

Research Paper

An Advanced Cloud IDS Using Random Forest-Based Ensemble Learning and Feature Reduction Techniques

¹T MUNI KUMARI, ²R HENDRA KUMAR

¹PG Scholar, Dept. of CSE, KMM Institute of Technology & Science, Ramireddipalle, Tirupati, AP, India.

²Associate Professor, Dept. of CSE, KMM Institute of Technology & Science, Ramireddipalle, Tirupati, AP, India.

Abstract: Cloud computing environments are increasingly vulnerable to sophisticated cyber-attacks, making efficient intrusion detection systems (IDS) essential for ensuring security. This paper proposes an enhanced cloud-based intrusion detection approach using hybrid ensemble learning and feature engineering techniques. Initially, data preprocessing and visualization-driven feature engineering are applied to reduce irrelevant features and improve data quality. The system integrates multiple machine learning and deep learning models, including Random Forest, Decision Tree, Support Vector Machine, Naïve Bayes, LSTM, and advanced ensemble methods such as Voting Classifier (RF + AdaBoost) and Stacking Classifier (RF + MLP with LightGBM), to improve detection performance. The ensemble approach combines the strengths of individual models to achieve higher robustness and accuracy. The proposed model is evaluated using benchmark datasets such as NSL-KDD and BoT-IoT. Experimental results demonstrate significant improvements in accuracy, precision, recall, and reduced execution time compared to traditional models. This approach provides an efficient and scalable solution for real-time intrusion detection in cloud environments.

Index terms - Cloud Computing, Intrusion Detection System (IDS), Ensemble Learning, Random Forest, Feature Engineering, Feature Selection, Stacking Classifier, Voting Classifier, Machine Learning, Deep Learning, LSTM, NSL-KDD, BoT-IoT, Cybersecurity

1. INTRODUCTION

Cloud computing has become a fundamental technology for delivering scalable and on-demand computing resources such as storage, networking, and processing power. However, the rapid growth of cloud environments has also increased the risk of cyber threats, including unauthorized access, data breaches, and distributed attacks. As a result, ensuring robust security mechanisms has become a critical requirement. Intrusion Detection Systems (IDS) play a vital role in monitoring network traffic and identifying abnormal activities to protect cloud infrastructures.

Traditional IDS approaches mainly rely on single machine learning models such as Random Forest, Decision Tree, Support Vector Machine, and Naïve Bayes. Although these models provide reasonable detection performance, they often suffer from limitations such as reduced accuracy, high false alarm rates, and inability to handle complex and large-scale cloud data. Moreover, many existing systems do not effectively utilize feature engineering and feature selection techniques, which are essential for reducing data dimensionality and improving model efficiency.

To address these challenges, this work proposes an enhanced cloud-based intrusion detection system using hybrid ensemble learning techniques combined with feature engineering. Initially, data preprocessing and visualization-based feature engineering are applied to identify and select the most relevant

features, thereby reducing redundancy and computational complexity. The proposed system integrates multiple models, including traditional machine learning algorithms and advanced deep learning approaches such as LSTM, to capture both static and sequential patterns in network data.

Furthermore, ensemble methods such as Voting Classifier (combining Random Forest and AdaBoost) and Stacking Classifier (integrating Random Forest, MLP, and LightGBM) are employed to leverage the strengths of individual models and improve overall detection performance. By combining multiple classifiers, the system achieves better generalization, higher accuracy, and improved robustness against diverse attack patterns.

The proposed approach is evaluated using benchmark datasets such as NSL-KDD and BoT-IoT. Experimental results demonstrate that the integration of feature engineering and hybrid ensemble models significantly enhances intrusion detection performance in terms of accuracy, precision, recall, and execution efficiency. This makes the system suitable for real-time and scalable cloud security applications.

