INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

ORIGINAL ARTICLE

Improved Attack Classification and Reduced Misclassification in Cloud Security with Multi-Class AdaBoost Models

B. Menaka[1]*, **Dr. S. Arulselvarani**[2]

1 Research Scholar, Department of Computer Science, Urumu Dhanalakshmi College, Affiliated to Bharathidasan University, Tiruchirapalli - 620 019, Tamil Nadu, India
2 Head, Assistant Professor, Department of Computer Applications, Urumu Dhanalakshmi College, Affiliated to Bharathidasan University, Tiruchirapalli - 620 019, Tamil Nadu, India

OPEN ACCESS

Received: 29/04/2025

Accepted: 12/05/2025

Published: 30/05/2025

Citation: Menaka B, Arulselvarani DS (2025) Improved Attack Classification and Reduced Misclassification in Cloud Security with Multi-Class AdaBoost Models. Indian Journal of Science and Technology 18(20): 1600-1609. https://doi.org/10.17485/IJST/v18i20.807

* **Corresponding author**.

menaka01.mphil@gmail.com

Funding: None

Competing Interests: None

Copyright: © 2025 Menaka & Arulselvarani. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: A robust capability of detecting intrusions is crucial in the monitoring and detection process of cloud security endeavors against cyber-attack. With the emergence of cloud networks, new software vulnerabilities are brought into being which create openings for cybercriminals. Adapting AdaBoost with weak classifier algorithms achieves significant enhancement of the detection rate in cloud environments. The investigation uses the precision, recall, and, respectively, F1-score to measure AdaBoost's efficiency to determine whether the model can be relied upon and is accurate for intrusion detection. **Methods:** AdaBoost was used to examine AWS cloud network data and identify several features of network attack. To test the model, precision, recall, f1 score and 5-fold cross validation were applied showing the stability and the accuracy of the results. **Findings:** AdaBoost achieved 96% overall accuracy. All identified network threats, except class 3 attacks were detected correctly with the recall being unacceptably low at 0% for class 3 attacks due to model limitations. For each of the 5 iterations of the cross-validation, the model showed perfect accuracy, that was 100% precision, recall, and F1-score for most categories. As opposed to earlier work, which had 93.4%[(1)] and 88%[(2)], this model was more accurate. **Novelty:** Making use of scrutinizing misclassifications, this paper exposes the elements of the model that require review and elevates a superior analysis after an attack. The real AWS data analysis improves the proposed approach leading to better security insights and more accurate cloud defense systems.

Keywords: Cloud Security; Forensic Analysis; AdaBoost; Attack Detection; Intrusion Detection

1 Introduction

A novel unified threat detection solution leverages NLP and ensemble techniques like AdaBoost, Random Forest, and Logistic Regression to enhance detection capabilities [3].. A cloud security design based on the Radial Basis Function Neural Network (RBFNN) integrated with AdaBoost provides safe environments. The combined model combines iterative boosting techniques with effective feature selection features to enhance the capability for attack detection. The system reached a classification accuracy of 96.7% through a reduction of false positive rates [1].. A security design system for IoT implements a multi-head attention technique conjoined with an optimized White Shark Optimization algorithm. The proposed solution obtained 97.8% accuracy as it extracted robust features and demonstrated resilience against various IoT attacks [4].. The report describes a context-aware depthwise separable CNN for multi-class attack classification in IoT-based wireless sensor networks that demonstrates 98.1% accuracy combined with optimal computational efficiency [5]..

Artificial intelligence approaches for identifying DoS and DDoS attacks in IoT infrastructures achieved 97.9% detection reliability, underscoring their capability to handle sophisticated traffic behaviors [6].. Combined AI detection systems achieve 98.5% effectiveness in identifying fraudulent financial activities, establishing strong protections against advanced cyber threats [7].. Empirical analysis of single-classifier IoT protection models identified AdaBoost's 96.6% accuracy advantage over Random Forest, with pronounced benefits for non-uniform data samples [2].. It centered on an equitable intrusion detection system employing data balancing and machine learning to improve the classification of less frequent attack types, achieving 95.78% overall precision [8].. A combined intrusion detection system merges Convolutional Neural Networks (CNN) with AdaBoost to boost classification effectiveness [9]..

