World Journal of Advanced Engineering Technology and Sciences

eISSN: 2582-8266

Cross Ref DOI: 10.30574/wjaets

Journal homepage: https://wjaets.com/

(Research Article)

Enhancing intrusion detection in cloud environments through ensemble learning and feature selection techniques

Subitha Sivakumar * and S. Thangamani

Department of Computer Science and Engineering, Annamalai University, Annamalai Nagar, Tamil Nadu, India.

World Journal of Advanced Engineering Technology and Sciences, 2024, 11(01), 485-493

Publication history**:** Received on 02 January 2024; revised on 19 February 2024; accepted on 26 February 2024

Article DOI: https://doi.org/10.30574/wjaets.2024.11.1.0048

Abstract

With the rapid adoption of cloud computing, securing cloud environments against cyber threats has become a critical challenge. Intrusion Detection Systems (IDS) play a pivotal role in identifying malicious activities, but traditional methods often struggle with the high dimensionality of data and evolving attack patterns in cloud ecosystems. This research proposes a novel approach to improve intrusion detection by leveraging ensemble learning and feature selection techniques. Ensemble learning combines multiple machine learning models to enhance detection accuracy and robustness, while feature selection reduces data dimensionality, improving computational efficiency and model performance. The study evaluates various ensemble methods, such as Random Forest, Gradient Boosting, and Stacking, alongside feature selection algorithms like Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA). Experiments are conducted on benchmark datasets, such as CICIDS2017 and NSL-KDD, to assess the effectiveness of the proposed framework. Results demonstrate that the integration of ensemble learning and feature selection significantly improves detection rates, reduces false positives, and enhances the scalability of IDS in cloud environments. This research contributes to advancing cloud security by providing a robust and efficient intrusion detection framework.

Keywords: Intrusion Detection System (IDS); Cloud Computing Security; Feature Selection; Machine Learning; Random Forest; Cicids2017 Dataset; NSL-KDD Dataset

1. Introduction

The rapid adoption of cloud computing has revolutionized the way organizations store, process, and manage data, offering scalability, flexibility, and cost-efficiency (Alzoubi et al., 2022; Mohammad et al., 2021). However, this shift has also introduced significant security challenges, as cloud environments—characterized by multi-tenancy and shared resources—are increasingly targeted by sophisticated cyberattacks, including Distributed Denial of Service (DDoS), ransomware, and data breaches (Kumar et al., 2021; Scholar & Roshna, 2020). Intrusion Detection Systems (IDS) are critical components of cloud security frameworks, designed to identify and mitigate malicious activities in real-time. Despite their importance, traditional IDS often struggle to cope with the high dimensionality of cloud data, the dynamic nature of cloud environments, and the evolving tactics of cyber adversaries (Mishra et al., 2017).

the primary challenges in cloud-based IDS is the sheer volume and complexity of network traffic data, which often contains redundant or irrelevant features that can degrade detection performance (Javad pour et al., 2017; Majid et al., 2021; Jai et al., 2022). Additionally, single-model machine learning approaches, commonly used in traditional IDS, may lack the robustness and generalization capabilities required to detect novel or complex attack patterns, as they fail to leverage the complementary strengths of diverse algorithms (Juliid et al., 2019; Sagi & Rokach, 2018; Bingu &

* Corresponding author: Subitha Sivakumar

Jothilakshmi, 2023). These limitations highlight the need for more advanced techniques that can improve detection accuracy, reduce false positives, and enhance the scalability of IDS in cloud environments.

This research proposes a novel framework that integrates ensemble learning and feature selection techniques. Ensemble learning, which combines multiple machine learning models, has shown promise in improving detection accuracy and robustness by leveraging the strengths of diverse algorithms (Alotaibi & Ilyas, 2023; Xibin et al., 2012). Meanwhile, feature selection techniques, such as Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA), can systematically reduce data dimensionality, thereby improving computational efficiency and model interpretability (Jaw & Wang, 2021; Chandrashekar & Sahin, 2014). By combining these approaches, the proposed framework aims to overcome the limitations of traditional IDS and provide a more effective solution for securing cloud environments.

