Contribution

As cloud computing keeps changing the way data is stored and provided, it is also creating severe security risks such as DoS and especially DDoS attacks. Conventional security systems cannot match the pace of these threats that are changing with every movement, scale, and complexity. The study is inspired by the increase in the need to have smarter, adaptive, and data-driven security systems that can identify threats early and with a lot of accuracy. AI, and especially ML offer a robust basis to automate the process of threat detection, learning data patterns, and improving the overall resilience of clouds in general. This study will aim to develop an effective detecting system capable of identifying complex attack structures and responding to cloud environments faster using machine learning models. The major contributions associated with this study include:

- Employing CICDDoS2019 dataset, which is a realistic and rich dataset, consisting of different DDoS attack vectors, to develop a threat detection framework.
- Performing systematic data pre-processing, including data cleaning, feature encoding, feature scaling, feature selection, and handling class imbalance to improve model quality and generalizability.
- Implementing a Random Forest classifier, known for its interpretability and high performance in classification tasks, to detect potential security threats.
- Demonstrated model convergence and generalization through training/validation accuracy and loss plots, indicating minimal overfitting.
- Testing the model based on typical measures of performance- F1-score, recall, accuracy, and precision to present a complete overview of the effectiveness of the model.

1.2. Novelty of the Paper

The novelty lies in the integration of an RF classifier with a rigorously optimized pre-processing pipeline, including ADASYN-based resampling and hybrid Genetic-Grasshopper feature selection, tailored for high-dimensional DDoS detection. This approach ensures balanced learning, minimizes model complexity, and enhances interpretability. By leveraging the CICDDoS2019 dataset and refining the input representation, the study presents a threat detection method that is both highly accurate and broadly applicable, outperforming traditional models and providing a scalable solution for real-world cloud security challenges.

1.3. Organization of the Paper

The outline of the paper is as follows In Section II, the relevant literature on cloud threat detection using AI is reviewed. Section III describes the approach and ML models that were utilised. Experimental results and evaluations are presented in Section IV. Section V provides a summary, a discussion of limits, and suggestions for future research.

2. Literature Review

This section discusses the most recent developments in cloud security by employing the use of AI, or specifically ML technologies that have been designed to improve threat detection, vulnerability identification, and more accurately predict threats, and respond in real time to security incidents. Some of the major works consulted are as follows:

Bharati and Tamane (2020) the capacity to detect SQL injection in cloud-based apps using machine learning protocols. The research is carried out by training classifiers to distinguish between harmful and benign payloads, using sets of both types of data. Its findings, which showed a detection rate of over 98%, show that ML is crucial in data protection and defence against SQL injections [11]. Tripathy, Gohil and Halabi (2020) There has been research into the possibility of using machine learning to identify SQL injection in software as a service applications. The work is meant to separate malicious and non-malicious target payloads, by going off classifiers trained on divergent malicious and non-malicious payloads. The results show that machine learning has a detection rate of over 98%, making it crucial for data protection and defence against SQL injection. This research also proves that various ML models may be compared for their efficacy in identifying SQL injection attacks [12].

Abusitta et al. (2019) introduce a collaborative IDS, which is one of the machine learning-based strategies, and incorporates a Denoising Autoencoder (DA) in order to effectively use the records of the past feedback to enable proactive decisions. After being developed and trained on a real-life dataset, the model demonstrated the highest accuracy of 95% in the GPU-enabled TensorFlow. Through this, inefficiencies and time loss incorporated in cooperative IDS policies are eliminated and, therefore, the policies are more feasible and accountable in identifying advanced attacks [13]. Garg et al. (2019) utilise a hybrid strategy combining convolutional neural networks (CNNs) and grey wolf optimisation (GWO) to identify anomalies in networks. Different from one another, the Improved-GWO (ImGWO) and Improved-CNN (ImCNN) models are best suited for different tasks: classification and selection, respectively. Additionally, compared to other state-of-the-art models, this one has the best accuracy (3.25%), detection rate (4.08%), FPR (3.62%), and F-score (8.52%) [14].

Parampottupadam and Moldovann (2018) investigated network intrusion detection using DL in real-time. They built a cloud-based system that uses binomial as its basis for model proof of method. The study found that appropriate selection of deep learning library was critical in real-time applications after comparing the H2O and DeepLearning4J libraries with other ML models. On the NSL-KDD training dataset, H2O models achieved an accuracy of over 99.5%, while on the testing dataset, they

only managed 83% accuracy [15]. Gao et al. (2018) introduce a robotic system that operates in the cloud that can identify network intrusions using an ensemble technique that combines fuzzy logic with semi-supervised fuzzy learning. The method constructs an ensemble-based system using supervised and unsupervised parts; it learns from labelled data and employs an analytical methodology based on fog. On the NSL-KDD dataset, the technique outperforms the state-of-the-art model with an accuracy of 84.54% and 71.29%, respectively. This approach gets rid of the issue of finding the latest attack patterns and data security concerns with cloud-based robotic systems [16].

Table I gives an overview summary of the main work on optimizing cloud security systems on the basis of artificial intelligence, describes the major contributions in each of the employed models, gives out the limitations of these works, and indicates where future research in the area of intelligent cloud threat arbitration is expected to head.