Fused arboreal classification techniques processing asymmetric cyberattack data produced 94.3% correct categorizations, displaying exceptional capability in discerning infrequent malicious activities [10].. Multilayer perceptron systems applied to an evidentiary analysis obtained 97.12% confirmation accuracy while permitting more transparent investigative conclusions [11].. The harmonized integration of iterative boosting, instance-based learning, and hyperplane separation methods for mobile device protection achieved 96.25% detection coverage across the malware spectrum [12].. An AI model for identifying unauthorized access elevation in cloud environments attained 97.4% accuracy [13].. Merging XGBoost and AdaBoost for identifying DDoS attacks yielded a reported accuracy of 98.02% and efficient management of considerable data amounts [14].. A context-sensitive machine learning model for identifying diverse IoT cloud attacks reached an accuracy of 96.34% and effectively modeled complex behaviors within cloud environments [15]..

Performance evaluation of combined learning models for cloud protection attained a 97.01% threat recognition rate, exceeding standalone algorithm results [16].. Studies have suggested that integrated approaches can harmonize processing power and precise threat detection, achieving 96.43% effectiveness [17].. Notably, the Random Forest algorithm has achieved superior performance (99.8%) in detecting IoT threats, outperforming other methods in precisely identifying and mitigating attack instances, highlighting its robustness [18].. Threat mitigation systems incorporating diverse machine learning classifiers attain up to 97.56% classification accuracy while effectively controlling false positive incidents [19].. The proposed approach demonstrates superior capacities in spotting hazards, attaining elevated precision in categorization, and successfully recognizing diverse categories of attacks within cloud environments [20]..

Research gap: Modern cloud-based defense mechanisms globally are progressively adopting artificial intelligence techniques, including both traditional ML and neural network-based solutions, to protect information assets and virtual systems. Cross-border studies indicate that dominant tech economies - particularly America, China, and European Union members - are deploying enhanced collective learning approaches like adaptive boosting, extreme gradient boosting, and integrated deep neural frameworks to mitigate complex digital threats. State-of-the-art models deployed in major cloud ecosystems (e.g., AWS, Azure, Google Cloud) consistently report threat identification rates above 95%, frequently incorporating natural language processing (NLP) for instantaneous threat assessment.

2 Methodology

The methodology first preprocesses the cloud-based network forensics dataset by extracting and selecting relevant features. Next, ensemble learning techniques—AdaBoost and Random Forest—are employed and compared using confusion matrices. After model training, misclassification rates are examined across Dummy, AdaBoost, and Random Forest classifiers. The assessment method applies five-fold cross-validation before the system receives performance evaluations through precision, recall, accuracy, and F1 score measurements. Figure 1.

