Deep Learning–based IDS framework for Cloud Data Security

Bhavna Gangwar[1], Nupa Ram Chauhan[2]

[1]Research Scholar, Teerthanker Mahaveer University, Moradabad
gangwarbhavna7@gmail.com
[2]Associate Professor, Teerthanker Mahaveer University, Moradabad
nrcua80@gmail.com

Abstract. Cloud computing's scalable, adaptable, and cost-efficient processing and storage capabilities have completely transformed big data management. However, moving sensitive data into dispersed cloud environments presents significant privacy and security issues, such as advanced persistent threats, data breaches, and illegal access. This paper proposed an integrated framework that integrates access control and a hybrid deep learning and machine learning-based intrusion detection system (IDS) for real-time threat detection to improve cloud data security. The framework includes data preprocessing, advanced SMOTE for balancing classes, and hybrid feature learning using 1D Residual Autoencoders (RAE) and Convolutional Neural Networks (CNN).The experimental set up is processed on benchmark datasets (UNSB-NB15, NSL-KDD, and WSN-DS), which shows that the model can successfully detect both frequent and rare attacks with an accuracy of 97.6%, 95% and 95.68%, respectively.The integration of intelligent intrusion detection with cloud data security ensures multi-layered security, increasing confidentiality, integrity and availability across cloud infrastructure. Future research will focus on integrating federated learning and blockchain-based trust management to enable decentralized, privacy-preserving, and adaptable cloud security solutions.

Keywords: XGBoost,LightGBM,CatBoost,Intrusion Detection System,Convolutional Neural Network
.

1 Introduction

As conventional systems quickly converge with digital technologies, cloud computing has become a transformative pillar of modern information infrastructure. That is, not only has digital technology transformed cloud computing, but new advances in automotive networking allow more data with more variety to be transmitted faster in distributed environments. Heterogeneous devices, sensors, and endpoints are now routinely exchanging data in real time across diverse networks in a cloud environment.Network security is now dually important and disruptive with automation, accountability and service scalability, but also new ways to facilitate cyber attack.

In terms of defending against today's sophisticated cyber attacks, illegal access, and data hacks, intrusion detection systems, play a pivotal role in these environments. However, in the real world significant advancements cannot be made with traditional IDS methods that also have limitations. Not only are IDS systems extremely sensitive to redundant and noisy data, they also diminish detection capabilities.Many existing systems exhibit unequal class distributions that make training models more challenging because there are often many more benign examples compared to malicious examples. Another limitation of traditional IDS models is their inability to locate and identify temporal and sequential dependencies within network traffic, leading to an inability to detect multi-step sophisticated attacks. [1]

One of the great challenges in network traffic management is a noise and redundancy in the data generated. Network data generated in the cloud creates volumes of data that is often noisy due to varying traffic patterns, embedded telemetry anomalies, and heterogeneous sources. Failure to manage these redundancies can result in increased computational costs and a decrease in IDS performance. Class imbalance is another issue that has become well known, and benchmark datasets reported typically exhibit a much larger benign or particular traffic subset than malicious gameplay. Real-world effectiveness of detection is ultimately compromised even with higher true positive rates of accuracy in models, however this skewed data distribution creates lower misses for the minority attack class. The presence of sequential dependencies in complicated attack patterns may be another aspect of detection failures in traditional models. Sophisticated attacks like multi-stage intrusion and advanced persistent threats (APTs) evolve over time and require profile memory and temporal awareness that IDSs possess in order to accurately identify that something is happening.

In order to enhance detection capabilities for cloud-based and distributed environments, this paper proposes a new hybrid IDS that employs machine learning and deep learning methodologies to improve detection. The suggested technique is created to appropriately mitigate the problem associated with imbalanced datasets, accordingly model sequential attack behavior characteristics of attacks, reduce noise effects, and improve classification awareness. As a means of improving dataset imbalance and ensuring balanced training, the more advanced counterpart of SMOTE generates synthetic minority

class samples to create realistic minority class dataset samples.In order to improve learning efficiency, a one-dimensional Residual Autoencoder (1D-RAE) is employed to compress high-dimensional data, while addressing noise.This hybrid IDS framework aims to provide a highly scalable, generalizable, and efficient solution to meet the needs of current cybersecurity across various network infrastructures. Based on the experimental evaluation on several benchmark datasets, the proposed method has also demonstrated low false acceptance rates while maintaining high accuracy in its detection capabilities.The goal of this study is to offer a scalable, high-performance IDS to leverage high dimensional, diverse datasets in a cloud environment.[2]

2 Background Literature

Cloud environments are rapidly increasing in size and complexity, drawing attention to the inadequacies of misconceptions of traditional intrusion detection systems (IDS). The data associated with cloud traffic can be high-dimensional, heterogeneous, and frequently unbalanced; therefore, advanced analytical models are required to handle large-scale data with high levels of accuracy. While recent developments in the context of machine learning, deep learning, and hybrid IDS configurations result in better detection, issues such as scalability, modelling temporal dependency, preserving privacy, and robustness have not been fully resolved.

As cybersecurity threats targeting cloud infrastructures continue to grow and evolve, traditional protection mechanisms are becoming less effective. In the earlier cloud security models, perimeter-based defenses (e.g., firewalls, authentication mechanisms, and access control lists) were used mainly for cloud security.These techniques worked against basic threats; however, they don't work against advanced, stealth and persistent threats. Attackers are focusing on vulnerabilities found in virtualized resources, APIs, multi-tenant architectures, and dynamic cloud systems. Because of these, intrusion detection systems have become an important and necessary part of cloud security frameworks. IDS monitor system behavior and network traffic to discover malicious activities, but traditional signature and anomaly-based systems are still limited by false positives in systems and detection systems of zero-day or developing attacks.

Deep learning techniques have garnered immense focus since they can automatically learn hierarchical and discriminative features from raw data, bypassing the manual feature engineering that traditional machine learning techniques would require. This makes DL approaches particularly well-suited to cloud intrusion detection, where patterns are complex, nonlinear, and correlated with time. Nevertheless, DL-based IDS in cloud computing environments remain hamstrung by several challenges that involve computational overhead, overfitting, data imbalance, privacy restriction, and scaling in actual cloud systems.

A promising way to address the limitations of traditional intrusion detection system (IDS) architectures for cloud security frameworks is to employ deep learning (DL) techniques. Because cloud environments are characterized by large, high-velocity, heterogeneous network traffic, IDS must be able to identify and warn of sophisticated attack patterns, as well as detect rapidly developing new threats. Deep learning satisfies these conditions by automatically identifying useful and hierarchical feature representations of raw or pre-processed traffic, thus obviating any reliance on manual feature engineering. Recent work has shown that 1D Residual Autoencoders are particularly well-suited to learning deep latent representations from high dimensional cloud network traffic. Their residual connections can be used to minimize vanishing gradient problems and prevent relevant information loss, and thus advance robust feature learning in the presence of imbalanced datasets. Thus, when this deep, discriminative features are fused with advanced gradient boosting ensemble models, classification accuracy increases significantly thereby offering a intelligent and scalable strategy for intrusion detection on cloud systems.