Table 1: Summary of Recent Studies Based on Cloud Security and Threat Detection

References	Approach	Dataset	Main Contributions	Limitations	Future Work
Bharati and Tamane (2020)	Machine Learning-based IDS using Random Forest classifier to detect anomalies	CSE-CIC-IDS-2018	Developed an ML-based anomaly detection system	Bharati and Tamane, 2020	Machine Learning-based IDS using Random Forest classifier to detect anomalies rather than signature-based misuse detection
Tripathy, Gohil and Halabi (2020)	ML-based classifiers for SQL Injection Detection	Malicious & benign SQL payloads	High accuracy (>98%) in detecting SQL injection at app level	Focuses solely on SQL injection; lacks scalability analysis	Extend to other web-based attacks, test in multi-tenant SaaS environments
Abusitta et al. (2019)	Denoising Autoencoder in Cooperative IDS	Real-life IDS feedback dataset	Enables proactive decision-making with partial IDS feedback, 95% detection accuracy	Delay in aggregation, partial feedback dependency	Incorporate reinforcement learning, real-time feedback prediction optimization
Garg et al. (2019)	Hybrid: Improved-GWO + CNN (ImGWO-ImCNN)	DARPA'98, KDD'99, Synthetic	Improved anomaly detection rate, accuracy, and F-score via hybrid optimization	Evaluation limited to benchmark datasets, needs validation in real-world network traffic	Apply model to dynamic cloud environments, reduce computational complexity
Parampottupadam and Moldovann (2018)	developed a cloud-based prototype for real-time binomial and multinomial classification utilizing deep learning.	NSL-KDD — Improved version of KDDCUP-99	Demonstrated high performance of deep learning models, especially H2O-based (≥99.5% accuracy on training data)	Accuracy dropped to ~83% on test data, indicating generalization issues	Extend evaluation to real-time cloud traffic data, study the scalability of deep learning libraries in production environments
Gao et al. (2018)	Fuzziness-based semi-supervised ensemble learning	NSL-KDD (KDDTest+, KDDTest-21)	Combines strengths of supervised and unsupervised NIDS models, removes noisy samples and achieves Accuracy: 84.54% (KDDTest+), 71.29% (KDDTest-21)	Accuracy on advanced test sets relatively low (71.29%)	Improve generalization to zero-day attacks, test with evolving cloud robotic systems

3. Methodology

The research methodology will start with the data pre-processing, during which any inconsistencies in data are cleaned up. The numerical values of the categorical values are transformed into numerical format by feature encoding. The feature distributions are then scaled to achieve uniformity. They select features that are relevant to their problem, and then they employ resampling techniques to fix the data imbalance while keeping important qualities. It is from this pre-processed dataset that the training and testing sets are constructed. They utilise an RF classifier because it excels at dealing with high-dimensional data and resilient nonlinear interactions. The model is evaluated using the test set following its training on the training set. When evaluating performance, standard metrics like recall, accuracy, precision, and F1-score are utilised. Figure 1 shows how artificial intelligence may strengthen cybersecurity by ensuring effective and dependable DDoS attack detection in cloud settings through a methodical process.

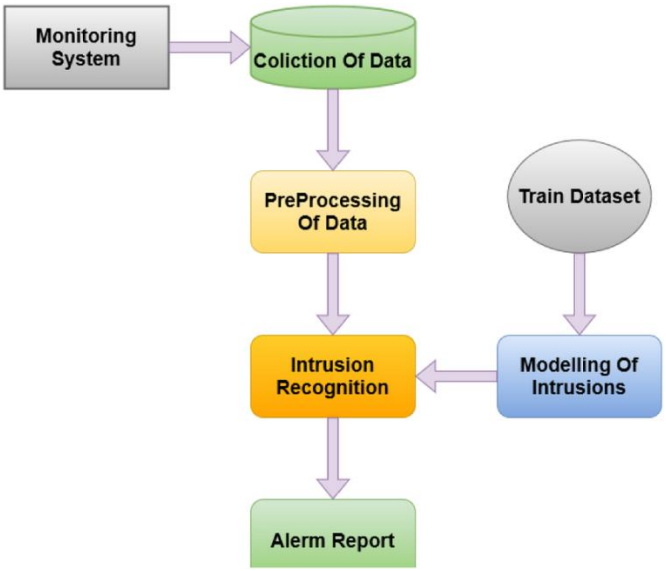

Figure 1: Proposed Flowchart for Threat Detection

3.1. Dataset Description

The Canadian Institute for Cybersecurity created the CICDDoS2019 dataset to train algorithms for identifying contemporary DDoS attacks. It includes both benign traffic and recent attack types conducted via TCP/UDP-based protocols, offering a realistic representation of real-world scenarios. The attacks are categorized into reflection-based (e.g., TFTP, LDAP, DNS) and exploitation-based (e.g., SYN, UDP, WebDDoS), totaling twelve classes and 88 features. This classification helps in analyzing and modeling different DDoS behaviors effectively. The following is the class distribution by attack category in Figure 2 of the dataset.