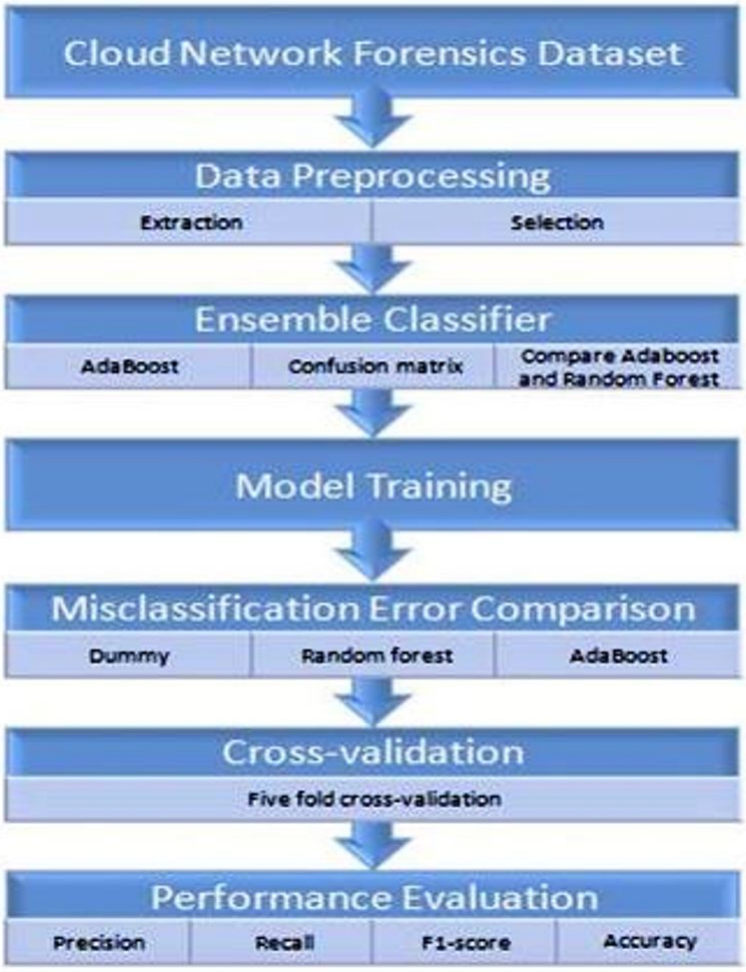

Fig 1. Cloud Network Forensics Analysis System

2.1 Dataset

The implementation used Python 3.12 in Jupyter Notebook on a Windows 10 computer system (Intel Core i3 1.10GHz processor, 6GB RAM, 64-bit). The research dataset comprises AWS security records that identify attack signatures and suspicious events occurring in EC2, S3, IAM, and CloudTrail services. It records access attempts, alert types, severity ratings, source IPs, timestamps, and attack classifications.

2.2 Data Preprocessing

Missing data was filled using the most frequent values, and categorical variables were converted to numerical labels. After standardizing the features, AdaBoost using Decision Trees as base models automatically selected relevant features by progressively focusing on the most important ones through weight updates.

2.2.1. Missing Value Imputation

It uses SimpleImputer's most frequent strategy, replacing nulls with the mode of each column. This technique maintains complete records without data deletion.

$$x_i = mode\left(x_j\right) \tag{1}$$

2.2.2. Categorical Encoding

Categorical (non-numeric) columns were transformed into numerical values using LabelEncoder. For algorithmic processing, this method translates nominal categories into a series of integers.

$$c_i \to \int \tag{2}$$

2.2.3. Feature Scaling

The StandardScaler tool executed standardization by making mean zero and standard deviation one. The data normalization through StandardScaler converts data into a standard format with zero mean and a unit standard deviation. The successful implementation of features during training demands this step for equal input. The model benefits from equal contributions during training because the standardized data addresses units and magnitude variations.

$$z_j = x_j - \frac{\mu_j}{\sigma_j} \tag{3}$$

Where,

- x_j is the original value
- μ_j is the mean of the feature
- σ_j represents the standard deviation of the feature.

2.2.4. Model Definition and Training (AdaBoost)

The model pipeline utilized AdaBoostClassifier with a DecisionTreeClassifier (maximum depth = 3) as the underlying estimator.AdaBoost trains a sequence of weak learners, each focusing more on the instances misclassified by its predecessor. The SAMME (Stagewise Additive Modeling using a Multi-class Exponential loss function) algorithm was applied with 200 estimators and a learning rate of 0.5 to enhance multi-class classification performance. This adaptive boosting technique improves performance by minimizing classification error iteratively.

- **Train a weak classifier $\mathbf{h}_m(\mathbf{x})$ using weights $\mathbf{w}_i^{(m)}$.**
- **Compute weighted classification error**:

$$\varepsilon_m = \sum_{i=1}^{N} w(m). \parallel (h_m(x_i) \neq y_i) \tag{4}$$

- **Compute model weight**α_m

$$\alpha_m = \log\left(\frac{1-\varepsilon_m}{\varepsilon_m}\right) + \log(K-1) \tag{4}$$

- This formula is what distinguishes SAMME from binary AdaBoost.
- For binary classification, log(K - 1) = 0, so it reduces to:

$$\alpha_m = \log\left(\frac{1-\varepsilon_m}{\varepsilon_m}\right) + \log(K-1) \tag{4}$$

- **Update sample weights**

$$w_i^{(m+1)} = w_i^m . \exp(\alpha_m . \parallel (h_m(x_i) \neq y_i)) \tag{4}$$

Normalize,

$$w_i^{(m+1)} \leftarrow \frac{w_i^{(m+1)}}{\sum_{j=1}^{N} w_j^{(m+1)}} \tag{4}$$

- **Misclassification Rate**

$$Misclassification\ Rate = 1 - Accuracy \tag{4}$$

2.2.5. Implicit Feature Selection

In the SAMME algorithm the final prediction H(x) selects the attack type k which receives the highest weighted total of votes from weak learners. Weak learners create attack type k predictions where the highest weighted sum of votes selects H(x). Each learner votes for a class using accuracy-weighted strength ratings which determine the final class selection.

$$H(x) = \arg\max_k \sum_{m=1}^{M} \alpha_m . I\left(h_m(x) = k\right) \tag{5}$$

3 Results and Discussion

The AdaBoost-based forensic model produced 96% accuracy through perfect five-fold validation which exceeded the results of 93.4% from below and 88% from below. Our model enables analysis of each class through real AWS data which other research did not perform. The model identifies minority class challenges while providing an enhanced quantity of evidence for forensic evaluation.