2. LITERATURE SURVEY

1. Security in cloud computing: Opportunities and challenges

The promise of cloud computing to deliver powerful, elastic, affordable, and manageable resources instantly over the Internet is impressive. Through efficient and shared use, cloud computing increases the possibilities of physical resources. Organizations and individual users are encouraged to move their apps and services to the cloud by the aforementioned benefits. The cloud computing concept is even being used to essential infrastructure, such as electricity generating and distribution plants. However, there are extra security risks associated with the services offered by third-party cloud service providers. Security risks are raised when user assets (data, apps, etc.) are moved outside of administrative control in a shared environment where many users are collocated. The security concerns arising from the very nature of cloud computing are described in detail in this

survey. Additionally, the study offers the most recent approaches to addressing security concerns that have been documented in the literature. Additionally, a quick overview of security flaws in mobile cloud computing is also provided. Finally, a review of the unresolved problems and potential avenues for further research is provided.

2. Securing Cloud Data: A Machine Learning based Data Categorization Approach for Cloud Computing

A new methodical approach called cloud computing (CC) enables users to store data on distant computers that are reachable over the internet. This method makes it easier to access and transfer important and personal data. As a result, demand is rising daily. A variety of data, including financial transactions, paper-based information, and multimedia material, can be stored on this. Additionally, by lowering operating and maintenance expenses, CC lessens the services' reliance on local storage. Regardless of the degree of security of the data, existing systems encrypt all data using the same key size, which increases processing time and expense. Furthermore, none of these methods improve secrecy and categorize the data with a high accuracy rate. This study presents a cloud computing strategy for data sensitivity based on automated data categorization. The suggested model is based on three levels of sensitivity: basic, confidential, and extremely confidential. It is trained with automated feature extraction utilizing Random Forest (RF), Naïve Bayes (NB), k-nearest neighbor (KNN), and support vector machine (SVM) classifiers. According to simulation findings, the suggested model attained 92% accuracy. This leads us to the conclusion that RF, NB, and KNN outperform SVM.

3. A Novel Anomaly Network Intrusion Detection System for Internet of Things Security

The requirements of computer system security make the automated detection of intruders an intriguing problem. Both wired and wireless networks are susceptible to several harmful assaults. Intrusion detection technology uses anomaly detection, a more modern method, to identify intrusions and thwart assaults. In order to increase the detection rate in the

Internet of Things environment, this research presents a unique network intrusion detection model based on machine learning techniques. The suggested method aims to use network intrusion detection to enhance the data categorization problem. We have developed and verified a pertinent classifier to significantly fulfill the security requirement after outlining our technique.

4. Network intrusion detection: a comparative study of four classifiers using the NSL-KDD and KDD'99 datasets

Protecting online identities against dangers of confidentiality, integrity, and accessibility is becoming a more pressing issue as the majority of people get access to the internet. In order to highlight and categorize suspicious behavior, a network intrusion detection system (IDS) by definition assists in locating and identifying unusual network traffic. It is an essential component of network security and serves as the first line of defense against a suspected attack by warning an administrator or other relevant staff about potentially harmful network behavior. Numerous scholarly articles suggest different artificial intelligence (AI) techniques for a precise network intrusion detection system (IDS). In order to train two benchmark datasets, the KDD'99 and the NSL-KDD, this study describes and contrasts four AI techniques. We provide a straightforward yet efficient data preparation technique since, in addition to model selection, data pretreatment is essential to correct results. Additionally, we assess and contrast the performance and accuracy of four widely used models: a stacked autoencoder (SAE) model, decision trees (DT), random forests (RF), and multi-layer perceptrons (MLP). The random forest classifier produced the most reliable and accurate results out of the four techniques.

5. A Reliable Network Intrusion Detection Approach Using Decision Tree with Enhanced Data Quality

Due to the recent advancements in the Internet of things (IoT) and cloud computing technologies and growing number of devices connected to the Internet, the security and privacy issues are important to be resolved and protect the data and computer network.

Real-time network data and resource monitoring is necessary for security. Real-time monitoring, identification, and alerting of intrusion events have been made possible by intrusion detection systems. Many machine learning (ML) approaches have recently been included into intrusion detection systems (IDS). Decision trees are one method that may use trustworthy network metrics to make wise choices by raising the detection rate and accuracy. In this study, we provide a dependable decision tree-based method for detecting network intrusions with improved data quality. In particular, a decision tree classifier is constructed for dependable intrusion detection after network data pretreatment and entropy decision feature selection are performed to improve the data quality and pertinent training. The suggested model can produce reliable results, according to an experimental analysis on two datasets. In fact, our model's accuracy with the NSL-KDD and CICIDS2017 datasets is 99.42% and 98.80%, respectively. In terms of accuracy (ACC), detection rate (DR), and false alarm rate (FAR), the new method has several benefits over the existing methods.