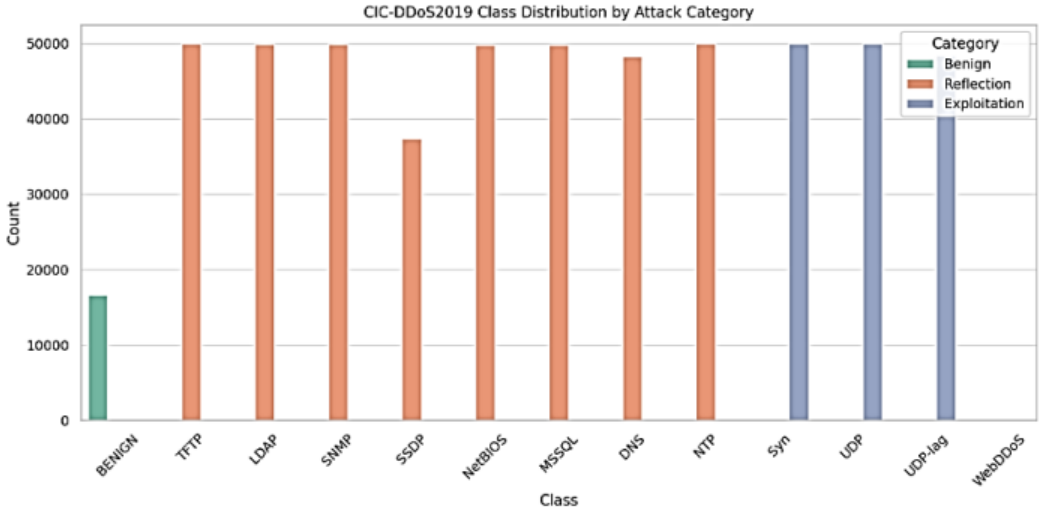

Figure 2: Class Distribution By Attack Category of CIC-Ddos2019 Data

In Figure 2, the x-axis lists various network protocols or attack types such as TFTP, LDAP, DNS, and UDP, alongside 'Benign' traffic. The y-axis represents the count. Most categories, especially those related to 'Reflection' and 'Exploitation' attacks, show high counts approaching 50,000, while 'Benign' traffic is considerably lower, around 16,000. This visualization effectively displays the distribution and prevalence of various attack types within the dataset.

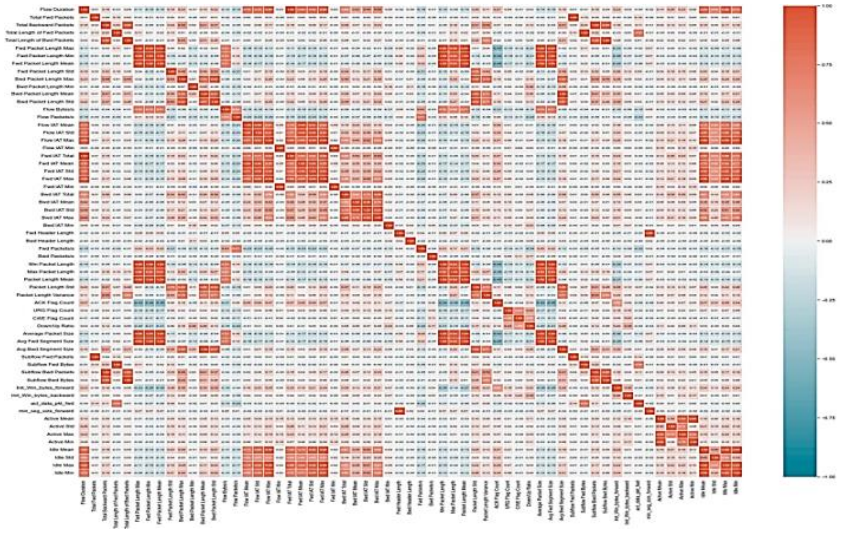

Figure 3: Correlation Heatmap of CIC-DDoS2019 Data

Figure 3 is a correlation heatmap, visually representing the relationships between numerous variables. Different shades of orange and teal indicate the strength and direction (positive or negative) of correlations, respectively. Darker colors signify stronger relationships. The diagonal line shows perfect self-correlation.

3.2. Data Preprocessing

Data pre-processing involves transforming raw data into a more manageable format for data science tasks like data mining and machine learning. For feature selection to work, machine learning algorithms require clean, numerical inputs, which is why data pre-processing is covered in this section. In order to do this, the following steps are taken:

- **Data Cleaning:** It involves finding and fixing inaccurate, lacking, irrelevant, or incomplete data by substituting the mean of the related attribute for missing or NaN values. A large amount of work goes into making sure the input data is accurate and clean because bad data can lead to biased results, increased error rates and less accuracy in a model.

3.3. Feature Encoding

There are several columns in the input datasets that have both category and numerical values. Feature encoding is the process of providing numerical representations to data that is not numerical in nature. Although each has advantages and disadvantages, one-hot encoding and label encoding are two common methods for doing this. Though it may be more effective, one-hot encoding greatly increases the complexity of the features. Label encoding was chosen for this study since it worked well [17]. Combining one-hot encoding with label encoding would provide more features and make the dataset bigger, which would use more resources. Label encoding alone giving each unique text of a feature a unique number starting at 0 works better because reducing the number of characteristics is the aim.

3.4. Feature Scaling

Scaling the dataset's characteristics will help to prevent any one feature from having an excessive impact on the model because of its size, particularly considering how diverse the dataset is. Mini-Max In order to standardise the characteristics to a certain range, scaling was typically utilised. Here, each attribute was standardised by removing its lowest value and dividing by the range to ensure that each contributed equally to the final projection [18]. Parameters that need to be inside a certain range can benefit from this method, which is commonly employed when the distribution is non-Gaussian. The min-max scaling was represented in Equation (1)

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \qquad (1)$$

Where, respectively, x_{min} and x_{max}at the feature's minimum and maximum values.

