Network Intrusion Detection System

Mr. T. Abdhul Raheem[1], Gajula Ranganath[2], Meda Vamshi Krishna[3], Peddathumbalam Sadiq Basha[4], Arekanti Francis[5]

[1 3 4 5]*Dept of CSE(DS), St. Johns College of Engineering and Technology, Yerrakota, Yemmiganur, Kurnool, AP, Affiliated by JNTUA, India*

[1] *M. tech, Assistant Professor Department of CSE,*

Abstract— The rapid growth of cloud computing has significantly increased the volume, velocity, and complexity of network traffic, creating new challenges for effective intrusion detection. Traditional signature-based and statistical anomaly-based Intrusion Detection Systems (IDS) are limited in detecting zero-day attacks, handling evolving threat patterns, and maintaining scalability in dynamic multi-tenant cloud environments. To address these limitations, this paper proposes an optimization-driven machine learning framework for cloud-based intrusion detection. The proposed system integrates embedded feature selection to reduce high-dimensional traffic attributes, Synthetic Minority Over-sampling Technique (SMOTE) to mitigate severe class imbalance, and a hybrid classification architecture combining Random Forest (RF) and Bi-directional Long Short-Term Memory (Bi-LSTM) networks. The feature selection mechanism enhances computational efficiency by eliminating redundant attributes, while synthetic sampling improves detection capability for minority attack classes. The hybrid model leverages ensemble-based classification and temporal sequence learning to capture complex traffic behaviours associated with multi-stage attacks. Experimental evaluation conducted on the CIC-IDS2017 benchmark dataset demonstrates strong classification performance with improved recall and reduced false positive rates, supporting the suitability of the proposed framework for real-time intrusion detection in cloud environments.

***Index Terms*— Cybersecurity, Cloud Computing, Intrusion Detection System, Machine Learning, Feature Selection, SMOTE, Random Forest, Bi-directional Long Short-Term Memory, CIC-IDS2017.**

I.INTRODUCTION

Cloud infrastructures generate high-volume dynamic traffic, increasing the complexity of intrusion detection. By offering services through Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), cloud computing enables organisations to deploy applications rapidly, scale resources dynamically, and reduce capital and operational costs. As a result, cloud platforms have become essential infrastructure for both enterprises and individuals. Despite its advantages, cloud computing introduces critical security challenges. Cloud environments are highly dynamic, rely extensively on virtualisation, and involve shared resources among multiple tenants. These characteristics significantly expand the attack surface and make traditional perimeter-based security mechanisms ineffective. Attackers exploit misconfigured services, insecure APIs, weak authentication mechanisms, and lateral movement within cloud networks. Additionally, the large volume and high velocity of cloud network traffic make manual monitoring impractical. Intrusion Detection Systems play a vital role in monitoring network traffic and identifying malicious activities. Traditional IDS approaches, such as signature-based detection, are effective only for known attack patterns and fail to detect novel or zero-day attacks. Anomaly-based IDS attempt to overcome this limitation by modelling normal behaviour, but they often generate excessive false positives in dynamic cloud environments. These limitations highlight the need for intelligent, adaptive, and scalable intrusion detection mechanisms. Machine Learning has gained significant attention in cybersecurity due to its ability to learn complex patterns from large-scale data. However, the direct application of ML models to cloud traffic presents challenges such as high-dimensional feature spaces, severe imbalance between benign and malicious traffic, and high computational costs that limit real-time deployment. This research addresses these challenges by

proposing an optimised ML-based IDS framework tailored for cloud environments.

II. LITERATURE REVIEW

Intrusion detection research has progressed from traditional signature-based systems, which detect known attack patterns but are ineffective against zero-day and polymorphic threats [2], to anomaly-based approaches that identify deviations from normal traffic behaviour but often generate high false positive rates in dynamic and elastic cloud environments [5]. To address these shortcomings, supervised machine learning techniques such as Decision Trees, Support Vector Machines, k-Nearest Neighbours, and Random Forests have been widely applied to enhance detection accuracy and scalability [6]. Among these, Random Forest demonstrates strong robustness, resistance to overfitting, and effective handling of high-dimensional data [9]; however, supervised approaches remain sensitive to class imbalance, where benign traffic significantly outweighs malicious samples, leading to biased classification performance. Deep learning architectures, particularly LSTM-based models, have further improved intrusion detection by capturing temporal dependencies and sequential traffic patterns associated with multi-stage attacks [4]. Despite their improved accuracy, deep learning methods introduce higher computational complexity and may not always meet real-time deployment constraints in cloud infrastructures. Consequently, recent studies emphasize the integration of feature selection techniques to reduce dimensionality and synthetic sampling strategies such as SMOTE to address class imbalance and improve minority attack detection [8]. Nevertheless, many existing frameworks treat these optimization components independently rather than as part of a unified architecture, highlighting the necessity for an integrated optimization-driven intrusion detection framework tailored for cloud environments.