3. METHODOLOGY

i) Proposed Work:

An improved cloud-based intrusion detection system utilizing hybrid ensemble learning and feature engineering approaches is presented in the proposed study. To find the most pertinent features and remove redundant or noisy data, the system first preprocesses the data before using visualization-driven feature engineering. By reducing features, detection efficiency is increased and computational complexity is decreased. In order to identify sequential patterns in network traffic, the revised dataset is then used to train many machine learning models, including Random Forest, Decision Tree, Support Vector Machine, and Naïve Bayes, as well as deep learning models like LSTM.

Advanced ensemble approaches, such as Voting Classifier (which combines Random Forest and AdaBoost) and Stacking Classifier (which integrates Random Forest, MLP, and LightGBM), are used to further enhance performance. By combining the

advantages of several classifiers, these ensemble models provide predictions that are more reliable and accurate. Benchmark datasets like NSL-KDD and BoT-IoT are used to assess the system, which reduces execution time while improving accuracy, precision, and recall. Because of this, the suggested approach is quite successful in detecting intrusions in cloud systems in real time.

ii) System Architecture:

The system architecture of the proposed cloud-based intrusion detection system consists of multiple sequential modules designed to ensure efficient data processing and accurate attack detection. Initially, the data exploration module is used to load datasets such as NSL-KDD and BoT-IoT into the system. This is followed by the data processing module, where preprocessing steps such as data cleaning, normalization, and feature engineering are performed. Visualization-based feature selection is applied to identify the most relevant features, reducing dimensionality and improving system performance. After preprocessing, the dataset is divided into training and testing sets using the data splitting module, enabling effective model evaluation.

In the next stage, the model generation module builds multiple machine learning and deep learning models, including Random Forest, Decision Tree, Support Vector Machine, Naïve Bayes, and LSTM. Advanced ensemble techniques such as Voting Classifier (RF + AdaBoost) and Stacking Classifier (RF + MLP with LightGBM) are implemented to enhance prediction accuracy and robustness. The system also includes a user interface module for user registration, login, and input submission. Finally, the prediction module processes user input through the trained ensemble model and displays the detection result (normal or attack). This layered architecture ensures scalability, reduced execution time, and improved detection performance in cloud environments.

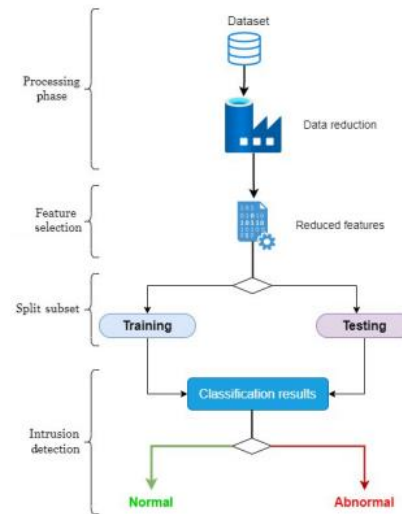


Fig.1. Proposed Architecture

iii) MODULES:

1. Data Exploration

This module is responsible for loading datasets such as NSL-KDD and BoT-IoT into the system. It helps in understanding the structure, distribution, and characteristics of the data through basic analysis and visualization, which is essential before preprocessing and model building.

2. Data Processing

In this module, the raw data is cleaned and transformed using preprocessing techniques such as handling missing values, normalization, and encoding. Additionally, feature engineering and feature selection are applied to reduce irrelevant features and improve the efficiency and accuracy of the model.

3. Data Splitting (Train & Test)

This module divides the processed dataset into training and testing subsets. The training data is used to build the models, while the testing data is used to evaluate their performance, ensuring that the model generalizes well to unseen data.

4. Model Generation

This module builds multiple machine learning and deep learning models such as Random Forest, Decision Tree, Support Vector Machine, Naïve Bayes, and LSTM. It also implements advanced ensemble techniques like Voting Classifier (RF + AdaBoost) and Stacking Classifier (RF + MLP with LightGBM) to improve prediction accuracy and robustness.