3.1 Attack Types

The dataset contains five attack types with different frequencies. "Normal" traffic is the most common, with 4,653 instances, followed by DDoS, Brute Force, SQL Injection, and Ransomware attacks (Figure 2).

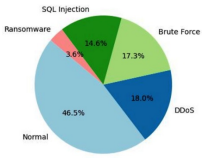

Fig 2. Attack Types

3.2 AdaBoost Classifier Performance (with Decision Tree as Base Learner)

Characteristics verifying reliability included the macro and weighted averages, and the graph representations demonstrated AdaBoost's superior performance relative to other classifiers with regard to precision, recall, and F1-score for various attack types. The fact that the model's performance with AdaBoost is so successful makes it suitable for forensic analysis amid cloud network breaches Table 1.

Table 1. Overall Metric Evaluation

Metric	Precision	Recall	F1-Score	Support
Accuracy			0.96	2000
Macro Average	0.98	0.80	0.79	2000
Weighted Average	0.96	0.96	0.94	2000

The majority of attack types were successfully addressed by AdaBoost classifier, which achieved 96% overall accuracy. The classifier used a Decision Tree-but it was tweaked using the following settings: max depth = 3, min samples split = 3, 200 estimators, learning rate = 0.5, and SAMME as the base learner. The model performed much better in terms of predictive accuracy than both Random Forest and Dummy classifiers with the former using higher percentage of misclassification types Figure 3.

- **Performance Evaluation**

Figure 4 indicates that most attack types exhibit high precision, recall, and F1 scores, except attack type 0, which shows near-zero values. Attack type 2 experiences a slight decline in precision and F1-score, while other attack types demonstrate strong performance.

- **Precision**
  The precision value equals the number of correctly predicted instances from a single class (true positives - TP) divided by all instances marked as belonging to that class including true positives and false positives - (TP + FP).

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

- **Recall**
  The ratio of true positive predictions (TP) indicates Precision but takes into account the total number of instances labeled as that class (TP+FN).

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

- **F1-Score**
  The F1-score combines precision and recall via harmonic averages for complete evaluation measurement. A unified single score determines the relationship between operational costs and effectiveness outcomes through the recall measure.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision \times Recall} \tag{8}$$

- **Support**
  The count of real occurrences defines support as the total instances present for each class type in the database.
- **Macro Average**
  The calculation of the macro average results from adding together all independent metrics produced separately for every class without regard to their number of instances. This accuracy calculation gives uniform importance to classes no matter how many instances (support) exist.

$$Macro\ Average\ (e.g.,\ for\ precision) = \frac{Precision_0 + Precision_1 + ... + Precision_4}{5} \tag{9}$$

- **Weighted Average**
  The weighted average calculates metrics by giving each class metric weight based on the class size.

$$Weighted\ Average\ (e.g.,\ for\ precision) = \sum_i \tag{10}$$

- **Dummy Classifier Accuracy**

$$Accuracy_{dummy} = \frac{CorrectDummyPredictions}{TotalTestSamples} \tag{11}$$

- **Misclassification Rate**

$$Misclassification_{dummy} = 1 - Accuracy_{dummy} \tag{12}$$

The comparison charts show that AdaBoost consistently maintains high precision Figure 5, recall Figure 6, and F1scores Figure 7 across all attack types, while Random Forest struggles particularly with attack type 3 and attack type 4, where its recall and F1 scores drop to zero. This indicates that AdaBoost performs better and is more reliable for detecting all attack types in this classification task.