III. PROPOSED SYSTEM ARCHITECTURE

The proposed Intrusion Detection System follows a multistage architecture designed to improve detection accuracy while minimising computational overhead. The system architecture consists of data acquisition, preprocessing, feature selection, class balancing, and hybrid classification stages. Network traffic data is initially collected from benchmark datasets that represent realistic cloud environments. The data is then pre-processed to remove noise, handle missing values, and normalise feature scales. Embedded feature selection using Random Forest feature importance is applied to identify the most relevant features and eliminate redundant or irrelevant attributes. To address the severe class imbalance inherent in intrusion detection datasets, SMOTE is applied exclusively to the training data. This technique generates synthetic samples for minority attack classes, enabling the model to learn attack patterns more effectively. The final detection stage employs a hybrid classification model that combines Random Forest and Bidirectional LSTM. A cascading classification strategy is adopted to balance accuracy and efficiency. Random Forest performs rapid classification for high-confidence samples, while Bi-LSTM analyses low-confidence samples to capture temporal dependencies. This design ensures high detection accuracy while maintaining real-time feasibility.

IV. METHODOLOGY

A. Dataset Description

The CIC-IDS2017 dataset. contains over 80 traffic features and multiple attack categories including DDoS, Port Scan, Botnet, and Brute Force attacks. It was selected due to its realistic traffic generation methodology and balanced representation of modern cloud-based attack patterns.

B. Data Preprocessing

Data preprocessing is a critical step in the proposed framework. It involves removing missing and infinite values, encoding categorical labels into numerical form, and normalising feature values using standard scaling. A stratified train-test split is performed to ensure consistent class distribution across training and testing datasets.

C. Feature Selection

Embedded feature selection is performed using Random Forest feature importance scores. This approach identifies features that contribute most significantly to classification decisions. Selecting the

top-ranked features reduces dimensionality, improves model efficiency, and mitigates overfitting.

D. Class Balancing

SMOTE is applied to the training dataset to address class imbalance. By generating synthetic samples for minority attack classes, SMOTE improves recall and detection capability for rare attacks without discarding valuable benign data.

E. Hybrid Classification

The hybrid RF–Bi-LSTM model combines fast ensemble based classification with deep temporal analysis. Random Forest provides quick and robust predictions, while Bi-LSTM captures sequential patterns in network traffic. This hybrid approach leverages the strengths of both models.

V. EXPERIMENTAL RESULTS

The experimental evaluation of the proposed Machine Learning based Intrusion Detection System was conducted to validate its effectiveness in detecting malicious activities within cloud environments while maintaining real-time performance. The experiments were performed using the CIC-IDS2017 dataset, which contains realistic benign and attack traffic representing modern network conditions. The dataset was divided into training and testing subsets using a stratified split to preserve class distribution. Performance evaluation was carried out using standard metrics such as accuracy, precision, recall, F1score, and false positive rate, which are widely used to assess intrusion detection systems. The proposed hybrid Random Forest–Bi-directional LSTM model was compared against baseline classifiers, including Decision Tree and kNearest Neighbours, to demonstrate the effectiveness of the optimisation framework. The proposed RF–BiLSTM model achieved 98.6% accuracy, 97.9% recall, and 98.0% F1-score. The false positive rate was reduced to 1.2%, outperforming Decision Tree (94.2%) and KNN (95.8%) classifiers.

A. Input Design

The input design of the proposed intrusion detection system focuses on efficiently handling large-scale network traffic data generated in cloud environments. The primary input to the system consists of network flow records extracted from the CIC-IDS2017 dataset, which include numerical features such as flow duration, packet length statistics, byte counts, interarrival times, and protocol-related attributes. Prior to model training, the input data undergoes preprocessing steps including removal of missing and infinite values, normalisation of numerical features, and encoding of class labels into a machine readable format. Feature selection is then applied to retain only the most relevant attributes, thereby reducing dimensionality and improving computational efficiency. The final processed feature vectors serve as input to the hybrid classification model, ensuring that the system receives clean, informative, and optimised data for accurate intrusion detection.