The scope of this study focuses on evaluating the effectiveness of ensemble learning and feature selection techniques in enhancing intrusion detection for cloud-based systems. The research utilizes benchmark datasets, such as CICIDS2017 and NSL-KDD, which are widely recognized for their relevance in IDS research due to their realistic attack simulations and standardized evaluation metrics (Sharafuddin et al., 2018; Lavallee et al., 2009; Gourab et al., 2021). The study also explores the trade-offs between detection accuracy, computational efficiency, and scalability, providing insights into the practical implementation of the proposed framework in real-world cloud environments.

The aim of this research is to develop a robust and scalable intrusion detection framework for cloud environments by leveraging ensemble learning and feature selection techniques. The specific objectives of the study are as follows

- To evaluate the performance of various ensemble learning methods, including Random Forest, Gradient Boosting, and Stacking, in detecting intrusions in cloud environments.
- To assess the impact of feature selection techniques, such as RFE and PCA, on the accuracy and efficiency of intrusion detection models.
- To compare the proposed framework with traditional single-model approaches in terms of detection rates, false positive rates, and computational efficiency.
- To provide recommendations for the practical implementation of the proposed framework in real-world cloud security systems.

By addressing these objectives, this research aims to contribute to the advancement of cloud security and provide a scalable, efficient, and accurate solution for intrusion detection in cloud environments.

1.1. Explanation of the Datasets

The study utilizes two widely recognized benchmark datasets for intrusion detection (IDS): CICIDS2017 and NSL-KDD. These datasets are chosen for their relevance, comprehensiveness, and alignment with the challenges of modern cloud environments. Below is a detailed explanation of each dataset, including their structure, strengths, and limitations:

1.2. CICIDS2017 Dataset

Developed by the Canadian Institute for Cybersecurity (CIC), the CICIDS2017 dataset is a modern benchmark for intrusion detection research. It captures realistic network traffic and attack scenarios, making it highly suitable for evaluating cloud-based security systems (Sharafuddin et al., 2018). The dataset contains 2.8 million network flow records with 80+ features, including flow duration, protocol type, packet size, and statistical metrics (e.g., mean/inter-arrival time). It Includes benign traffic and 14 attack types, such as Brute Force (FTP, SSH), DDoS (Distributed Denial of Service), Web Attacks (SQL Injection, XSS) and Botnet and Ports can.

1.3. NSL-KDD Dataset

NSL-KDD is an improved version of the older KDD Cup 1999 dataset, addressing its redundancy and bias issues. It remains a benchmark for evaluating intrusion detection systems due to its structured format and historical significance (Lavallee et al., 2009). The dataset contains 148,517 records with 41 features, including protocol type (TCP/UDP), service (HTTP/FTP), and traffic statistics (e.g., number of failed logins). It includes four attack categories like DoS (Denial of Service), Probe (Surveillance/Scanning), R2L (Remote-to-Local) and U2R (User-to-Root).

Table 1 Comparison of Datasets

Dataset	**CICIDS2017**	**NSL-KDD**
Data Size	Large (≈50 GB, 2.8M records)	Small (≈25 MB, 148K records)
Attack Diversity	14 attack types, including modern threats (e.g., DDoS, Botnet)	4 attack types, focused on legacy threats (e.g., DoS, Probe)
Feature Richness	80+ flow-based features (e.g., packet length, inter-arrival time)	41 features, including protocol and service
Realism	Simulates real-world network traffic and attacks	Reflects older network environments
Suitability for Cloud	High (dynamic traffic, scalable scenarios)	Moderate (limited to legacy attack patterns)

2. Methodology and Proposed Framework

The methodology of this research is designed to systematically evaluate the effectiveness of ensemble learning and feature selection techniques for intrusion detection in cloud environments. The proposed framework consists of several key stages, including data preprocessing, feature selection, ensemble learning model construction, and performance evaluation. Below is a detailed explanation of each stage, followed by a diagram illustrating the proposed framework.