3.5. Feature Selection

Feature selection aims to increase classification performance by identifying the best features, cutting down on training time, and improving accuracy. They do this by using a fitness function that is optimised using a hybrid Genetic-Grasshopper algorithm, which strikes a compromise between maximising accuracy and minimising characteristics. The Grasshopper Optimization Algorithm (GOA) mimics swarm behaviour for efficient global search, while the Genetic Algorithm (GA) applies evolutionary operators to refine solutions. Together, they effectively select feature subsets that enhance classification accuracy with fewer features. Equation (2) represents the calculation of feature selection:

$$F = max\left(Acc + w_f\left(1 - \frac{F_s}{F_t}\right)\right) \qquad (2)$$

Where the length of the chosen features is denoted by F_s, the accuracy by Acc, the weight factor by w_f , and the total number of features by F_t. The weight factor's value being near to one indicates that both the accuracy improvement and feature minimisation objectives are seen as being equally significant.

3.6. Handling Data Imbalance

Network intrusion detection is just one of many real-world machine learning applications where imbalanced datasets pose a big difficulty. Classifier performance may suffer as a result of imbalance, particularly in extremely imbalanced datasets involving minority threat categories. To get around this obstacle, two common methods are used: oversampling. The oversampling method replicates instances or generates synthetic samples until all minority classes are represented equally. ADASYN, or Adaptive Synthetic Sampling Approach, addresses imbalance in machine learning datasets by focusing on the minority class [19]. ADASYN uses k-nearest neighbor technique to produce synthetic data points and analyses minority class instance density. The synthetic data generation mechanism prioritizes instances in low-density zones, showing minority under-representation. Thus, giving the model a more balanced representation improves classification performance and helps it learn from under-represented cases.

3.7. Data Splitting

In this analysis, the datasets are split into two separate sections with a ratio of 75/25. While 75% of the data is used for model training, 25% is reserved for performance testing and verification.

3.8. Propose Random Forest Classifier (RFC)

The RFC is a well-proven ensemble prediction method that employs many decision trees to arrive at its final prediction. It has demonstrated effectiveness in various regression and classification scenarios. When building the decision tree, randomly selecting data nodes also improves the classifier's overall performance. Success in classification is largely influenced by the overall number of trees and leaves, since the decision tree partitions the feature space into L regions representing RL for a total of L leaves. [20]. Decision trees employ this feature space to predict their final output, which can be expressed mathematically as Equations (3) and (4). The majority vote of the trees determines the ultimate expected outcome. Since tweaking yields optimal performance in the evaluation step, the total number of leaves and trees are the two most important RFC hyperparameters [21]. Careful consideration should be given to the selection of these parameters, as beyond a certain point, further increases in their values further add computing complexity.

$$f(x) = \sum_{i=1}^{L} constant_L * \prod(x, R_L) \qquad (3)$$

$$\prod(x, R_L) = \begin{cases} 1 & if\ x \in R_L \\ 0 & otherwise \end{cases} \qquad (4)$$

In which the input space is partitioned into L parts R_L, Each one of which is related to a fixed value. This measure is then predicted as a sum of these constants times an indicator function $\prod(x, R_L)$ = 1 explanantie x is inside a region R_L and 0 otherwise.

3.9. Performance Measurement Parameters

This section explains the principal metrics that help to evaluate the activity of the model. Such measures will permit meaningful comparison and confirmation of results in providing information concerning the accuracy of the model, consistency of the model, and overall efficacy of the model. It is needed to define main terms before demonstrating the metrics of evaluation when introducing it:

- **TP (True Positive):** The quantity of information that is useful in terms of its accuracy and possible applications.
- **FP (False Positive):** An amount of data with both positive and negative predicted values.
- **FN (False Negative):** A large number of data that have negative actual measure along with a positive expected measure.
- **TN (True Negative):** An amount of data with a negative actual value in addition to a negative anticipated value.

The performance of the model can be measured with the following metrics:

3.9.1. Accuracy

The usefulness of accuracy as an evaluation metric depends on the consistency of the datasets and the near-equalities of the false positive and false negative values [22]. A classifier's accuracy is determined by how effectively it can predict the data points, as Equation (5) illustrates.

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \quad (5)$$

3.9.2. Precision

The accuracy can be described as the proportion of predicted positive observations that genuinely materialised through accurate prediction of positive observations. The form in percentage terms is provided by Equation (6).

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

3.9.3. Recall

Equation. (7) shows "recall," which is the fraction of positive observations projected to be true out of all the real class observations.

$$Recall(Rc) = \frac{TP}{TP+FN} \quad (7)$$

3.9.4. F1-score

The F1-score, which is the harmonic mean of recall and precision, is a balanced assessment of the model's accuracy and reliability [23]. It provides a single measure that encompasses both characteristics and is specified in Equation (8) as:

$$F1\ score(F1) = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (8)$$

3.9.5. Receiver Operating Characteristic (ROC) Curve

The sensitivity and false positive rates at various thresholds can be seen graphically in a ROC curve. The model's overall performance may be measured by looking at the area under the curve (AUC), which is defined in Equation (9). A number closer to 1 implies higher performance.

$$\text{AUC} = \int_0^1 \text{TPR(FPR)d}(FPR) \quad (9)$$

TPR and FPR are used here.