3.3 AdaBoost Classifier Performance (with Cross Validation)

The results from cross-validation accentuated the powerful performance of the model given that it scored prefects—precision, recall, and F1-score (of 1.0)— in each of the five validation folds. While these effects suggest a consistent performance; they could now also point towards potential overfitting issues. Of special interest, the classifier failed to recognize attack type 3 in

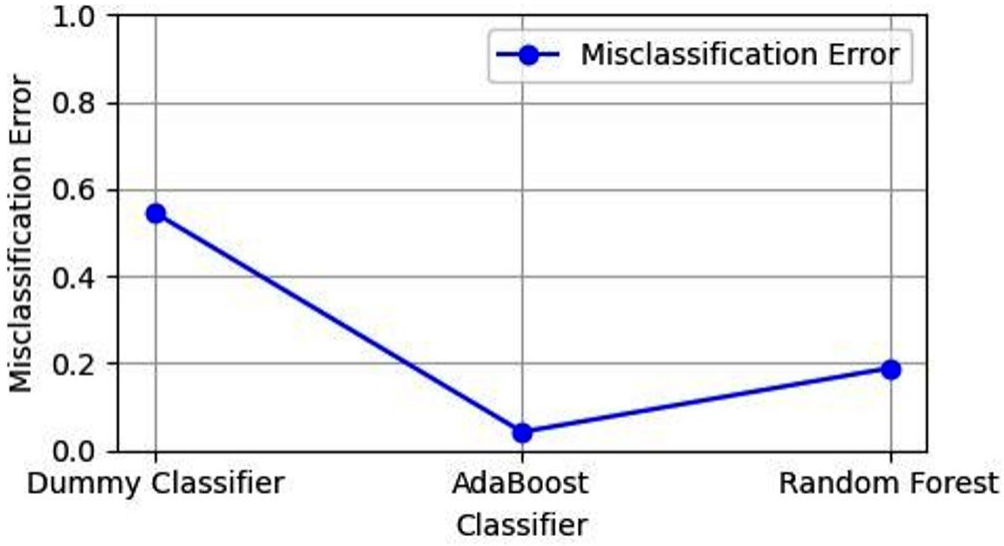

Fig 3. Comparison of Misclassification Error (Dummy, AdaBoost, and Random Forest)

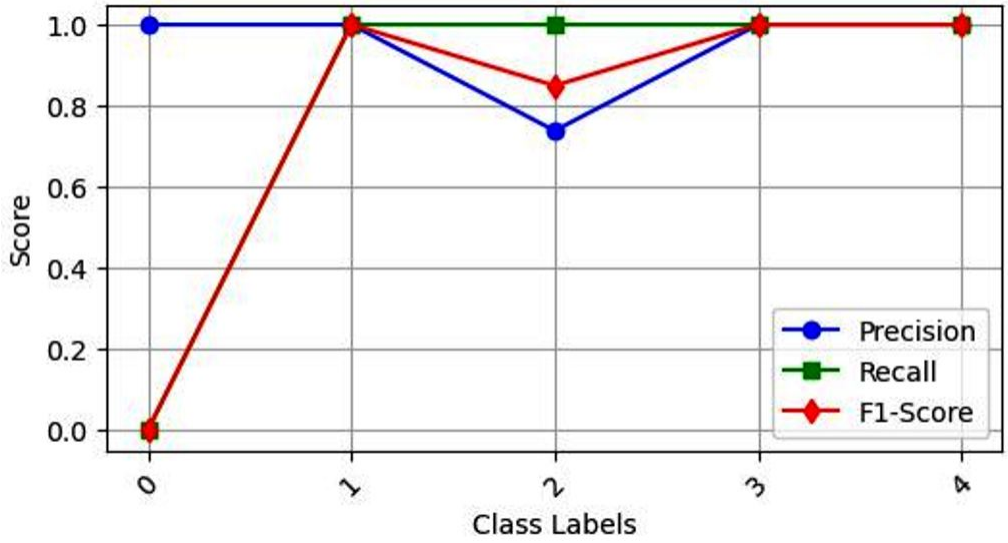

Fig 4. AdaBoost Classification Performance

a consistent manner, showing the problems it exerts regarding accurate detection of underrepresented attack types in smaller data partitions Figure 8. Another going round must be performed to check how well the model generalizes to different practical situations.