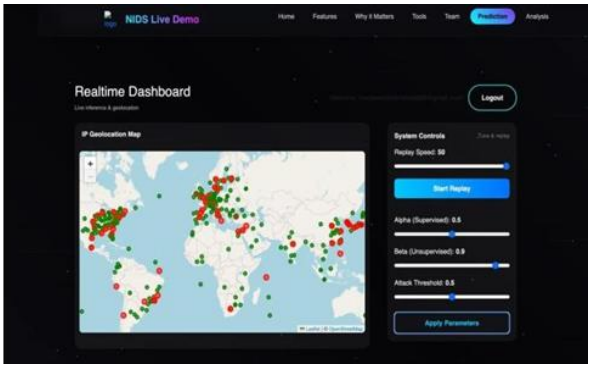
B. Objectives

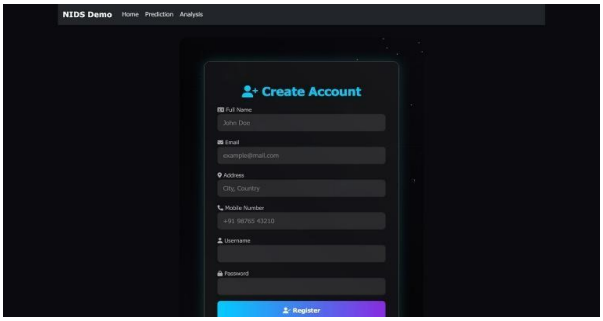
The primary objective of the experimental evaluation is to assess the effectiveness of the proposed IDS in accurately distinguishing between benign and malicious network traffic under realistic cloud conditions. Specifically, the experiments aim to evaluate the detection accuracy of the hybrid model, measure its ability to identify minority attack classes through recall and F1-score, and analyse the false positive rate to ensure reduced alert fatigue. Another key objective is to evaluate the computational efficiency of the system by analysing inference time and overall processing overhead, which are critical factors for real-time cloud deployment. Additionally, the experiments aim to validate the impact of feature selection and SMOTE-based class balancing on overall system performance by comparing results with baseline models that do not employ these optimisation techniques.

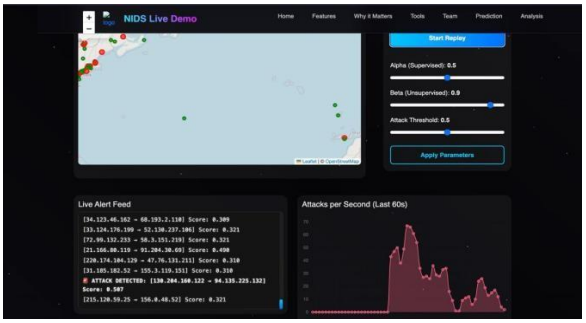
C. Output Design

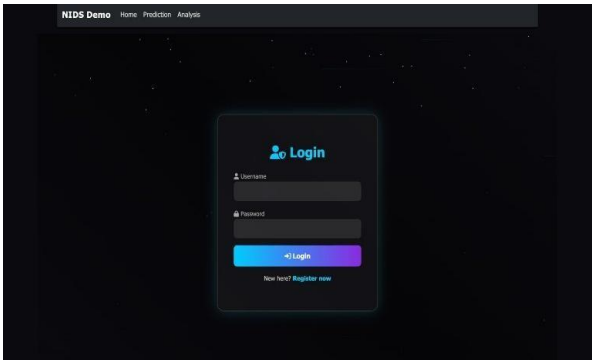
The output design of the proposed intrusion detection system is structured to provide clear and actionable results for security administrators. The primary output of the system is a classification label indicating whether the incoming network traffic is benign or malicious, along with the corresponding attack category when applicable. In addition to classification results, the system generates performance metrics such as accuracy, precision,

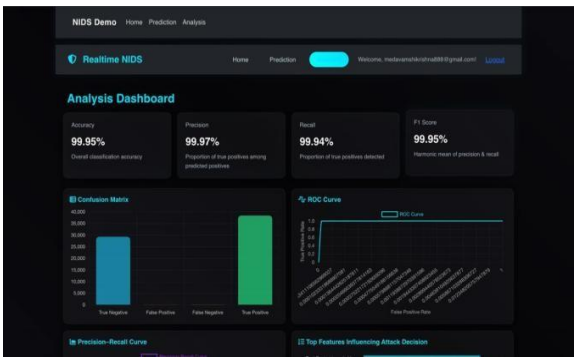
recall, F1-score, and false positive rate to support evaluation and monitoring. These outputs can be visualised through tabular summaries and graphical representations such as confusion matrices and performance charts. The output design ensures that the system not only detects intrusions accurately but also presents results in a clear and interpretable manner, facilitating effective decision-making and timely response in cloud security environments.