3.9.6. Loss

The categorical cross-entropy loss function (in Equation 10), which lends itself to multi-class classification task of our problem. To reduce the loss, the optimizer Adam is applied to specify the parameters of the model by iteration and improve the results.

$$L = -\sum_{i=1}^{N} y_i\ log(p_i) \quad (10)$$

Where y_iRegarding the actual labels, and p_iHe labels predicted by the model.

4. Result Analysis and Discussion

This paper explores the use of leading-edge ML methods to strengthen cloud security by discovering threats due to the application of intelligent algorithms. Specifically, an RF model was employed to perform threat detection tasks, utilizing the ensemble learning feature to achieve improved classification results. The experimental evaluations were carried out using the Python language and the Google Colab Pro environment, which gives users access to a robust computing framework with 25 GB of RAM. Numerous libraries, including Scikit-learn, helped with the model's training and assessment. The RF model showed an exceptional accuracy rate of 99.97%, as shown in Table II, indicating its precision and reliability in identifying security concerns. With a 99.97 percent accuracy and a 99.97 percent recall, the model is very capable of identifying threats and has a very low FPR. The respective F1-score of 99.98 percent validates the well-balanced and extremely successful performance and the model proves suitable in the practical application of cloud security implementation in the real world.

Table 2: Performance Metrics of the Propose Model for Threat Detection.

Metrics	Random Forest
Accuracy	99.97
Precision	99.97
Recall	99.97
F1score	99.98

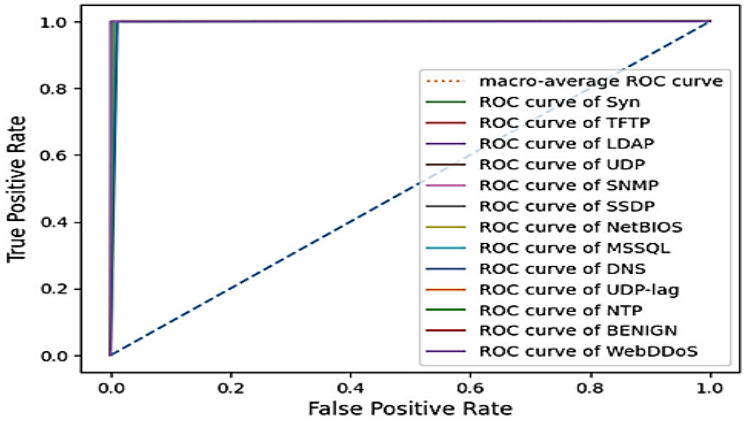

Figure 4: Multi-Class ROC Curves for Random Forest Model

The ROC curve in Figure 4 shows the TPR vs the FPR, with each line denoting the ROC curve for a particular traffic class (e.g., Syn, UDP, Web DDoS, Benign). Because most curves quickly approach the top-left corner, showing a high TPR with a low FPR, the plot shows the model's strong detection accuracy across numerous attack vectors. Moreover, the macro-average ROC curve indicates strong overall performance.

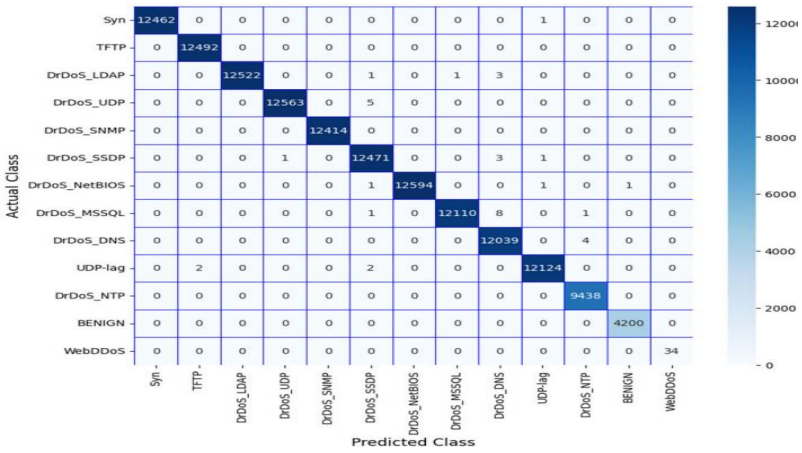

Figure 5: Multi-Class Confusion Matrix for Random Forest Model

Figure 5 shows the multi-class confusion matrix for the RF model, showcasing its strong classification performance across various DDoS attack types and benign traffic. The diagonal dominance of high values indicates accurate predictions for most classes, particularly for Syn, TFTP, DrDoS_LDAP, and DrDoS_SNMP, with minimal misclassifications. Notably, the BENIGN class also shows high precision with 4200 correct classifications. The sparse distribution of off-diagonal components indicates that the model is resilient in classifying multi-class network data and successfully differentiates between different types of attacks.