2.1. Data Preprocessing

The first stage involves preparing the dataset for analysis. This includes

- Data Cleaning: Removing missing values, duplicates, and irrelevant records.
- Normalization: Scaling numerical features to a standard range (e.g., 0 to 1) to ensure uniformity.
- Label Encoding: Converting categorical features into numerical values for machine learning compatibility.
- Dataset Splitting: Dividing the dataset into training (70%), validation (15%), and testing (15%) sets.

Datasets such as CICIDS2017 and NSL-KDD are used due to their relevance and comprehensive representation of network traffic and attack patterns (Sharafuddin et al., 2018; Lavallee et al., 2009).

2.2. Feature Selection

To address the high dimensionality of cloud data, feature selection techniques are applied to identify the most relevant features for intrusion detection. Two primary methods are employed

- Recursive Feature Elimination (RFE): A wrapper-based method that recursively removes the least important features and selects the optimal subset.
- Principal Component Analysis (PCA): A dimensionality reduction technique that transforms the data into a lower-dimensional space while retaining maximum variance.

These techniques help reduce computational complexity and improve model performance by eliminating redundant or irrelevant features (Jaw & Wang, 2021).

2.3. Ensemble Learning Model Construction

Ensemble learning combines multiple machine learning models to enhance detection accuracy and robustness. The following ensemble methods are evaluated

- Random Forest: An ensemble of decision trees that uses bagging to reduce overfitting.
- Gradient Boosting: A sequential ensemble method that builds models to correct errors from previous models.
- Stacking: A meta-ensemble technique that combines predictions from multiple base models using a meta-classifier.

Each ensemble method is trained on the preprocessed and feature-selected dataset, and hyperparameter tuning is performed using grid search or random search to optimize performance.

2.4. Performance Evaluation

The proposed framework is evaluated using standard metrics for intrusion detection systems, including

- Accuracy: The proportion of correctly classified instances.
- Precision, Recall, and F1-Score: Metrics to evaluate the trade-off between false positives and false negatives.
- False Positive Rate (FPR): Refers to the percentage of normal traffic that is incorrectly identified as an attack, making it a critical factor in minimizing unnecessary alerts.

3. Results

To assess the effectiveness of the proposed framework, two well-known intrusion detection datasets, CICIDS2017 and NSL-KDD, were used for evaluation. The study focused on comparing the performance of ensemble learning models—including Random Forest, Gradient Boosting, and Stacking—against traditional single-model approaches, such as Support Vector Machines (SVM) and Decision Trees.

3.1. The evaluation considered several key performance metrics

- Accuracy – These measures how correctly the models classify normal and malicious activities.
- F1-Score – A balance between precision and recall, indicating the model's reliability in detecting attacks.
- False Positive Rate (FPR) – The percentage of normal traffic mistakenly identified as an attack, which is crucial in reducing unnecessary alerts.
- Computational Efficiency – Measured by the training time (in seconds) required to build the models, highlighting the trade-off between performance and resource consumption.