Combined with traditional machine learning classifiers, deep learning models can provide an optimal balance between strong feature representation, and classification that is interpretable. Within several of the most recent developments, a deep learning architecture is first used to transform, denoise, and enhance features, and then machine learning models are used to classify cloud traffic across multiple attack classes with strong performance. Following from this, these hybrid DL-ML systems have performed well in threat detection - particularly for low frequency, covert, and newer intrusions that would be difficult for traditional signature-based IDS systems to identify. This is mainly due to the pattern capturing capabilities of deep learning networks that encapsulate rich and latent behavior and the interpretability of machine learning classifiers that provide fast and accurate decision boundaries. In relation to these continuing efforts, the remainder of this section reviews some recent additions to the ever-growing literature that identify new algorithms, hybrid IDS architectures, and distributed learning

methods to improve threat detection in cloud-centric and distributed methodology.

In a study conducted by, Almabadi and Ullah and et. al. tried to examined the potential for knowledge distillation to optimize cloud–edge intrusion detection systems. Following their approach, lightweight edge models could maintain high detection performance by learning knowledge from a relatively advanced teacher model hosted in a cloud environment for enhanced efficient, privacy-focused, and minimal communication[5].Similarly,Yu and et al. had proposed an Enhanced Heuristic Optimizer for feature selection, contributing to adjusted performance by dimensionality reduction of features, while improving detection performance and false-alarm rates, via adaptively tuning parameters with a heuristic search[6]. However, the study offered no validation with real-time data either in high-dimensional or encrypted cloud traffic scenarios. In the same year, Farrukh and et al. prototyped dual-modality IDS referred to as XG-NID, integrating innovative large language models (LLMs) and diverse graph neural networks (GNNs) to represent contextual features and structural patterns in cloud network data that reflected improved performance when identifying intrusions[7]. Adding to this trend, Najafli and et al. took a step forward by creating an RL-based hybrid IDS that supports the fog-to-cloud continuum and can make decisions based on the situation at hand to improve the quality of detection performance in dynamic and distributed systems.[8]

Chowdary and et al. suggested a hybrid intrusion detection model that integrates various analytic methods to improve detection accuracy while significantly reducing false positives. Their model showed promising performance across benchmark datasets and improved scalability and adaptability across multiple types of attack. However, the evaluation limited their model to simulated cloud environments and do not confirm the model validation in a real-world or real-time cloud deployments[9]. Long and et al. developed a transformer-based intrusion detection system IDS for cloud environments and utilized attention mechanisms to model complex spatial–temporal behavior in network traffic. Their paper showed high detection accuracy and a strong level of resilience against new threats. However, the model comes with a significant computational burden on the model during training and inference[10].Similarly, Al-Ghuwairi and et al. created a machine-learning framework for time-series anomaly detection in cloud intrusion situations.Their method successfully captured temporal changes in victim-side behaviors to identify attacks in their early stages; however, as victim-side and network traffic rapidly increased or was highly volatile, its detection performance declined[11]. Similarly, Nizamudeen and et al. ,concurrently proposed a deep learning-based IDS that utilized swarm intelligence, allowing a set of distributed intelligent agents to collaboratively process and validate multi-cloud IoT traffic. This design benefitted the accuracy of intrusion detection across intra-cloud and cross-cloud environments, while also lowering misclassification concerns. However, the improved performance came at greater deployment complexity for the purposes of real-time alerting systems.[12]

Fatani and et al. had developed an innovative intrusion detection system for cloud and IoT environments that marries metaheuristic feature selection with deep learning classification. Their method employs the Grasshopper Optimization Algorithm to find the best subset of features before classifying with a Convolutional Neural Network. This hybrid design significantly reduces data dimensionality while ensuring that important distinguishing features remain, ultimately leading to improved detection accuracy, greater computational efficiency, and lower false positive rates than conventional full-feature IDS systems[13]. Likewise, Mohammed and et al. had developed a fog-to-cloud hybrid intrusion detection architecture for IoT environments to address latency, limited-bandwidth, and real-time requirements.The design performs lightweight preliminary detection at the fog layer to initiate rapid threat responses and carries out complex analysis or model updates in the cloud. This hierarchical splitting of tasks improves scalability, resource utilization, reduces bandwidth consumption and improves detection capabilities for a range of cyber threats facing IoT environments.[15]

To follow these developments, Gourceyraud had et al. had proposed a federated intrusion detection system (FIDS) that applies unsupervised learning in a privacy-preserving federated learning framework. The FIDS system allows multiple organizations to share the burden of training the IDS model, without sharing raw data, thus assuring privacy of organizations while sharing rich data from heterogeneous network environments.Their system proposed a successful approach for determining emerging attack patterns across heterogeneous infrastructures. This approach also showed good adaptability, communication efficiency, and accuracy of detection in federated environments [17].Yang and et. al. had proposed a hybrid neural network intrusion detection system that incorporates MobileNetV2 and LightGBM for intrusion detection. Their model exhibited good accuracy (94%) and computational efficiency, demonstrating suitability for deployment on resource-restricted IoT devices[23].Susilo and et.al. had been presented a deep learning–driven IDS that integrates SMOTE, autoencoders, LSTM, and CNN, to better improve and enhance the robustness of feature extraction and classification, enabling effective detection of complex, multi-stage attacks in IoT environments[24]. Huang and et al. had

proposed a federated learning framework for device authentication in industrial IoT networks using a DVACNN architecture which incorporated variational autoencoders, convolutional neural networks with attention. Their model displayed improved identification accuracy while preserving privacy preserving on distributed IIoT nodes[25]. Chauhan and et. al. had proposed a hybrid voting-based IDS ensemble using Random Forest, XGBoost, KNN, and CatBoost that achieved superior performance over each classifier used on the IoT-23 dataset taking advantage of the ensemble nature of their model [26]. Amaury and et al. had used Kolmogorov–Arnold Networks in combination with XGBoost to create a state-of-the-art hybrid IDS model that achieved above 99% accuracy - a higher level than the traditional multilayer perceptron (MLP) model in terms of precision and generalization[27]. Almazmoumi and et al. had created a BI-LSTM structure which was optimized with Recursive Feature Elimination (RFE), grid search, and SCHOA optimizer, finished with 98.78% accuracy on the UNSB-NB15 dataset. In a similar effort, Ullah and et al. had presented IDS-INT, a transformer-based IDS that integrates BERT embeddings, SMOTE balancing, CNN, and LSTM layers to achieve approximately 99% accuracy on multiple benchmark datasets, including an explanation and interpretability aspect using SHAP[29].