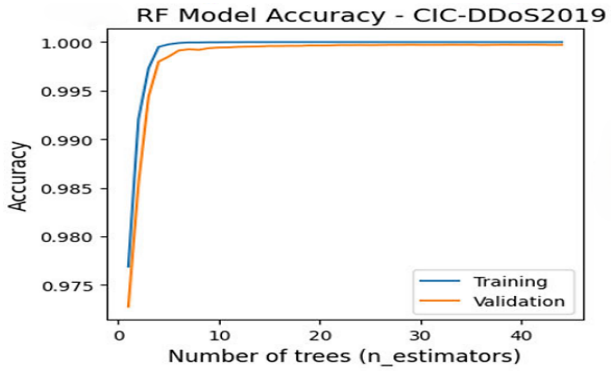

Figure 6: Training and Validation Accuracy for Random Forest

Figure 6 shows the relationship between the number of trees (n_estimators) and the RF model's training and validation accuracy on the CIC-DDoS2019 dataset. As the number of trees grows, the graph reveals that both accuracies are getting close to being perfect, reaching a level of about 10 trees. The model's ability to generalise to new data is enhanced when there is little variation between the training and validation curves, which indicates that overfitting has not occurred. Using the dataset, the RF model accurately and consistently identified the type of DDoS attack.

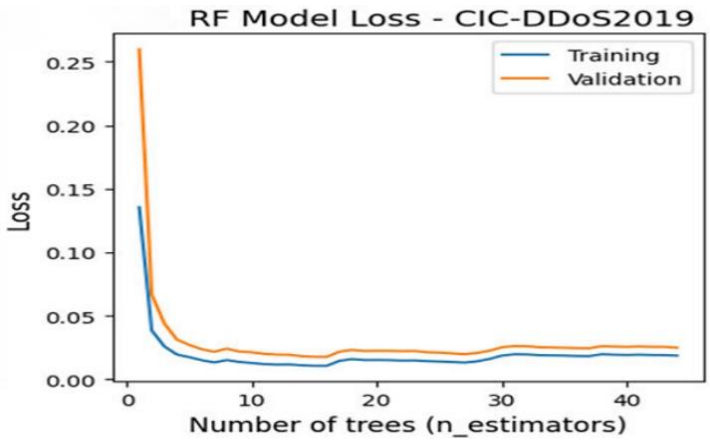

Figure 7: Training and Validation Loss Curve of Random Forest Model

Figure 7 illustrates that both loss curves have a sharp drop, and that as the tree count grows, the curves quickly level out. Overfitting is not an issue and the model does a good job of generalizing from the training data since the validation and training loss curves are so close to each other. This minimal loss means that the model is effective in learning and correctly detecting the security risks associated with clouds through sophisticated ML.

4.1. Comparative Analysis

Cloud security system risk identification using machine learning models is compared in this section. The RF model was clearly better than all other models, as shown by Table III, which shows that it had the maximum accuracy of 99.97%. This remarkable result highlights the high capability of RF in detecting the threat to the cloud accurately making it the most credible model to protect intelligent systems. GB, LR and SVM on the other hand achieved 96.7%, 95.0%, 94.32% accuracies respectively and could not come close to the high accuracy exhibited by the RF model.

Table 3: Accuracy Comparison of Machine Learning Models for Threat Detection

Models	Accuracy
Random Forest model	99.97
Gradient Boosting[24]	96.7
Logistic Regression[25]	95.0
Support Vector Machine[26]	94.32

Several distinct advantages pertain to the suggested RF-based model's ability to identify DDoS threats in cloud environments. Effective analysis with high-dimensional nonlinear data is supported by its ensemble learning architecture, which also ensures extremely accurate classification against different sorts of attacks. It is possible to employ the model to decrease FP replies and successfully identify threats, as evidenced by the near-perfect outcomes shown by the performance measures accuracy, precision, recall, and F1-score. The enhanced generalisation and multi-category differentiation skills, as seen in ROC curves and confusion matrices, are proof of this. There is a small issue with overfitting, but the model is very adaptable to real-world situations, as seen by the steady training and validation curves. With its constant performance outperforming other models including GB, LR, and SVM, the proposed RF model appears to be a reliable solution for intelligent cloud security systems.

5. Conclusion and Future Scope

Smart and adaptive security applications are essential in the wake of increasing cyberattacks on cloud systems. With the strength of ensemble learning, RF became one of the most effective models in Cloud environment to detect DDoS attacks. It showed excellent results as the accuracy rate was 99.97 percent signifying high strength and low misclassification. Its absolute advantage over the other options such as GB, LR and SVM, makes it a good choice to be considered in the actual use. This method provides

both detectability and interpretability and resilience in addition to being effective in detection, thus the method is suitable in dynamic and high-stakes cloud settings. In the future, work needs to be done to reduce computational overheads to implement edges in real time, to build on the continuous learning processes to be flexible and to look at hybrid models, which merge the feature extraction capabilities of deep learning with the decision clarity of RF. The further extension of the model in its ability to identify zero-day and cross-platform threats, integrating explainable AI to be more transparent, and adapting it to multi-cloud environments will also make it even more applicable. With cloud ecosystems becoming increasingly large and complex, the explainable of AI and cybersecurity is becoming central to the capability to keep up with more complex attack vectors. The areas of future work should be associated with the further development of the model's scalability, implementation of real-time adaptive feedback, validation using various datasets in the cloud, and generalization against zero-day attacks on dynamic multi-tenant clouds.