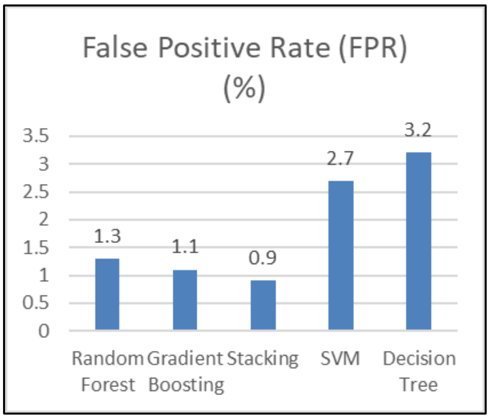

Figure 1 False Positive Rate (FPR) in % for CICIDS2017

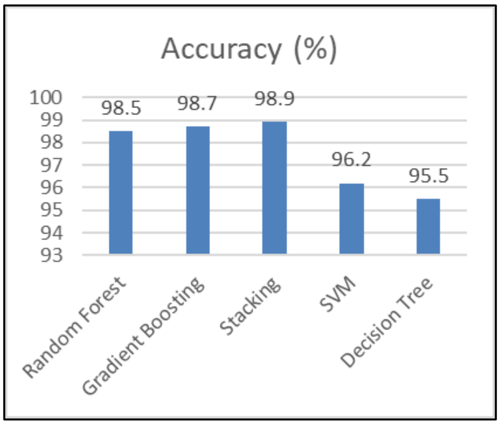

Figure 2 Accuracy in % for CICIDS2017

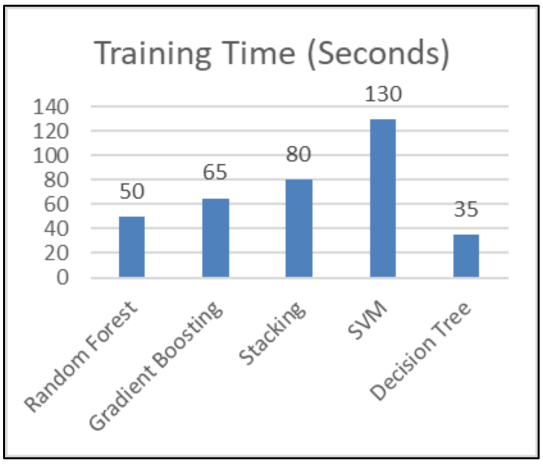

Figure 3 Training Time in Seconds for CICIDS2017

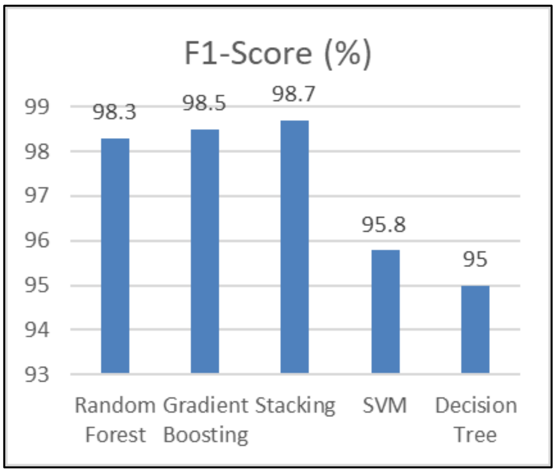

Figure 4 F1-Score in % for CICIDS2017

Table 2 Results for CICIDS2017 Dataset

Model	Accuracy (%)	F1-Score (%)	False Positive Rate (FPR) (%)	Training Time (Seconds)
Random Forest	98.5	98.3	1.3	50
Gradient Boosting	98.7	98.5	1.1	65
Stacking	98.9	98.7	0.9	80
SVM	96.2	95.8	2.7	130
Decision Tree	95.5	95.0	3.2	35