5. User Signup & Login

This module manages user registration and authentication. It allows users to securely create accounts and log in to access the intrusion detection system, ensuring controlled and authorized usage of the application.

6. User Input

This module allows users to provide input data for prediction. The input is processed and formatted according to the trained model requirements so that it can be effectively used for intrusion detection.

7. Prediction

This is the final module where the trained ensemble model is applied to the user input. The system analyzes the data and displays the prediction result, indicating whether the activity is normal or an intrusion, along with performance metrics if required.

iv) ALGORITHMS:

1. Random Forest (RF)

Random Forest is an ensemble learning algorithm that builds multiple decision trees using random subsets of data and features. The final prediction is made by combining the outputs of all trees (majority voting), which improves accuracy and reduces overfitting in intrusion detection.

2. Decision Tree (DT)

Decision Tree is a supervised learning algorithm that creates a tree-like structure of decisions based on feature values. It splits the data into branches to reach a final classification, making it simple, interpretable, and effective for detecting patterns in network traffic.

3. Support Vector Machine (SVM)

Support Vector Machine is a supervised algorithm that identifies the optimal boundary (hyperplane) to separate different classes of data. It is highly effective for handling complex and high-dimensional datasets, making it suitable for intrusion detection tasks.

4. Naïve Bayes (NB)

Naïve Bayes is a probabilistic classifier based on Bayes' theorem with an assumption of feature independence. It is fast and efficient, making it suitable for large-scale datasets and real-time intrusion detection applications.

5. Voting Classifier (RF + AdaBoost)

Voting Classifier is an ensemble method that combines predictions from multiple models such as Random Forest and AdaBoost. The final output is

determined based on majority voting, which enhances prediction accuracy and model stability.

6. Stacking Classifier (RF + MLP with LightGBM)

Stacking Classifier is an advanced ensemble technique where multiple base models (RF and MLP) are combined, and their outputs are passed to a meta-learner (LightGBM). This approach improves performance by learning the best combination of predictions from different models.

7. Multi-Layer Perceptron (MLP)

MLP is a type of artificial neural network consisting of multiple layers of neurons that learn complex nonlinear relationships in data. It is useful for capturing hidden patterns in intrusion detection datasets.

8. AdaBoost

AdaBoost is a boosting algorithm that improves model performance by giving more importance to misclassified data points in each iteration. It combines weak learners into a strong model, enhancing detection capability.

9. LightGBM

LightGBM is a gradient boosting framework that builds decision trees efficiently with faster training speed and lower memory usage. It is suitable for handling large-scale datasets in cloud-based intrusion detection systems.

10. Long Short-Term Memory (LSTM)

LSTM is a specialized recurrent neural network designed to learn long-term dependencies in sequential data. It is particularly useful for detecting temporal patterns and anomalies in network traffic over time.

4. EXPERIMENTAL RESULTS

The proposed cloud-based intrusion detection system was evaluated using benchmark datasets such as NSL-KDD and BoT-IoT to validate its effectiveness. Initially, multiple machine learning models including Random Forest, Decision Tree, Support Vector Machine, and Naïve Bayes were trained and tested. Further, deep learning (LSTM) and hybrid ensemble techniques such as Voting Classifier (RF + AdaBoost) and Stacking Classifier (RF + MLP with LightGBM) were implemented to enhance detection performance. The integration of feature engineering

and feature selection significantly reduced data dimensionality, leading to faster execution and improved model efficiency. The results demonstrate that ensemble models outperform individual models in terms of accuracy, precision, and recall.

From the system interface results, it is observed that the model successfully predicts network behavior based on user input parameters. The prediction output clearly distinguishes between normal and abnormal activities, as shown in the result screen indicating “No attack detected in the cloud.” This confirms the model’s capability to accurately classify network traffic in real-time scenarios. Overall, the proposed hybrid ensemble approach achieves higher detection accuracy, reduced false positives, and better generalization compared to traditional intrusion detection methods, making it suitable for scalable and real-time cloud security applications.