References

1. A. Qayyum *et al.*, "Securing Machine Learning in the Cloud: A Systematic Review of Cloud Machine Learning Security," *Front. Big Data*, vol. 3, Nov. 2020, doi: 10.3389/fdata 2020.587139.
2. M. Abubakar, S. C. G. Varma, H. Likki, H. Gp, H. S, and H. M. S, "Leveraging AI and Machine Learning for Enhanced Cloud Security and Performance," 2020.
3. Bindra and M. Sood, "Detecting DDoS attacks using machine learning techniques and contemporary intrusion detection dataset," *Autom. Control Comput. Sci.*, vol. 53, no. 5, pp. 419–428, 2019.
4. D. Bhamare, T. Salman, M. Samaka, A. Erbad, and R. Jain, "Feasibility of supervised machine learning for cloud security," in *2016 International Conference on Information Science and Security (ICISS)*, 2016, pp. 1–5.
5. S. I. Shyla and S. S. Sujatha, "Cloud security: LKM and optimal fuzzy system for intrusion detection in cloud environment," *J. Intell. Syst.*, vol. 29, no. 1, pp. 1626–1642, 2019.
6. S. Singamsetty, "Fuzzy-Optimized Lightweight Cyber-Attack Detection For Secure Edge-Based Iot Networks," vol. 6, no. 07, pp. 1028–1033, 2019, doi: 10.53555/jcr.v6.
7. A. Das, W.-M. Lin, and P. Rad, "A distributed secure machine-learning cloud architecture for semantic analysis," in *Applied Cloud Deep Semantic Recognition*, Auerbach Publications, 2018, pp. 131–159.
8. J. Hou, J. Qian, Y. Wang, X.-Y. Li, H. Du, and L. Chen, "ML Defense: Against Prediction API Threats in Cloud-Based Machine Learning Service," in *Proceedings of the International Symposium on Quality of Service*, New York, NY, USA: ACM, Jun. 2019, pp. 1–10. doi: 10.1145/3326285.3329042.
9. S. Oduri, "Integrating Ai Into Cloud Security : Future Trends And Technologies," 2019.
10. R. S. S. Kumar, A. Wicker, and M. Swann, "Practical machine learning for cloud intrusion detection: Challenges and the way forward," in *Proceedings of the 10th ACM workshop on artificial intelligence and security*, 2017, pp. 81–90.
11. M. P. Bharati and S. Tamane, "NIDS-Network Intrusion Detection System Based on Deep and Machine Learning Frameworks with CICIDS2018 using Cloud Computing," in *2020 International Conference on Smart Innovations in Design, Environment, Management, Planning and Computing (ICSIDEMPC)*, 2020, pp. 27–30. doi: 10.1109/ICSIDEMPC49020.2020.9299584.
12. D. Tripathy, R. Gohil, and T. Halabi, "Detecting SQL Injection Attacks in Cloud SaaS using Machine Learning," in *2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, 2020, pp. 145–150. doi: 10.1109/BigDataSecurity-HPSC-IDS49724.2020.00035.
13. A. Abusitta, M. Bellaiche, M. Dagenais, and T. Halabi, "A deep learning approach for proactive multi-cloud cooperative intrusion detection system," *Futur. Gener. Comput. Syst.*, vol. 98, pp. 308–318, Sep. 2019, doi: 10.1016/j.future.2019.03.043.
14. S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya, and R. Ranjan, "A Hybrid Deep Learning-Based Model for Anomaly Detection in Cloud Datacenter Networks," *IEEE Trans. Netw. Serv. Manag.*, vol. 16, no. 3, pp. 924–935, 2019, doi: 10.1109/TNSM.2019.2927886.
15. S. Parampottupadam and A.-N. Moldovann, "Cloud-based Real-time Network Intrusion Detection Using Deep Learning," in *2018 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, 2018, pp. 1–8. doi: 10.1109/CyberSecPODS.2018.8560674.
16. Y. Gao, Y. Liu, Y. Jin, J. Chen, and H. Wu, "A novel semi-supervised learning approach for network intrusion detection on cloud-based robotic system," *IEEE Access*, vol. 6, pp. 50927–50938, 2018.
17. A. Hameed and M. Suleman, "AI-Powered Anomaly Detection for Cloud Security: Leveraging Machine Learning and DSPM," 2019, doi: 10.13140/RG.2.2.23781.51685.
18. A. Majeed and N. Ahmad, "DSPM in Cloud Security: AI-Driven Anomaly Detection Using Machine Learning Models," 2019.
19. Z. Hu, L. Wang, L. Qi, Y. Li, and W. Yang, "A novel wireless network intrusion detection method based on adaptive synthetic sampling and an improved convolutional neural network," *IEEE access*, vol. 8, pp. 195741–195751, 2020.
20. M. Idhammad, K. Afdel, and M. Belouch, "Detection System of HTTP DDoS Attacks in a Cloud Environment Based on Information Theoretic Entropy and Random Forest," *Secur. Commun. Networks*, vol. 2018, no. 1, pp. 1–13, Jun. 2018, doi: 10.1155/2018/1263123.