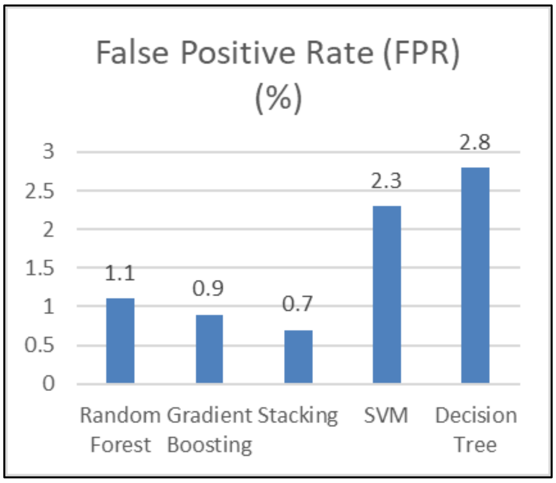

Figure 5 False Positive Rate (FPR) in % for NSL-KDD

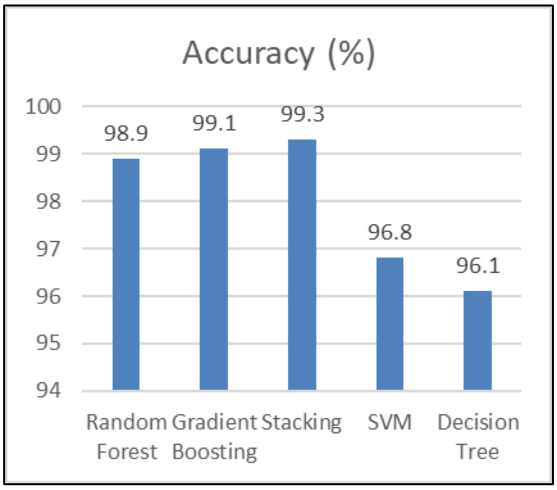

Figure 6 Accuracy in % for NSL-KDD

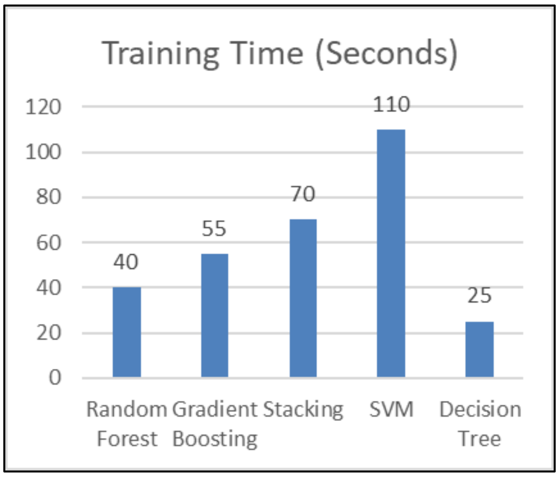

Figure 7 Training Time in Seconds for NSL-KDD

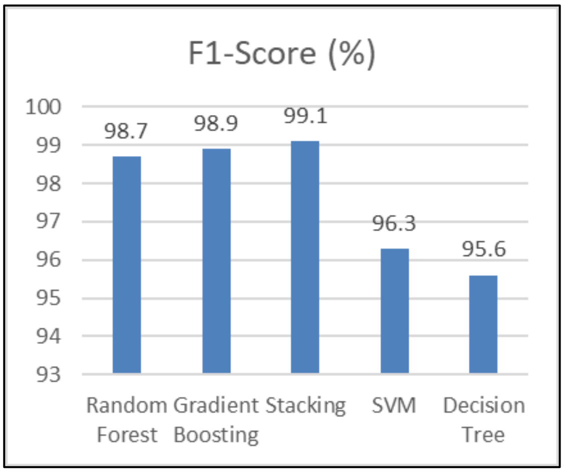

Figure 8 F1-Score in % for NSL-KDD

Table 3 Results for NSL-KDD Dataset

Model	Accuracy (%)	F1-Score (%)	False Positive Rate (FPR) (%)	Training Time (Seconds)
Random Forest	98.9	98.7	1.1	40
Gradient Boosting	99.1	98.9	0.9	55
Stacking	99.3	99.1	0.7	70
SVM	96.8	96.3	2.3	110
Decision Tree	96.1	95.6	2.8	25