Accuracy: The ability of a test to differentiate between healthy and sick instances is a measure of its accuracy. Find the proportion of analysed cases with true positives and true negatives to get a sense of the test's accuracy. Based on the calculations:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Accuracy} = \frac{(TN + TP)}{T}$$

Precision: The accuracy rate of a classification or number of positive cases is known as precision. Accuracy is determined by applying the following formula:

$$\text{Precision} = \frac{\text{True positives}}{(\text{True positives} + \text{False positives})} = \frac{TP}{(TP + FP)}$$

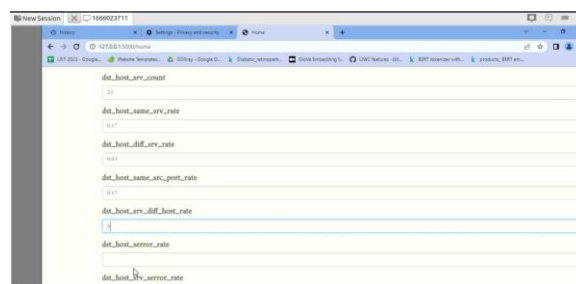
$$\text{Precision} = \frac{TP}{(TP + FP)}$$

Recall: The recall of a model is a measure of its capacity to identify all occurrences of a relevant machine learning class. A model's ability to detect class instances is shown by the ratio of correctly predicted positive observations to the total number of positives.

$$\text{Recall} = \frac{TP}{(FN + TP)}$$

F1-Score: A high F1 score indicates that a machine learning model is accurate. Improving model accuracy by integrating recall and precision. How often a model gets a dataset prediction right is measured by the accuracy statistic..

$$F1 = 2 \cdot \frac{(\text{Recall} \cdot \text{Precision})}{(\text{Recall} + \text{Precision})}$$



det_host_srv_count	21
det_host_name_srv_rate	0.27
det_host_diff_srv_rate	0.01
det_host_name_src_port_rate	0.01
det_host_diff_src_port_rate	0.01
det_host_srv_diff_host_rate	0
det_host_server_rate	0
det_host_srv_server_rate	0

Fig2 input values

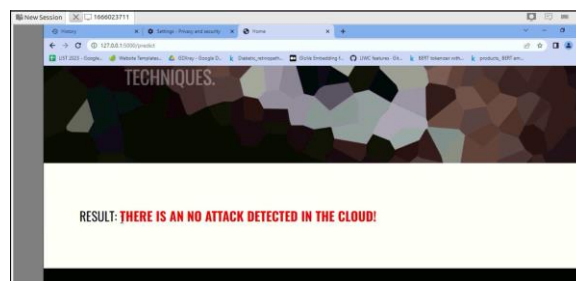


Fig.3. predicted results

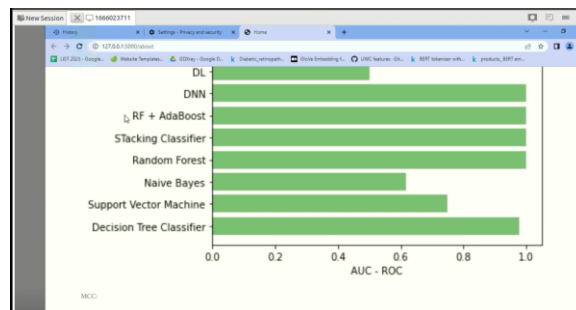


Fig.4. Accuracy graph

5. CONCLUSION

This paper presents an enhanced cloud-based intrusion detection system using hybrid ensemble learning and feature engineering techniques. The proposed approach integrates multiple machine learning and deep learning models, along with advanced ensemble methods such as Voting and Stacking, to improve detection performance. Feature engineering and feature selection effectively reduce data dimensionality, resulting in faster execution and improved model accuracy.

Experimental results demonstrate that the proposed system achieves higher accuracy, precision, and recall compared to traditional models while reducing false positives. The ability to accurately classify network traffic as normal or abnormal makes the system suitable for real-time cloud security applications. Overall, the proposed model provides a robust, scalable, and efficient solution for detecting intrusions in cloud environments.