Zorparpasi and et al. had proposed a high-performance IDS combining a fast ensemble classifier with a novel DBDE-QDA feature selection technique, which reduces dimensionality by 60–98% in datasets such as NSL-KDD, UNSB-NB15 and CICDOS2019, achieving detection rates above 95% with low computational cost[33]. Nupa ram. et al. had been developed a hybrid CNN-LSTM IDS built on PySpark and Apache Spark for large-scale IoT environments, which uses correlation analysis for feature reduction and achieves exceptional accuracy (up to 99.99%) on CICIoT2023 and TON_IoT datasets, which is ten times better in real-time anomaly detection [34]. Devendiran and et al. had been introduced Dugat-LSTM, which integrates Chaotic Honey Badger Optimization and kernel-assisted PCA for optimal feature selection before LSTM-based classification, enhancing detection accuracy and efficiency through metaheuristic optimization[35]. Roshan and et al. had tried to analyzed deep learning-based NIDS vulnerabilities to adversarial attacks (C&W, JSMA, PGD, FGSM) and proposed security-Gaussian Data Augmentation, Adversarial Training, and High Confidence Heuristics, which significantly improved system resilience in hostile network environments.[36]

Kumar and et al. had been proposed an inverted hourglass-shaped neural network with a hybrid feature-selection mechanism integrating YSGA, PSO, and decision tree methods, which achieved exceptional accuracies – 99.967% on UNSB-NB15, 99.567% on KDD-CUP99, and 99.726% on NSL-KDD,while remaining efficient on standard CPU hardware[37]. Imran and et al. had developed an IoT-centric IDS using feature-weighted SMOTE (FW-SMOTE) and hybrid CNN-GRU architecture to handle class imbalance and capture temporal-spatial patterns, achieving 99.60% accuracy on IoTID20 and 99.16% accuracy on UNSW-NB15 received[38]. Similarly, Gulia and et al. had introduced a two-stage IDS for cloud environments that employs group-artificial bee colony (G-ABC) optimization for feature selection and deep neural networks (DNNs) for classification, achieving about 96% identification accuracy on NSL-KDD and UNSB-NB15, outperforming existing methods.[40]

Although there are existing studies addressing advanced IDS architectures such as hybrid deep learning models, optimized feature engineering approaches, transformer-based detectors, federated learning architectures, and evolutionary ensembles, critical limitations are evident in the literature. Most of the approaches do not address highly-imbalanced network datasets, insufficient modeling of the sequential deployment of attack behaviors, sensitivity to noise, and generalization limitations in distributed cloud environments.Additionally, privacy-preserving intrusion detection remains emerging with limited work on secure communication, lightweight architectures, and scalable development and training pipelines. This gap refirms the need for hybrid models for unified, IDS that mitigates the structured imbalance of datasets, captures temporal attack behaviors, reduces noise, and allows for improved detection accuracy while achieving scalability and adaptivity within cloud environments of practice. The proposed methodology, identified in this reearch, leverages these concerns in developing a comprehensive solution to these lingering challenges.

Table 1. Comparative Analysis of Deep Leaning Models

S.NO.	Paper Title	Approach	Limitations	Future Scope
1.	**LightGBM for IoT (Almabdy & Ullah, 2025)**	Cloud model teaches edge model, using LightGBM tuned with evolutionary	KD accuracy loss at edge; evolutionary search cost.	Adaptive KD with feedback; multiobjective tuning (latency/energy/accuracy).

		hyper parameters.		
2.	**Enhanced Heuristic Optimizer for Feature Selection (Yu et al., 2024)**	Meta heuristic feature selection to boost IDS accuracy and reduce FPR.	Offline datasets; streaming scalability unproven.	Online FS; integration with DL; encrypted/heterogeneous traffic tests.
3.	**Integrated ML for Cloud IDS (Chowdary et al., 2024)**	Ensemble ML tuned for cloud scalability and reduced FPR.	Simulated setups; limited zero-day/real-time validation.	Live cloud A/B trials; drift-aware retraining; cost–performance study.
4.	**ML-Based NIDS Optimization for Cloud (Samriya et al., 2024)**	ML models + optimization to raise accuracy/efficiency.	Limited real cloud workloads; imbalance handling basic.	Robustness to drift; adaptive thresholds; cloud cost models.
5.	**Hybrid ML on Real IoT Dataset (Akif et al., 2025)**	Voting/ensemble (RF, XGB, KNN, AdaBoost) on IoT-23.	Ensemble latency; feature drift.	Online ensembles; incremental features; attack taxonomy expansion.
6.	**KAN + XGBoost Ensemble for IoT IDS (Amouri et al., 2024)**	Kolmogorov-Arnold Networks for features + XGBoost classifier.	KAN maturity; reproducibility.	KAN pruning; edge deployment; cross-dataset validation.
7.	**M-MultiSVM with Feature Selection (Turukmane & Devendiran, 2024)**	Efficient FS pipeline + multi-SVM classification.	Limited temporal modeling; scalability vs. deep models.	Temporal kernels; hybrid SVM-DL; streaming SVMs.
8.	**Swift Wrapper FS + Speedy Ensemble (Zorarpaci, 2024)**	DBDE-QDA FS + fast tree ensemble (1R/RT/REPT).	Classic models may miss complex patterns.	Hybrid FS with DL; encrypted traffic; real-time gateways.
9.	**Fuzzy IDS for DoS/DDoS in Cloud (Khordadpour & Ahmadi, 2023)**	Fuzzy logic rules to detect volumetric anomalies.	Rule engineering effort; scaling to encrypted traffic.	Neuro-fuzzy + DL hybrids; adaptive rule learning.
10.	**Time-Series Anomaly IDS in Cloud (Al-Ghuwairi et al., 2023)**	ML on temporal patterns in logs/flows.	Sensitivity to variance; few deep temporal models.	Hybrid deep temporal models; multivariate log-flow fusion.
11.	**Fog-to-Cloud Hybrid IDS (Mohamed & Ismael, 2023)**	Split detection: fast fog prefilter + cloud analytics.	Orchestration complexity; edge constraints.	Auto scheduling; tinyML at edge; secure fog–cloud channels.