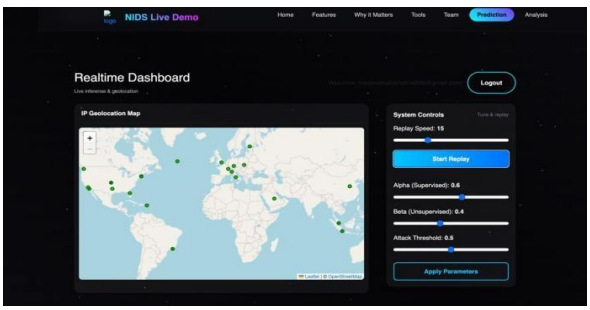


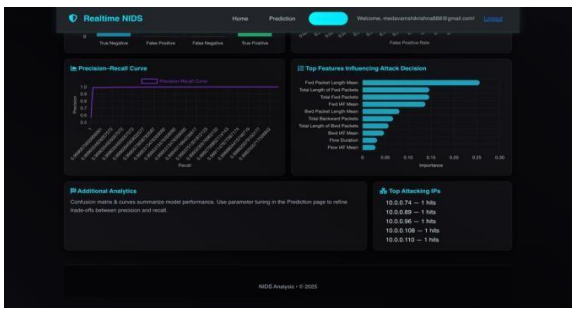












VI. SYSTEM TESTING

System testing was conducted to ensure the correctness, reliability, and robustness of the proposed Machine Learning based Intrusion Detection System under realistic cloud-like conditions. The testing process included unit testing to verify the functionality of individual modules such as data preprocessing, feature selection, class balancing, and classification, followed by integration testing to validate seamless interaction between these components. Black-box testing was employed to evaluate system behaviour using various input traffic scenarios without internal knowledge of implementation, while white-box testing was used to verify logical correctness and control flow within the system. Stress testing was performed by supplying large volumes of network traffic data to assess system stability and response time under high-load conditions, and performance testing evaluated detection accuracy, false positive rate, and inference time to ensure real-time feasibility. The testing results confirmed that the system operates accurately and efficiently, maintains stability under heavy traffic conditions, and meets the functional and performance requirements for deployment in cloud security environments.

VII. CONCLUSION

An optimization-driven machine learning framework for cloud-based intrusion detection was developed by integrating embedded feature selection, SMOTE-based class balancing, and a hybrid Random Forest–BiLSTM architecture. The feature selection mechanism reduced dimensionality and computational overhead, enabling efficient handling of high-volume network traffic data, while synthetic oversampling improved detection of minority attack classes without degrading benign traffic classification. By combining the rapid decision capability of Random Forest with the temporal sequence modelling strength of Bi-directional LSTM, the proposed system enhanced detection reliability and reduced false alarms. Experimental evaluation on the CIC-IDS2017 dataset demonstrated strong classification performance with improved recall and low false positive rates, supporting the feasibility of deploying the model for real-time intrusion detection in dynamic cloud environments. Overall, the integration of optimization techniques with hybrid learning models provides a scalable and effective solution for strengthening cloud network security.

VIII. FUTURE SCOPE

The proposed intrusion detection framework can be further enhanced in several directions to improve its applicability and effectiveness in real-world cloud environments. Future work may focus on deploying the system in live cloud infrastructures to enable real-time traffic monitoring and automated threat response. Advanced deep learning architectures such as transformer-based models and graph neural networks can be explored to capture complex traffic relationships and lateral movement attacks. Incorporating explainable artificial intelligence techniques would help security analysts understand detection decisions and improve trust in the system. Additionally, federated learning can be adopted to preserve data privacy across multi-tenant cloud environments while enabling collaborative model training. Continuous learning and automated retraining mechanisms can also be integrated to address evolving attack patterns and concept drift, ensuring long-term adaptability and robustness of the intrusion detection system.

REFERENCES

[1] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” in Proc. Int. Conf. Information Systems Security and Privacy (ICISSP), 2018, pp. 108–116.

[2] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, “A detailed analysis of the KDD CUP 99 data set,” in Proc. IEEE Symp. Computational Intelligence for Security and Defense Applications, 2009, pp. 1–6.

[3] Y. K. Saheed, A. Alzahrani, and M. A. Khan, “A machine learning-based intrusion detection system for IoT network attacks,” Sensors, vol. 22, no. 9, pp. 1–21, 2022.

[4] X. Xu, L. Zhang, and Q. Chen, “CNN–BiLSTM based network intrusion detection for cybersecurity applications,” Expert Systems with Applications, vol. 233, 2024.

[5] N. Wankhade and A. Khandare, “Optimization of deep generative intrusion detection system for

cloud computing," IEEE Access, vol. 11, pp. 1–14, 2023.
[6] J. Kshirsagar and S. Kulkarni, "Feature selection and classification techniques for network intrusion detection systems," International Journal of Information Security, vol. 22, no. 3, pp. 345–359, 2023.
[7] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in Proc. IEEE Int. Joint Conf. Neural Networks, 2008, pp. 1322–1328.
[8] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321–357, 2002.
[9] L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.