6. FUTURE SCOPE

The proposed system can be further enhanced by exploring more advanced deep learning architectures such as Transformers and hybrid CNN-LSTM models to improve the detection of complex and evolving cyber-attacks. Additionally, incorporating real-time streaming data processing using technologies like Apache Kafka or Spark can enable faster and more scalable intrusion detection in dynamic cloud environments.

Further improvements can be achieved by applying advanced optimization and hyperparameter tuning techniques such as Bayesian Optimization or Genetic Algorithms to enhance model performance. The system can also be extended to support multi-class attack classification instead of binary detection, along with deployment in real-world cloud platforms for practical validation. Integrating explainable AI (XAI) techniques will also help in improving the transparency and trustworthiness of the intrusion detection system.

REFERENCES

- [1] M. Ali, S. U. Khan, and A. V. Vasilakos, Security in cloud computing: Opportunities and challenges, *Information Sciences*, vol. 35, pp. 357–383, 2015.
- [2] A. Singh and K. Chatterjee, Cloud security issues and challenges: A survey, *Journal of Network and Computer Applications*, vol. 79, pp. 88–115, 2017.
- [3] P. S. Gowr and N. Kumar, Cloud computing security: A survey, *International Journal of Engineering and Technology*, vol. 7, no. 2, pp. 355–357, 2018.
- [4] A. Verma and S. Kaushal, Cloud computing security issues and challenges: A survey, in *Proc. First International Conference on Advances in Computing and Communications*, Kochi, India, 2011, pp. 445–454.
- [5] H. Alloussi, F. Laila, and A. Sekkaki, L'état de l'art de la sécurité dans le cloud computing: Problèmes et solutions de la sécurité en cloud computing, presented at *Workshop on Innovation and New Trends in Information Systems*, Mohamadia, Maroc, 2012.
- [6] J. Gu, L. Wang, H. Wang, and S. Wang, A novel approach to intrusion detection using SVM ensemble with feature augmentation, *Computers and Security*, vol. 86, pp. 53–62, 2019.
- [7] Z. Chiba, N. Abghour, K. Moussaid, A. E. Omri, and M. Rida, A cooperative and hybrid network intrusion detection framework in cloud computing based snort and optimized back propagation neural network, *Procedia Computer Science*, vol. 83, pp. 1200–1206, 2016.
- [8] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, Survey of intrusion detection systems: Techniques, datasets and challenges, *Cybersecurity*, vol. 2, p. 20, 2019.
- [9] A. Guezzaz, A. Asimi, Y. Asimi, Z. Tbatou, and Y. Sadqi, A global intrusion detection system using PcapSockS sniffer and multilayer perceptron classifier, *International Journal of Network Security*, vol. 21, no. 3, pp. 438–450, 2019.
- [10] A. Guezzaz, S. Benkirane, M. Azrou, and S. Khurram, A reliable network intrusion detection

approach using decision tree with enhanced data quality, *Security and Communication Networks*, vol. 2021, p. 1230593, 2021.

[11] B. A. Tama and K. H. Rhee, HFSTE: Hybrid feature selections and tree-based classifiers ensemble for intrusion detection system, *IEICE Trans. Inf. Syst.*, vol. E100.D, no. 8, pp. 1729–1737, 2017.

[12] M. Azrour, J. Mabrouki, G. Fattah, A. Guezzaz, and F. Aziz, Machine learning algorithms for efficient water quality prediction, *Modeling Earth Systems and Environment*, vol. 8, pp. 2793–2801, 2022.

[13] M. Azrour, Y. Farhaoui, M. Ouanan, and A. Guezzaz, SPIT detection in telephony over IP using K-means algorithm, *Procedia Computer Science*, vol. 148, pp. 542–551, 2019.

[14] M. Azrour, M. Ouanan, Y. Farhaoui, and A. Guezzaz, Security analysis of Ye et al. authentication protocol for internet of things, in *Proc. International Conference on Big Data and Smart Digital Environment*, Casablanca, Morocco, 2018, pp. 67–74.

[15] M. Azrour, J. Mabrouki, A. Guezzaz, and A. Kanwal, Internet of things security: Challenges and key issues, *Security and Communication Networks*, vol. 2021, p. 5533843, 2021.