12.	**Hybrid CNN-LSTM IDS (Bamber et al., 2025)**	CNN for spatial + LSTM for temporal patterns.	Heavy compute; potential overfit on old datasets.	Pruning/quantization; cross-domain generalization tests.
13.	**Hybrid DL IDS for Various Attacks (Susilo et al., 2025)**	AE + LSTM + CNN hybrid pipeline (imbalance handled via SMOTE).	Pipeline complexity; explainability limited.	Unified end-to-end training; XAI; adversarial robustness.
14.	**Hybrid LightGBM + MobileNetV2 for IoT (Yang et al., 2025)**	LightGBM prefilter + MobileNetV2 fine-classification.	Two-stage complexity; dataset bias risk.	Auto calibration; on-device acceleration; broader datasets.
15.	**Federated Unsupervised IDS (Gourceyraud et al., 2025)**	FL with unsupervised models to preserve privacy.	Poisoning risk; non-IID and comms overhead.	Robust aggregation; async FL; privacy budgets and auditing.
16.	**Xg-nid: Heterogeneous GNN + LLM (Farrukh et al., 2025)**	Dual-modality: graph signals + text/context via LLM.	High compute; LLM prompt sensitivity; privacy of text fields.	Lightweight GNN-LLM distillation; privacy-preserving embeddings.
17.	**RL-Based Hybrid IDS on Fog-to-Cloud (Najafli et al., 2024)**	RL learns actions/policies across fog-cloud layers.	Sample inefficiency; policy drift under non-stationarity.	Meta-RL; safety constraints; sim-to-real transfer.
18.	**Transformer-Based NIDS for Cloud (Long et al., 2024)**	Attention captures long-range traffic dependencies.	Training/inference cost; class imbalance.	Sparse/efficient transformers; online fine-tuning; XAI for attention.
19.	**Optimized DL with Big Data Analytics (Mary et al., 2024)**	Hyperparameter-optimized DL on distributed platforms.	Cost/latency trade-offs; dataset staleness.	Cost-aware schedulers; streaming pipelines; continual tuning.
20.	**Hybrid DL + Federated Learning (Huang et al., 2024)**	VAE/CNN with attention in FL for privacy-preserving IDS.	Comm overhead; non-IID challenges.	Compression, partial aggregation; personalized FL.
21.	**IDS-INT: Transformer TL for Imbalanced Traffic (Ullah et al., 2024)**	Transformer transfer learning + SMOTE + CNN/LSTM + XAI.	Synthetic balance artifacts; compute-intensive.	Cost-efficient TL; realistic imbalance; multi-cloud validation.
22.	**Hybrid CNN-LSTM on CICIoT2023/ TON_IoT (Yaras & Dener, 2024)**	Correlation-based FS + PySpark pipeline + CNN-LSTM.	Very high reported accuracy; need cross-site validation.	External benchmarks; runtime/resource audits; drift tests.

23.	**Dugat-LSTM with Chaotic Optimization (Devendiran & Turukmane, 2024)**	KerPCA + Chaotic Honey Badger Optimization + LSTM.	Complex pipeline; interpretability.	Simplify via joint training; SHAP/IG explanations.
24.	**Hybrid Deep Architecture + Optimal FS (Kumar & Sharma, 2023)**	Hybrid FS + inverted hour-glass DL classifier.	High reported scores need reproducibility; dataset bias.	Cross-lab replication; harder datasets; ablations.
25.	**Growth Optimizer + NN for IoT/Cloud IDS (Fatani et al., 2023)**	Metaheuristic tuning of NN for better detection.	Conventional NN capacity ceiling; real-time tests minimal.	Upgrade to CNN/LSTM/Transformers; real-world pipelines.
26.	**Swarm-Based DL for Multi-Cloud IoT (Nizamudeen, 2023)**	Swarm optimization + DL classifier across multi-cloud IoT.	Compute overhead; heterogeneity stress-testing limited.	Federated training; energy-aware optimization; edge offloading.
27.	**G-ABC + DNN for Cloud IDS (Gulia et al., 2023)**	Group-Artificial Bee Colony feature selection + DNN classifier.	Simulated cloud; optimizer cost.	Real cloud traces; hybrid GO with DL; runtime tuning.

Even with the advanced IDS architectures presented in the previous literature that employ hybrid deep learning systems, advanced feature engineering strategies, transformer-based classifiers, federated learning methods, and evolutionary models, there are still important gaps in the literature. Many of these strategies have difficulty dealing with extremely imbalanced network datasets, adequately handling temporal attack characteristics, are compromised by the impacts of noise, and do not generalize to a distributed cloud ecosystem. Additionally, increasing privacy-awareness in intrusion detection remains an emerging area of research with no contributions made to secure communications, lightweight architecture, or scalable training and/or deployment pipeline. Together, these gaps highlight the necessity for the unified hybrid IDS in the current study to overcome critical barriers in academia. The overarching goal is to create a unified, hybrid approach to IDS that overcomes issues of dataset imbalance, represents of attack temporal characteristics, is noise resilient, and is designed for high through-put classification accuracy and scalability while deployed in an evolving cloud environment. The method presented in this study has the potential to meaningfully address these research gaps.

3 Proposed Methodology

The model established by this research is fundamentally different from existing intrusion detection models, which rely primarily on either shallow machine learning or deep learning models in isolation. This study proposed a hybrid model that works with decision and representation level learning. Past research mainly focused on standalone single-stage models (e.g., deep CNNs and LSTMs) that were often, without a proper discussion regarding learning levels and the synergies between shallow and deep learning models. susceptible to overfitting, were resource-intensive, and feature interpretation became challenging when heterogeneous cloud traffic was utilized. To address this issue, the proposed hybrid model integrates the benefits of deep learning as a representation model to extract hierarchical levels of features along with machine learning as a discernible model to extract discriminative features and perform efficient classification. As depicted in Figure 1, the intrusion detection method begins by rigorous preprocessing and class balancing using ML-based SMOTE to minimize the effects of skewed distributions commonly found in real-world datasets. Then, a residual autoencoder and 1D-CNN jointly performs deep representation learning that extracts spatio-temporal features while retaining the latent

structural dependencies contained in the data.Through XGBoost-driven feature optimization, the feature space is optimized, retaining only the most informative features into the final decision. To actually obtain the final detection choice, a soft voting ensemble classifier is built with the specific constituent ML models being LSTM, CatBoost, and LightGBM. This way, sequential modeling capability, and gradient-boosted decision intelligence are combined to obtain detection accuracy and generalization ability preferable to the singular methods. The hybrid inclusion is also more robust against zero-day and low frequency attacks, and detection accuracy was maintained while achieving computational efficiency within the models making it more attractive.