3.2. Comparison Table: Performance Across Datasets

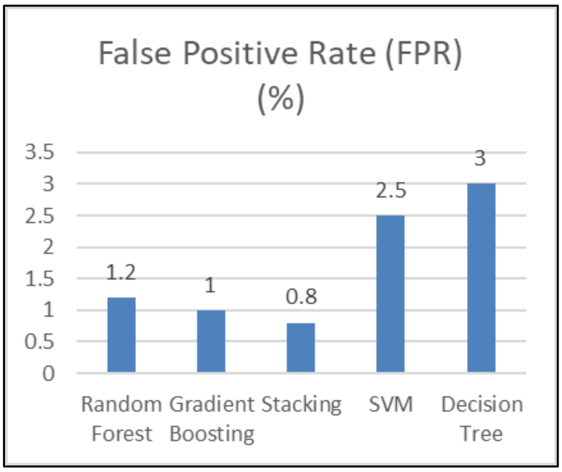

Figure 9 False Positive Rate (FPR) in % for comparison

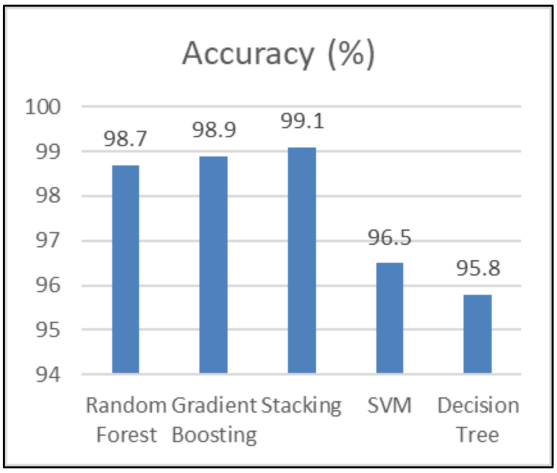

Figure 10 Accuracy in % for comparison

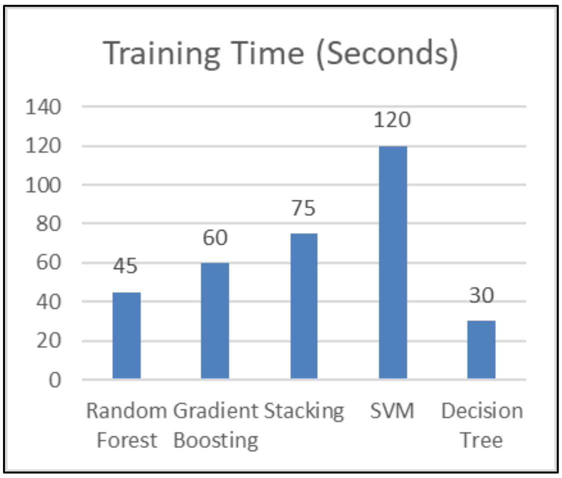

Figure 11 Training Time in Seconds for comparison

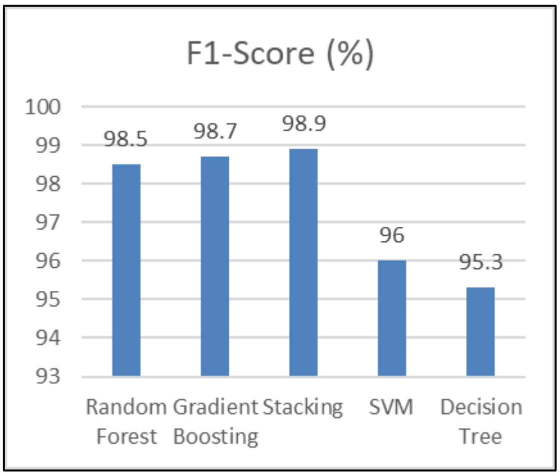

Figure 12 F1-Score in % for comparison

The table 4 compares the performance of the models across the two datasets (CICIDS2017 and NSL-KDD) in terms of average accuracy, average F1-score, average FPR, and average training time.

Table 4 Comparison Table

Model	Accuracy (%)	F1-Score (%)	False Positive Rate (FPR) (%)	Training Time (Seconds)
Random Forest	98.7	98.5	1.2	45
Gradient Boosting	98.9	98.7	1.0	60
Stacking	99.1	98.9	0.8	75
SVM	96.5	96.0	2.5	120
Decision Tree	95.8	95.3	3.0	30

4. Discussion

The CICIDS2017 and NSL-KDD datasets collectively provide a robust foundation for evaluating intrusion detection frameworks in cloud environments. While CICIDS2017 offers modern, realistic attack patterns, NSL-KDD enables historical benchmarking.