21. A. Tesfahun and D. Lalitha Bhaskari, "Intrusion detection using random forests classifier with SMOTE and feature reduction," *Proc. - 2013 Int. Conf. Cloud Ubiquitous Comput. Emerg. Technol. CUBE 2013*, pp. 127–132, 2013, doi: 10.1109/CUBE.2013.31.
22. K. Kaur and V. Zandu, "A Secure Data Classification Model in Cloud Computing Using Machine Learning Approach," *Int. J. Grid Distrib. Comput.*, vol. 9, no. 8, pp. 13–22, Aug. 2016, doi: 10.14257/ijgdc.2016.9.8.02.
23. P. M. Khilar, V. Chaudhari, and R. R. Swain, "Trust-based access control in cloud computing using machine learning," in *Cloud Computing for Geospatial Big Data Analytics: Intelligent Edge, Fog and Mist Computing*, Springer, 2018, pp. 55–79.
24. I. Bolodurina, A. Shukhman, D. Parfenov, A. Zhigalov, and L. Zabrodina, "Investigation of the problem of classifying unbalanced datasets in identifying distributed denial of service attacks," *J. Phys. Conf. Ser.*, vol. 1679, no. 4, 2020, doi: 10.1088/1742-6596/1679/4/042020.
25. M. S. Elsayed, N.-A. Le-Khac, S. Dev, and A. D. Jurcut, "DDoSNet: A Deep-Learning Model for Detecting Network Attacks," in *2020 IEEE 21st International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, 2020, pp. 391–396. doi: 10.1109/WoWMoM49955.2020.00072.
26. P. G. O. Prakash, K. Sasirekha, and D. Vistro, "A DDOS prevention system designed using machine learning for cloud computing Environment," *Int. J. Manag.*, vol. 11, no. 10, pp. 1797–1806, 2020, doi: 10.34218/IJM.11.10.2020.167.
27. Polu, A. R., Vattikonda, N., Buddula, D. V. K. R., Narra, B., Patchipulusu, H., & Gupta, A. (2021). Integrating AI-Based Sentiment Analysis with Social Media Data for Enhanced Marketing Insights. *Available at SSRN 5266555*.
28. Katari, A., & Kalla, D. (2021). Cost Optimization in Cloud-Based Financial Data Lakes: Techniques and Case Studies. *ESP Journal of Engineering & Technology Advancements (ESP-JETA)*, *1*(1), 150-157.
29. Polu, A. R., Vattikonda, N., Gupta, A., Patchipulusu, H., Buddula, D. V. K. R., & Narra, B. (2021). Enhancing Marketing Analytics in Online Retailing through Machine Learning Classification Techniques. *Available at SSRN 5297803*.
30. Polu, A. R., Buddula, D. V. K. R., Narra, B., Gupta, A., Vattikonda, N., & Patchipulusu, H. (2021). Evolution of AI in Software Development and Cybersecurity: Unifying Automation, Innovation, and Protection in the Digital Age. *Available at SSRN 5266517*.
31. Gupta, K., Varun, G. A. D., Polu, S. D. E., & Sachs, G. Enhancing Marketing Analytics in Online Retailing through Machine Learning Classification Techniques.
32. Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., Vangala, S. R., Polam, R. M., & Kamarthapu, B. (2021). Big Data and Predictive Analytics for Customer Retention: Exploring the Role of Machine Learning in E-Commerce. *International Journal of Emerging Trends in Computer Science and Information Technology*, *2*(2), 26-34.
33. Polam, R. M., Kamarthapu, B., Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., & Vangala, S. R. (2021). Data Security in Cloud Computing: Encryption, Zero Trust, and Homomorphic Encryption. *International Journal of Emerging Trends in Computer Science and Information Technology*, *2*(3), 70-80.
34. Polam, R. M., Kamarthapu, B., Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., & Vangala, S. R. (2021). Big Text Data Analysis for Sentiment Classification in Product Reviews Using Advanced Large Language Models. *International Journal of AI, BigData, Computational and Management Studies*, *2*(2), 55-65.
35. Kamarthapu, B., Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., Vangala, S. R., & Polam, R. M. (2021). Advanced Machine Learning Models for Detecting and Classifying Financial Fraud in Big Data-Driven. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *2*(3), 39-46.
36. Vangala, S. R., Polam, R. M., Kamarthapu, B., Kakani, A. B., Nandiraju, S. K. K., & Chundru, S. K. (2021). Smart Healthcare: Machine Learning-Based Classification of Epileptic Seizure Disease Using EEG Signal Analysis. *International Journal of Emerging Research in Engineering and Technology*, *2*(3), 61-70.
37. Gangineni, V. N., Pabbineedi, S., Penmetsa, M., Bhumireddy, J. R., Chalasani, R., & Tyagadurgam, M. S. V. (2021). Strengthening Cybersecurity Governance: The Impact of Firewalls on Risk Management. *International Journal of AI, BigData, Computational and Management Studies*, *2*(4), 60-68.
38. Pabbineedi, S., Penmetsa, M., Bhumireddy, J. R., Chalasani, R., Tyagadurgam, M. S. V., & Gangineni, V. N. (2021). An Advanced Machine Learning Models Design for Fraud Identification in Healthcare Insurance. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *2*(1), 26-34.
39. Tyagadurgam, M. S. V., Gangineni, V. N., Pabbineedi, S., Penmetsa, M., Bhumireddy, J. R., & Chalasani, R. (2021). Enhancing IoT (Internet of Things) Security through Intelligent Intrusion Detection Using ML Models. *International Journal of Emerging Research in Engineering and Technology*, *2*(1), 27-36.
40. Penmetsa, M., Bhumireddy, J. R., Chalasani, R., Tyagadurgam, M. S. V., Gangineni, V. N., & Pabbineedi, S. (2021). Next-Generation Cybersecurity: The Role of AI and Quantum Computing in Threat Detection. *International Journal of Emerging Trends in Computer Science and Information Technology*, *2*(4), 54-61.