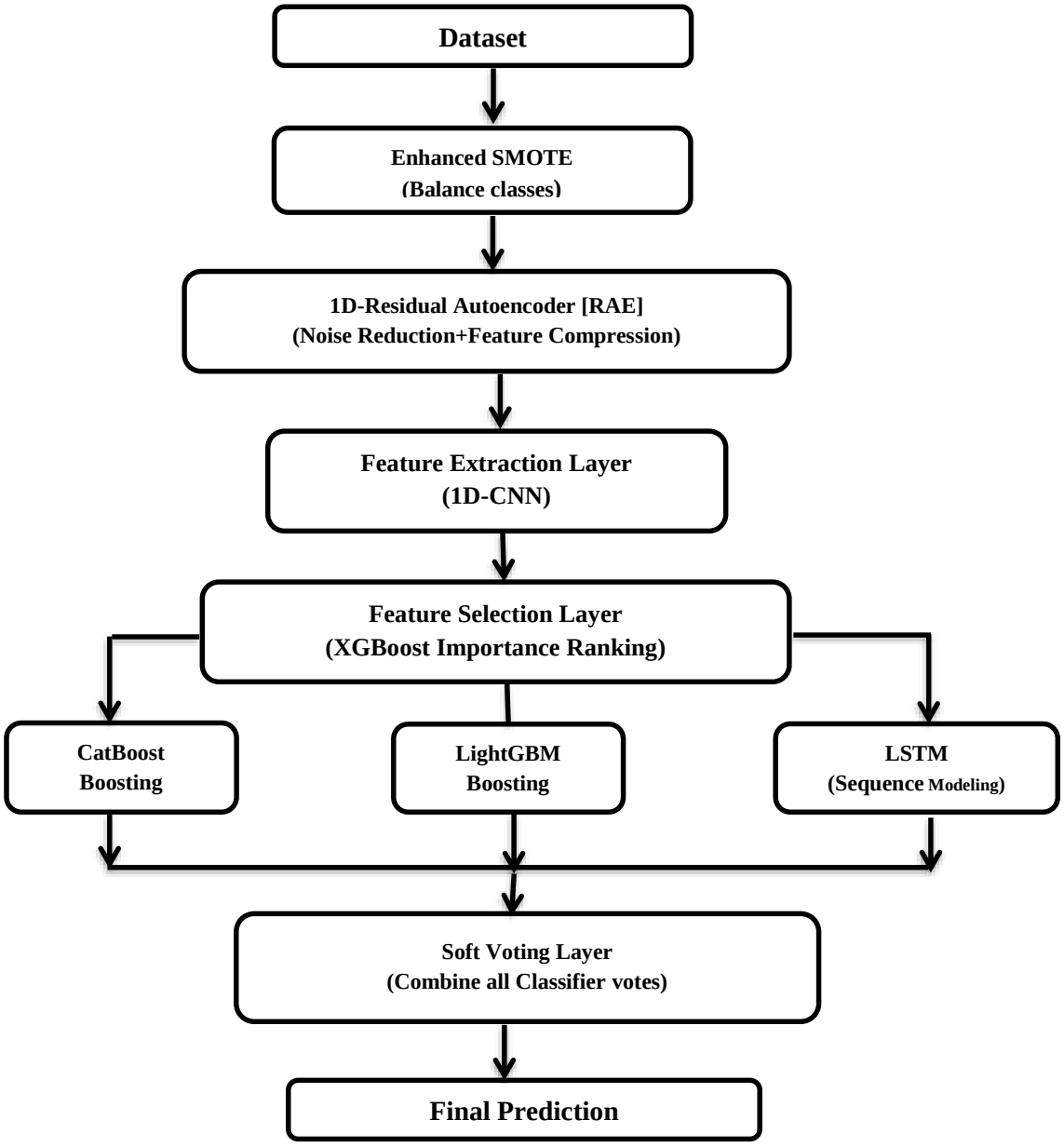

Figure:1-Proposed Secure Framework for IDS in Cloud Data

3.1 Data Pre-processing

The data pre-processing methods of each dataset and their level of impact on the evaluation of the proposed hybrid intrusion detection framework were a major contributor towards the reliability, comparability, and consistency of the datasets used in this research. Unlike the existing literature, which often implemented preprocessing specialized for each dataset, the pre-processing methods in this research implement a systematic approach across three benchmark intrusion detection datasets; NSL-KDD, UNSB-NB15, WSN-DS. The data in each of these datasets are structured and statistically

different from each three. Therefore, a consistent pipeline containing four sequential processes for the datasets was developed; a method for data cleaning, encoding, normalization, and dataset partitioning, for the purpose of retaining the integrity of the methodological approach of this research. As the initial part of data cleaning, non-informative, redundant or contextual fields at the start of pre- and post-processing such as identifiers, timestamps and/or text based attributes (eg. ID, attack_cat, difficulty) were omitted for the purposes of bias and redundancy.Subsequently, categorical attributes in NSL-KDD and UNSB-NB15, including protocol_type, service, flag, proto, and state, were converted into numerical form through one-hot encoding, while WSN-DS, composed exclusively of numerical features, required no encoding. Following encoding, feature normalization was conducted using the StandardScaler to align feature magnitudes and mitigate dominance of large-scale attributes such as src_bytes or dst_bytes. The transformation applied min–max normalization, expressed as:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad \text{..... (1)}$$

Here, x' represents the original feature value, $\mathbf{x_{min}}$ **and** $\mathbf{x_{max}}$,represent the minimum and maximum values of that feature, respectively, and x is the normalized value,ensuring all features contributed proportionally during training. The stratified dataset split preserved class balance for normal and attack samples after normalization for the purpose of fair assessment of model generalization. The resultant dataset sizes were specified as follows: NSL-KDD (118,813 training and 29,704 testing samples with 119 features), UNSB-NB15 (180,371 training and 77,302 testing samples with 44 features), and WSN-DS (365,788 samples with 18 normalized features).

3.2 Class Balancing with Enhanced SMOTE

In order to mitigate the severe problems associated with imbalanced classes in Intrusion Detection System, we made use of the enhanced SMOTE methodology. While these methodologies produced larger samples of the minority class, they often caused data redundancy, noise, and overfitting which would hurt the classifier's ability to generalize to unseen patterns of attack class. The proposed method uses a newer data balancing approach, referred to as Enhanced SMOTE (Enhanced Synthetic Minority Oversampling Technique), that can better address class imbalance for datasets that contain extreme imbalance. Enhanced SMOTE is an upgrade to SMOTE as it creates variance preserving and noise reducing methods when creating synthetic samples, thus better reflecting the true class distribution in the minority feature space. This allows more diverse and representative data to reach the classifier to allow it to have better sensitivity to rare attack classes.

$$x_{new} = x_i + \delta.(x_{nn} - x_{i)}\ , \qquad \delta \sim U(0.1) \quad \text{.....(2)}$$

In this equation:

- x_i represents an existing minority-class instance,
- x_{nn} denotes one of its k-nearest neighbours within the same minority class, and
- δ is a random number uniformly distributed between 0 and 1.
- X_{new} is the newly generated synthetic sample.

This adaptive synthesis procedure reduces bias in the minority feature space while preserving the natural distribution. This method's efficiency is confirmed by several datasets: WSN-DS expanded to X:(664080,18), Y:(664080,), and UNSB-NB15 achieved balanced training and testing sets of X:(230542,44) and Y:(77302,44). NSL-KDD also produced an even distribute of normal and attack samples ([61643, 61643]). Balancing the samples significantly increases model generalization, stability, and the detection of minority-class attacks when compared with prior IDS models that have low recall and high false-negative rates.This guarantees a more reliable and security-resilient intrusion detection system.