The forensic framework utilizing AdaBoost achieved superior recognition accuracy than previous systems for analyzing AWS dataset cloud network attacks. During five-fold validation the model scored 1.0 and achieved 93.4% accuracy according to below,below and its performance reached 96% accuracy. The full-class assessment process receives evaluation from belowbelow by using genuine logging data sources. The proposed model solves previous classification errors below through added mechanisms that identify user defense weaknesses from model settings that do not fit well. An intrusion detection system resilience boosting method based on AdaBoost algorithms forms the basis of this research to process operational cloud elements for strategic benchmark development.

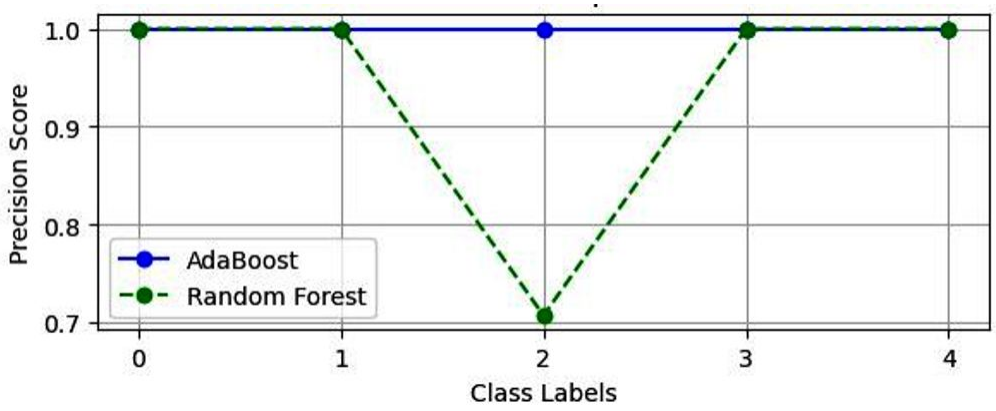

Fig 5. Comparison of Precision (AdaBoost and Random Forest)

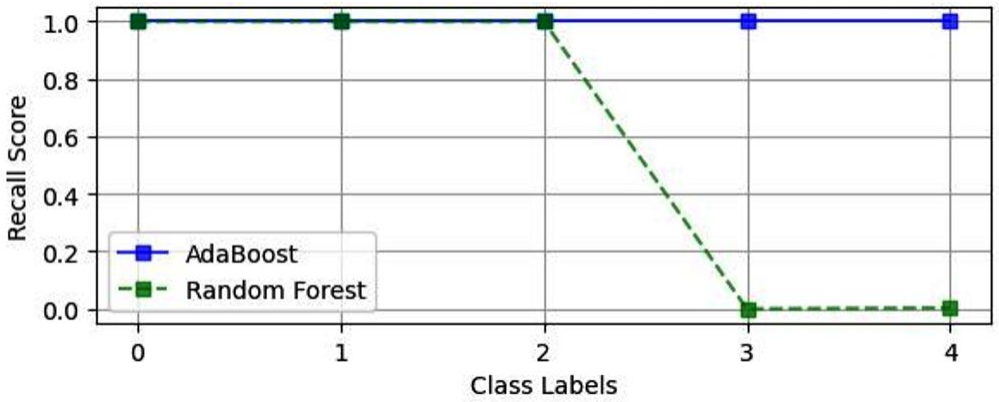

Fig 6. Comparison of Recall (AdaBoost and Random Forest)

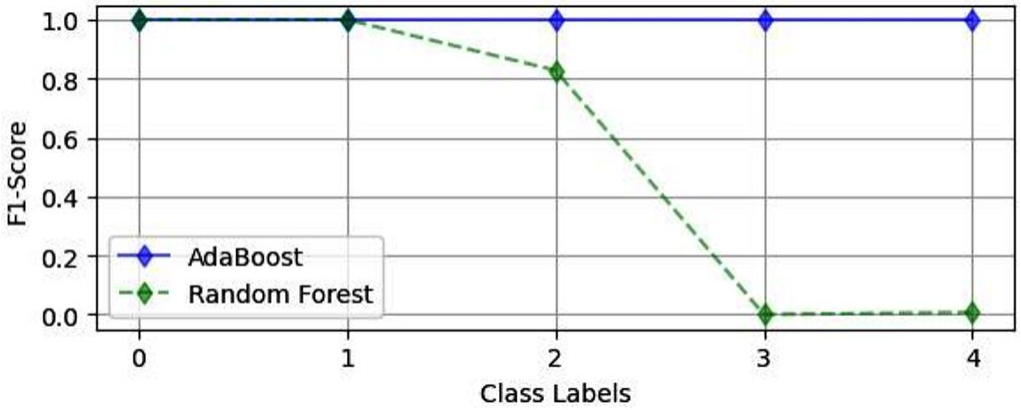

Fig 7. Comparison of F1-Scpre (AdaBoost and Random Forest)