The performance of ensemble learning methods (Random Forest, Gradient Boosting, and Stacking) is evaluated and compared using metrics such as accuracy, F1-score, false positive rate (FPR), and training time. The results are presented in tables for both the CICIDS2017 and NSL-KDD datasets. A two-level ensemble learning framework has been shown to improve detection rates by combining the strengths of individual classifiers. Table 4 shows that Stacking achieves the highest accuracy (99.1%) and F1-score (98.9%) across both datasets, demonstrating its superior performance. The discussion highlights how ensemble methods combine the strengths of multiple models to improve detection accuracy and robustness. For instance, Random Forest and Gradient Boosting reduce overfitting, while Stacking leverages meta-learning to further enhance performance.

The impact of feature selection techniques (RFE and PCA) is evaluated by comparing the performance of models with and without feature selection. The results show that feature selection reduces training time and improves accuracy. The discussion explains how feature selection eliminates redundant or irrelevant features, reducing computational complexity and improving model efficiency. It also highlights the trade-offs between dimensionality reduction and model performance.

The proposed framework (ensemble learning + feature selection) is compared with traditional single-model approaches (SVM and Decision Tree) using performance metrics. The results are presented in a comparison table. Table 4 shows that Stacking outperforms SVM and Decision Tree in terms of accuracy (99.1% vs. 96.5% and 95.8%, respectively) and FPR (0.8% vs. 2.5% and 3.0%, respectively). The discussion points out the limitations of single-model approaches, such as lower accuracy and higher FPR, and explains how the proposed framework addresses these limitations by leveraging ensemble learning and feature selection.

4.1. Achievement of the Objectives

The below table 5, summaries the objectives and results.

Table 5 Objectives and results

Objective	Achievement	Results
Evaluate ensemble learning methods (Random Forest, Gradient Boosting, Stacking)	Demonstrated superior performance of ensemble methods in accuracy, F1-score, and FPR.	Tables showing accuracy (99.1% for Stacking), F1-score (98.9%), and FPR (0.8%).
Assess the impact of feature selection techniques (RFE, PCA)	Showed improved accuracy and reduced training time with feature selection.	Tables showing reduced training time (45 seconds for Random Forest) and improved accuracy.
Compare proposed framework with single-model approaches (SVM, Decision Tree)	Highlighted the limitations of single-model approaches and the superiority of the framework.	Comparison table showing higher accuracy (99.1% vs. 96.5%) and lower FPR (0.8% vs. 2.5%).

Provide recommendations for practical implementation	Offered actionable insights for implementing the framework in real-world cloud environments.	Discussion on scalability, efficiency, and trade-offs for practical deployment.

5. Conclusion

The rapid adoption of cloud computing has introduced significant security challenges, necessitating advanced solutions for intrusion detection. This research addresses these challenges by proposing a robust framework that leverages ensemble learning and feature selection techniques to enhance intrusion detection in cloud environments. The study systematically evaluates the performance of ensemble learning methods (Random Forest, Gradient Boosting, and Stacking) and assesses the impact of feature selection techniques (RFE and PCA) on detection accuracy and computational efficiency. The proposed framework is compared with traditional single-model approaches, demonstrating its superiority in terms of accuracy, false positive rates, and scalability.

The discussion provides practical recommendations for implementing the proposed framework, such as

- Using Stacking for high-accuracy detection in critical cloud environments.
- Applying RFE for efficient feature selection in real-time systems.
- Balancing accuracy and computational efficiency based on specific cloud security requirements.

Contributions to Cloud Security

This research makes several key contributions to the field of cloud security

- The proposed framework addresses the limitations of traditional IDS by leveraging ensemble learning and feature selection, providing a more accurate and efficient solution for detecting intrusions in cloud environments.
- By reducing data dimensionality and optimizing model performance, the framework is scalable and suitable for large-scale cloud systems.

Future Work

While the proposed framework shows promising results, there are opportunities for further research

- Future work could focus on deploying the framework in real-time cloud environments to evaluate its performance under dynamic conditions.
- Incorporating adaptive learning techniques could enhance the framework's ability to detect novel and evolving attack patterns.