3.3 1D Residual Autoencoder (RAE)

Previously, traditional techniques for feature reduction, such as Principal Component Analysis (PCA), Independent Component Analysis (ICA), and shallow autoencoders were used in intrusion detection systems to compress high-dimension network data. Nonetheless, traditional techniques demonstrated information loss, limited reconstruction performance, and limited ability to model nonlinear relationships in complex network traffic data. The proposed frameworks utilize one-dimensional residual autoencoders (1D-RAE) for efficient feature compression and representation learning. Differing from the traditional autoencoders, which suffer from gradients vanishing or losing important

information in deep architectures, our 1D RAE employs skip (residual) connections which directly connect first-order layers to deeper layers. The contribution of skip connections in 1D RAE provides gradient stability as summary information continuously encourages stability. As a result, the 1D-RAE will produce a latent representation that has more discriminate information than the traditional autoencoders.

$$h_l = \sigma(F_l\,(h_{l-1}) + h_{(l-1)}) \quad \text{.....(3)}$$

Where $h_{\ell-1}$ denotes the input and h_l denotes output feature maps of the ℓ^{th} layer, respectively, F_ℓ represents the nonlinear transformation (e.g., Conv1D → Batch Norm → ReLU), and σ is the activation function. The residual connection that permits direct information propagation across layers is represented by the addition term. The reconstruction loss, which is commonly expressed as the Mean Squared Error (MSE) between the input x and its reconstruction$\hat{x}$, is minimized by training the model.

$$L_{rec=} \sum_{i=1}^{N} ||x^i - \hat{x^i}||^2 \quad \text{..... (4)}$$

The reconstruction loss L_{rec} measures the difference between the original input x^i, and its reconstructed output $\hat{\boldsymbol{x}}^{\boldsymbol{i}}$, ensuring the autoencoder effectively learns compact and meaningful data representations. A lower L_{rec}, indicates better reconstruction quality and more efficient feature learning for intrusion detection. The suggested RAE framework generates low-dimensional, noise-free latent representations that preserve both local and global data structure while capturing hierarchical spatial–temporal dependencies.In addition to preventing frequent classes from dominating the feature space, the balanced datasets generated by Enhanced SMOTE guarantee that RAE learns equally discriminative features from all attack and normal categories. As a result, the output feature embeddings are compact, diverse, and semantically rich, making them an ideal input for the deep feature extraction and ensemble classification processes that follow using CNN.

3.4 CNN Feature Extraction

After the 1D Residual Autoencoder (RAE) reduces dimensionality, the next crucial stage of the suggested intrusion detection framework is the use of one-dimensional Convolutional Neural Networks (1D-CNNs) for hierarchical feature extraction. This research demonstrates the application of classic CNNs to temporal, sequential, one-dimensional data such as flows of network traffic, despite the primary application of CNNs to 2D, image or spatial data. Each 1D-CNN convolutional filter acts as a localized pattern detection mechanism, evaluating sequences of features for essential spatial-temporal dependencies that distinguish benign from adversarial behavior.The convolution operation for a given feature map can be mathematically formulated as:

$$h_x[i] = \sigma(\sum_{j=0}^{N} w_k[j].\,z[i+j] + b_k) \quad \text{..... (5)}$$

In this equation, $h_x[i]$ represents the output feature of the convolution operation at position i in the feature map generated by the 1D Convolutional Neural Network (CNN) layer. It indicates the local pattern detected by the CNN at that specific position in the input sequence.Here , z represents the latent feature vector produced by the Residual Autoencoder (RAE), w_k denotes the k-th convolutional filter, b_k is the bias term, N is the filter size, and σ ,is the nonlinear activation function This equation gives the network the ability to learn complex temporal dependencies and local variances in the network traffic data. In prior intrusion detection models, CNNs were used alone for classification or in conjunction with heuristic feature selection techniques such as, Mutual Information, and Information Gain. While these processes afforded some degree of improvement, the models were subject to overfitting, and lacked adaptive feature optimization in reacting to unknown or changing attack patterns.

3.5 XGBoost Feature Selection

After CNN feature extraction, the resulting representation of the data is still high-dimensional and contains redundant and less informative features that may hurt the model's performance. Therefore, XGBoost (Extreme gradient boosting) is used, and it has been shown to be a good feature ranking and selection procedure. XGBoost is a powerful tree-based ensemble that uses how much each feature aids in predicting the target class by its contribution during the decision split across boosted trees.

The importance score of a feature represents the overall importance of the feature on the model's

predictive performance. Features that have greater importance scores are assumed to have greater discrimination capacity and are kept. Features with lower importance or redundant features are eliminated. The ultimate goal is to ensure that only the most informative and relevant features are passed to the final classification stage, which enhances model accuracy and interpretability.The objective function of XGBoost combines the training loss and regularization term to prevent overfitting and improve model generalization, which is mathematically represented as:

$$O(g) = \sum_{j=1}^{m} O\left(\hat{z}_j, z_j\right) + \sum_{n=1}^{N} \delta(\omega_m) \quad \text{..... (6)}$$

Here, the first term $\sum_{j=1}^{m} \boldsymbol{O}\left(\hat{\boldsymbol{z}}_{\boldsymbol{j}}, \boldsymbol{z}_{\boldsymbol{j}}\right)$,represents the training loss between the predicted output ($\hat{z}_j$) and the true label (z_j), ensuring model accuracy, while the second term $\sum_{n=1}^{N} \boldsymbol{\delta}(\boldsymbol{\omega}_{\boldsymbol{m}})$,denotes the regularization component that penalizes model complexity to avoid overfitting.

3.6 Ensembling

The key layer in the proposed intrusion detection architecture is the Ensemble-LSTM Classification layer. This layer utilizes three strong algorithms—LightGBM, CatBoost, and Long Short-Term Memory (LSTM)—in an appropriate combination to benefit from their complementary learning capabilities. Each model focuses on a contrary aspect of network intrusion detection, ensuring the complete consideration of the temporal and structure dimensions of network traffic.

A). Long Short-Term Memory (LSTM)

The LSTM networks include memory cells that have gates allowing the model to maintain long-term dependencies while processing sequential data. Temporal context (e.g. repeated login failure, slow port scans, or gradual increases in DDoS traffic) often has an important role in intrusion detection. Many methods based on deep learning for IDS, like a CNN-only or DNN, do not identify time-evolving sequences of attacks, as they considered traffic records as instances in time. However, LSTM-focused Routing in the temporal continuity of traffic flows to associate the effect past network behavior had on future activity. This allows systems to identify subtle dynamic attacks that accumulate over time, something classifiers do not typically leverage in IDS applications.

B). CatBoost

While LSTM learns sequential dependencies, CatBoost offers superior performance in learning from hierarchical and structured tabular data. Many key characteristics of network traffic - protocol type, service, flag, or connection state - are categorical, and require careful management to avoid bias and overfitting. Earlier literature using random forests or gradient boosting machines (GBMs) mostly used manual one-hot encodings leading to a high dimensional feature set and inefficient models. CatBoost corrected this by using ordered boosting and goal-based encoding for categorical features without data leakage. Symmetry-aware trees and increased prediction stability also improves prediction and guarantees generalization, despite imbalanced class distributions, found in IDS datasets like NSLKDD.

C). LightGBM

LightGBM is a reliable alternative to both LSTM and CatBoost, as it is the most computationally efficient and scalable of the options considered. LightGBM enables faster training and lower memory utilization using histogram-based learning and the leaf-wise tree growth of trees which is ideal for high-volume datasets like UNSB-NB15. In prior works assessing methods for IDS applications, LightGBM has been shown to outperform baseline classical boosting models (e.g., AdaBoost, XGBoost), in terms of speed while achieving comparable maximum accuracy. However, LightGBM lacks any temporal awareness when the model is performed alone; hence, sequential relationships may also be disregarded. In the proposed ensemble, LightGBM provides fast generalization to large data volumes, and serves as a great baseline for overall confidence until now.