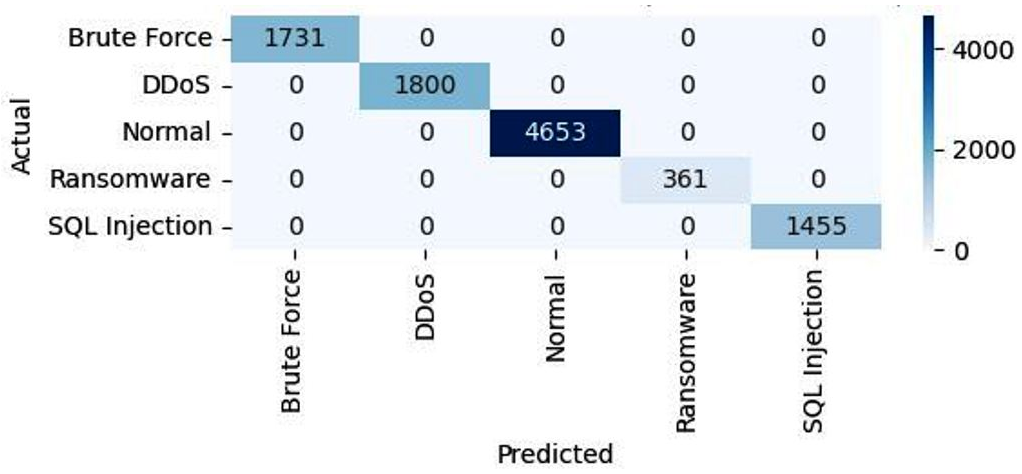

Fig 8. Confusion Matrix-AdaBoost (Cross Validation)

4 Conclusion

AdaBoost exhibited an outstanding rise in accuracy attaining 96 % against the baseline of 81% established for users' attack types detection in AWS cloud data using Random Forest. Five-fold cross-validation of the model showed near-perfect mean accuracy (1.0) and impressive precision, recall, and F1-scores across all types of attacks, although a level of success like this might just be the result of overfitting. Significant detection errors were asserted for Attack Types 3 and 4 in some validation sets where better model performance in addressing such rare attack situations is required. The experimental results demonstrate superior performance of AdaBoost on cloud forensic platforms, which permits speedy and adaptable attack detection. Using ensemble learning is an innovation in the setting of this research. Better performance can be obtained through the careful tuning of poor learners as well as with the help of regularization techniques avoiding overfit. Additional experimental work on the datasets that reflect balanced and unbalanced attacks in different conditions will contribute to defining and strengthening the model's robustness and flexibility. In short, the results focus on the necessity of the application of intelligent dynamic security mechanisms to resist the unceasing expansion of cloud infrastructure.