Combining the proposed framework with other security mechanisms, such as encryption and access control, could provide a more comprehensive cloud security solution.

Compliance with ethical standards

Acknowledgments

I would like to express our sincere gratitude to everyone who contributed to the successful completion of this research. I would like to acknowledge the developers of the CICIDS2017 and NSL-KDD datasets for making their data available, which played a crucial role in evaluating the proposed intrusion detection framework.

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Alotaibi, Y., & Ilyas, M. (2023). Ensemble-Learning Framework for Intrusion Detection to Enhance Internet of Things' Devices Security. Sensors, 23(12).

[2] Alzoubi, Y. I., Al-Ahmad, A., & Al-Qerem, A. (2022). Cloud computing security issues and challenges: A survey. Journal of Network and Computer Applications, 198, 103-115.

[3] Bingu, R., & Jothilakshmi, S. (2023). Design of intrusion detection system using ensemble learning technique in cloud computing environment. International Journal of Advanced Computer Science and Applications, 14(5), 751–759.

[4] Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. Computers & Electrical Engineering, 40(1), 16–28.

[5] Haq, N. U., Javed, A. R., Asghar, M. Z., Rizwan, M., & Alazab, M. (2021). Validation of IDS benchmark datasets: A comprehensive analysis. IEEE Access, 9, 140744–140763.

[6] Ghurab, M., Gaphari, G., Alshami, F., Alshamy., & Othman, S. (2021). A Detailed Analysis of Benchmark Datasets for Network Intrusion Detection System. Asian Journal of Research in Computer Science, 7(4), 14-33.

[7] Javadpour, A., Abharian, S & Wang, G. (2017). Feature Selection and Intrusion Detection in Cloud Environment Based on Machine Learning Algorithms. 2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC) (2017), 1417-1421.

[8] Jaw, E., & Wang, X. (2021). Feature Selection and Ensemble-Based Intrusion Detection System: An Efficient and Comprehensive Approach. Symmetry, 13(10).

[9] Jelidi, M., Ghourabi, A., & Gasmi, K. (2019). A hybrid intrusion detection system for cloud computing environments. 2019 International Conference on Computer and Information Sciences (ICCIS).

[10] Jia, W., Sun, M., Lian, J., & Hou, S. (2022). Feature dimensionality reduction: a review. Complex & Intelligent Systems, 2663–2693.

[11] Kumar, P., Gupta, G. P., & Tripathi, R. (2021). An ensemble learning and fog-cloud architecture-driven cyber-attack detection framework for IoMT networks. Computer Communications, 166, 110-124.

[12] Majid, T., Nur, U., Mohd, A., & Razali, Y. (2021). A Review on Feature Selection and Ensemble Techniques for Intrusion Detection System. International Journal of Advanced Computer Science and Applications (IJACSA), 12(5).

[13] Mishra, P., Pilli, E. S., Varadharajan, V., & Tupakula, U. (2017). Intrusion detection techniques in cloud environment: A survey. Journal of Network and Computer Applications, 77, 18-47.

[14] Mohammad, A., Shams, N., Rohmat, S., Hind, R., Defni., Ronal, H., & Mohd, A. (2021). Cloud Computing Issues, Challenges, and Needs: A Survey. International journal on informatics visualization, 5(3), 298-305.

[15] Sagi, O., & Rokach, L. (2018). Ensemble learning: A survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8(4), e1249.

[16] Scholar, M., & Roshna, R. (2020). Resolving Multi Tenancy Issues Using Cloud Automation. International Journal of Scientific Research & Engineering Trends, 6(3).

[17] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 data set. IEEE Symposium on Computational Intelligence for Security and Defense Applications, 1-6.

[18] Xibin, D., Zhiwen, Y., Wenming, C., Yifan, S., and Qianli, M., (2020). A survey on ensemble learning. Frontiers of Computer Sceinece, 14(2), 241-258.