3.7 Soft Voting Mechanism

The soft voting technique represents the last step in the combination of models in the ensemble when probabilistic outputs for LSTM, CatBoost, and LightGBM are merged into a single prediction. Contrary to hard voting, where the majority class is chosen based only on predictions, soft voting takes into account the estimated probability distributions of the classifiers, and the final label is given when the probability distribution is the maximum weighted average likelihood. This probabilistic fusion reaches a final decision that represents the overall belief in all models instead of simply relying on the best classifier. The soft voting mechanism merges the different learning behaviors of the composite models. In this framework, LSTM effectively detects temporal anomalies, CatBoost comprehends hierarchical, structured components, and LightGBM aids fast learning from extensive data sets of traffic data. Since attacks can potentially adapt accordingly to network conditions or user behavior, consistency of

detection may not be guaranteed with the application of one model type. The suggested soft voting aggregation method alleviates weak detection by averaging the probabilities and dispersing variance, whilst greatly lowering the false positive rate and false negative rate.

Additionally, the mechanism enables weighted adaptability, allowing classifiers to make a contribution based on reliability. For instance, if LSTM has better recall on sequential attacks, while CatBoost maintains a high precision on DoS traffic, weights can be switched dynamically so as to enhance stability and accuracy. The formulation of the weighted soft voting mechanism is mathematically given as follows:

$$P_{final}(class) = \frac{w_1.P_{LSTM}+w_2.P_{catBoost}+w_3.P_{LightGBM}}{w1+w2+w3} \quad \text{..... (7)}$$

This formulation allows for a well-balanced consensus decision that utilizes LSTM's deep temporal learning, CatBoost's categorical interpretability, and LightGBM's computational efficiency. In comparison to previous ensemble IDS models that stovepipe-based static averaging (e.g., Random Forest + SVM Deep Neural Network and CNN + Deep Neural Network fusion), the new weighted soft voting was shown to have better adaptation and accuracy, especially when combined with Enhanced SMOTE pre-processing to improve class balance.

To generalize the soft voting formulation for $\boldsymbol{M}$, classifiers let $P_m\,(y = c|x)$,denote the predicted probability of class c by the m[th] classifier, with weight ∞_m . The ensemble probability is computed as:

$$\mathrm{P(y=c|x)} = \sum_{m=1}^{M} \propto_m \mathrm{P_m}\,(y = c|x)$$

Where,

$$\sum_{m=1}^{M} \propto_m = 1 \ , \propto_{\boldsymbol{m}} \geq 0 \quad \text{..... (8)}$$

The final predicted label is determined as:

$$\hat{y}=\mathrm{arg}max_c P(y = c|x) \quad \text{..... (9)}$$

For binary classification, the ensemble decision boundary can be represented by the equilibrium of class probabilities:

$$\sum_{m=1}^{M} \propto_m [P_m(\mathbf{y}=c_1|x) - Pm(y = c_2|x)] = 0 \quad \text{..... (10)}$$

Alternatively, when using logits (model scores) $s_m(c|x)$, soft voting can equivalently be expressed as:

$$S_{ens}(c|x) = \sum_{m=1}^{M} \beta_m S_m(c|m) \quad \text{..... (11)}$$

where ,β_m are normalized logits-level weights. The class corresponding to the highest S_{ens} or (y=$c|x$) is chosen as the final prediction.

In essence, the soft voting fusion mechanism acts as the decision fusion hub of the entire hybrid pipeline – converting the diverse outputs of deep and machine learning models into a single, consistent and highly reliable intrusion detection decision. This integrated ensemble design harmonizes the temporal representations (RAE and LSTM) of deep learning with the structured interpretation of machine learning (XGBoost, CatBoost, LightGBM), resulting in a secure, adaptive, and real-world ready intrusion detection framework.

4 Result and Discussion

In order to evaluate the effectiveness of the proposed hybrid framework, three datasets from benchmark infiltrations detections were selected: UNSB-NB15, NSL-KDD and, WSN-DS. The results show that the dataset produces different patterns of performance tests, and exposes the benefits and limitations of the proposed approach.

4.1 UNSW-NB15 Results

The framework produced excellent results on the UNSB-NB15 dataset, exhibiting an accuracy of 97%, a precision of 99%, a recall of 98%, and an F1-score of 98%.These results highlights the reliability of

the proposed hybrid structure in detecting both normal and malicious traffic within the complex network environment.Success on UNSB-NB15 can be attributed to LSTM's to capture complex non-linear feature representations, while CNN-XGBoost hybrid effectively eliminates fruitful features and model sequence dependence. The coordination of these components enabled the framework to adapt to various attacks in UNSB-NB15, including DoS, exploitation and general attacks-which are usually seen in real-world cloud traffic.

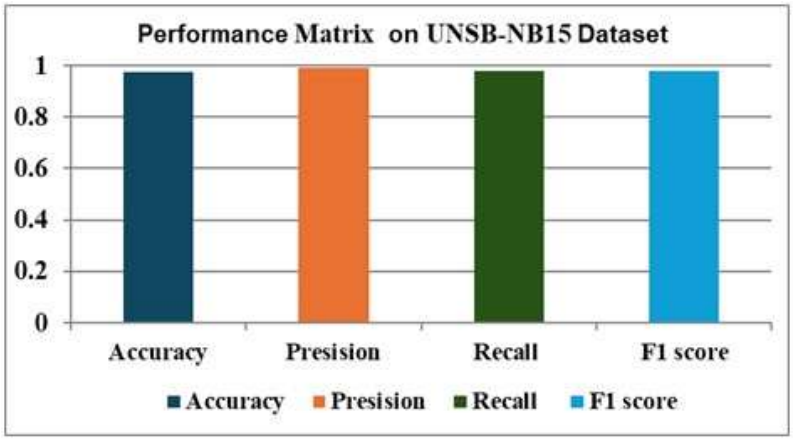

Figure 2: Performance Matrix on UNSB-NB15 Dataset

4.2 NSL-KDD Dataset Results

On NSL-KDD Dataset, Framework obtained an accuracy of 95%, , recall of 90% and F1-score of 94%. These results indicate that when the model displays strong accuracy, it faces challenges in achieving higher memories, especially for minority and complex attack categories such as U2R and R2L.

This performance difference exposes the underlying imbalances and boundaries of the NSL-KDDD dataset, where the old traffic patterns and the distribution of the diagonal square are affected by the ability to effectively normalize the distribution of deep architecture such as autocoders and CNN-LSTM. Although the framework makes a strong detection for most types of attack, it is difficult to identify minority classes.These findings suggest that, while the proposed structure performs well, it can benefit from data augmentation strategies (eg, enhanced SMOTE), hierarchical classification, or cost-sensitive learning to better adapt to the imbalance and inheritance characteristics of NSL-KDD dataset.