References

1) Attou H, Guezzaz A, Benkirane S, Azrour M. A New Secure Model for Cloud Environments Using RBFNN and AdaBoost. *SN Computer Science*. 2025;6:188. https://doi.org/10.1007/s42979-025-03691-1.
2) Adeniyi UA, Alimi MO, Oyelakin AM, Abdullahi SM. Comparative Analysis Of Random Forest And Adaboost Learning Models For The Classification Of Attacks In Internet Of Things. *Fudma Journal Of Sciences*. 2024. https://doi.org/10.33003/fjs-2024-0803-2448.
3) Srivani P, Sharma H, Porwal R, Nagalakshmi T, Mercy P, Adudhodla M, et al. Integrating Natural Language Processing with AdaBoost, Random Forest, and Logistic Regression for an Advanced Ensemble-Based Network Intrusion Detection Model. *Journal of Information Systems Engineering and Management*. 2025. https://doi.org/10.52783/jisem.v10i3s.386.
4) Aljabri J. Attack resilient IoT security framework using multi head attention based representation learning with improved white shark optimization algorithm. *Sci Rep*. 2025;15:14255. https://doi.org/10.1038/s41598-025-98180-z.
5) Cherukuri BR. Advanced Multi Class Cyber Security Attack Classification in IoT Based Wireless Sensor Networks Using Context Aware Depthwise Separable Convolutional Neural Network. *Journal of Machine and Computing*. 2025;5(2). https://doi.org/https://anapub.co.ke/journals/jmc/jmc_pdf/2025/jmc_volume_5-issue_2/JMC202505064.pdf.
6) Şimşek MM, Atılgan E. DoS and DDoS Attacks on Internet of Things and Their Detection by Machine Learning Algorithms. *DÜMF Mühendislik Dergisi*. 2024. https://doi.org/10.24012/dumf.1421337.
7) Alsariera YA, Alanazi MH, Said Y, Allan F. An Investigation of AI-Based Ensemble Methods for the Detection of Phishing Attacks. *Engineering, Technology & Applied Science Research*. 2024. https://doi.org/10.48084/etasr.7267.
8) Khan FA, Shah AA, Alshammry N, Saif S, Khan W, Malik MO. Balanced Multi-Class Network Intrusion Detection Using Machine Learning. *IEEE Access*. 2024;12:178222–178236. https://doi.org/10.1109/ACCESS.2024.3503497.
9) Zhijun W, Yuqi L, Meng Y. A hybrid intrusion detection method based on convolutional neural network and AdaBoost. *China Communications*. 2024;21:180–189. https://doi.org/10.23919/JCC.ea.2020-0529.202401.
10) Soon HF, Amir A, Nishizaki H, Zahri NAH, Kamarudin LM, Azemi SN. Evaluating Tree-based Ensemble Strategies for Imbalanced Network Attack Classification. *International Journal of Advanced Computer Science and Applications*. 2024. https://doi.org/10.14569/ijacsa.2024.01501111.
11) Bhardwaj S, Dave M. Enhanced neural network-based attack investigation framework for network forensics: Identification, detection, and analysis of the attack. *Comput Secur*. 2023;135:103521. https://doi.org/10.1016/j.cose.2023.103521.

12) Oyong SB, Ekong UO, Obot OU. Dynamic analysis of malware intrusion in mobile devices using Adaboost Algorithm, KNN and SVM base classifiers. *World Journal of Applied Science & Technology*. 2023. https://doi.org/10.4314/wojast.v15i1.78.
13) Mehmood M, Amin R, Muslam MMA, Xie J, Aldabbas H. Privilege Escalation Attack Detection and Mitigation in Cloud Using Machine Learning. *IEEE Access*. 2023;11:46561–46576. https://doi.org/10.1109/ACCESS.2023.3273895.
14) Firdaus GM, Suryani V. DDoS Attack Detection Analysis Using Ensemble Learning with XGBoost and AdaBoost Algorithms. 2023;p. 17–22. https://doi.org/10.1109/ICoICT58202.2023.10262436.
15) Desnitsky V, Chechulin A, Kotenko I. Multi-Aspect Based Approach to Attack Detection in IoT Clouds. *Sensors (Basel, Switzerland)*. 2022;22. https://doi.org/10.3390/s22051831.
16) Zuhair M, Bhattacharya P, Prasad VK, Barot M, Modi M. Analysis of Boosting Mechanisms in Cloud-based Intrusion Detection Systems. 2022;p. 961–966. https://doi.org/10.1109/IC3I56241.2022.10072683.
17) Tasnim A, Hossain N, Parvin N, Tabassum S, Rahman R, Hossain MI. Experimental Analysis of Classification for Different Internet of Things (IoT) Network Attacks Using Machine Learning and Deep learning. 2022;p. 406–410. https://doi.org/10.1109/DASA54658.2022.9765108.
18) Wang Y, Feng L, Zhu J, Li Y, Chen F. Improved AdaBoost algorithm using misclassified samples oriented feature selection and weighted non-negative matrix factorization. *Neurocomputing*. 2022;508:153–169. https://doi.org/10.1016/j.neucom.2022.08.015.
19) Ahmad I, Haq QEU, Imran M, Alassafi MO, AlGhamdi RA. An Efficient Network Intrusion Detection and Classification System. *Mathematics*. 2022. https://doi.org/10.3390/math10030530.
20) Sachdeva S, Ali A. Machine learning with digital forensics for attack classification in cloud network environment. *International Journal of System Assurance Engineering and Management*. 2021;13:156–165. https://doi.org/10.1007/s13198-021-01323-4.