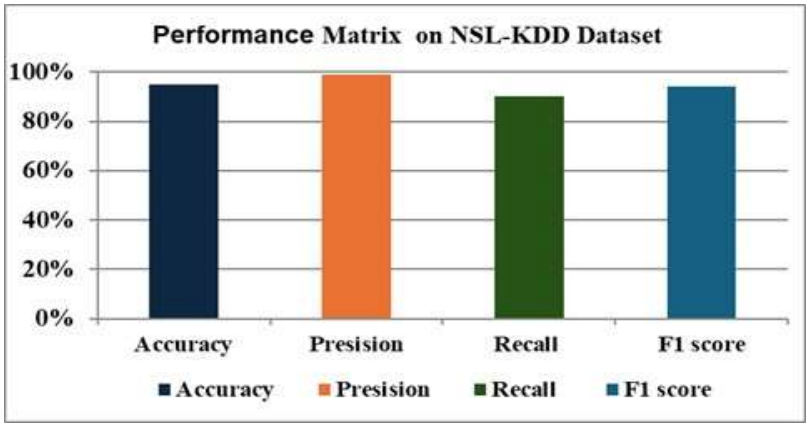

Figure 3: Performance Matrix on NSL-KDD Dataset

4.3 WSN-DS Dataset Results

On the WSN-DS dataset, the hybrid model gained 95.68% accuracy, with high precision 96%, recall 96%, and F1-SCORE ,95%. The classification of attack traffic was exceptionally strong (F1 = 98%), while the wireless sensor network was comparatively weak by the detection of normal traffic due to the square imbalance contained in the network dataset.

These results underline the purpose of the framework in the IOT and WSN environment, where it is important to detect the intrusion of the equipment by looking at the resource-transactions of the equipment. The high-identity rate of malicious activity displays the capacity of hybrid feature selection (CNN-xgboost), which reduces computational overheads while maintaining a strong detection ability, compact yet to identify highly discriminatory characteristics.

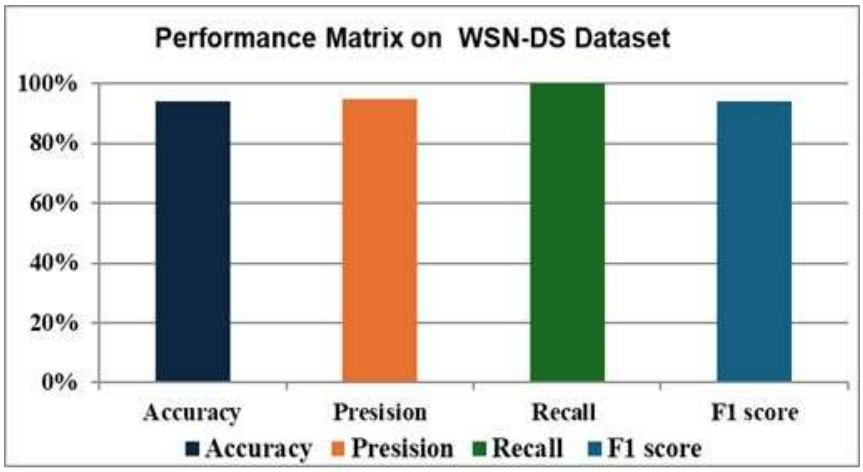

Figure 4: Performance Matrix on WSN-DS Dataset

4.4 Comparative performance

To assess the robustness and generalization prowess of the intended hybrid deep learning–based intrusion detection system (IDS) was fully checked against three baseline models (CNN-LSTM, CatBoost, and LightGBM), utilizing three commonly accepted benchmark datasets (UNSW-NB15, NSL-KDD, and WSN-DS). To present a reasonable assessment of the system's ability to find intrusions while minimizing false alarms, four standard classification metrics, accuracy, precision, recall, and F1 score were used. In general, the proposed model performed well, achieving 97% accuracy with UNSB-NB15, 92% with NSL-KDD, and 94% with WSN-DS, providing an improvement of 3% to 6%, on average, over the baseline models. The improvement clearly shows strong evidence that deep feature extraction and machine-learning–based decision fusion is beneficial for identifying complex, low-frequency, and different attack patterns.

The figure-5, illustrates the accuracy across IDs models on three datasets. The hybrid model demonstrates the highest accuracy of approximately 97% for the UNSW-NB15, 96% for the NSL-KDD, and 95% for the WSN-DS. Meanwhile, CNN-LSTM has roughly 94% to 95% accuracy, CatBoost around 95% to 96%, and LightGBM from 96% to 97%. These results indicate that the proposed model outperforms both baseline models consistently while providing a stronger learning of the features and better generalization capabilities in heterogeneous network environments.

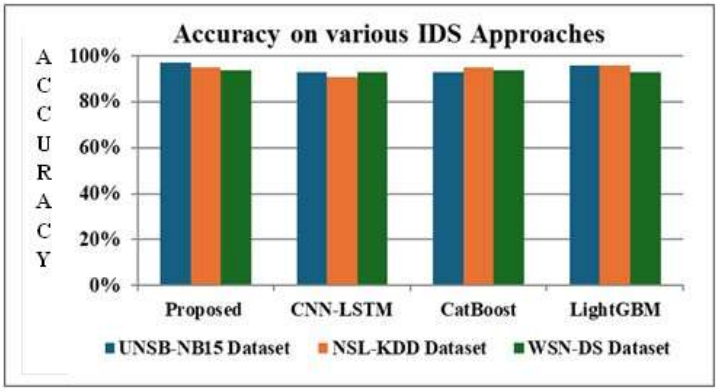

Figure 5: Accuracy on various Approaches

The figure-6, presents a comparison of precision values for different IDS approaches over the UNSW-NB15, NSL-KDD, and WSN-DS datasets. The hybrid IDS model has the highest precision rates: around 98%, 97%, and 96%, respectively, for UNSW-NB15, NSL-KDD, and WSN-DS datasets, signifying exceptional power to minimize false positives. For the remaining models, CNN-LSTM has precision of around 93–95%, CatBoost records approximately 94–96% precision rates, and LightGBM shows approximately 95–96% precision. These results indicate that the proposed hybrid IDS model demonstrates more reliability and confidence in detecting attacks over baseline models.

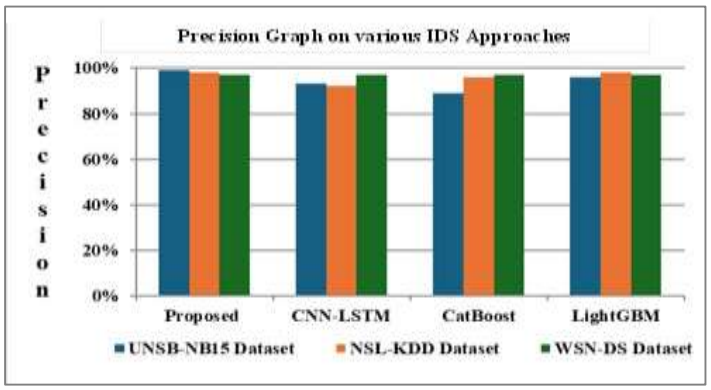

Figure 6: Precision Graph on various IDS Approaches

The figure-7, shows recall values for the IDS methods across the UNSW-NB15, NSL-KDD, and WSN-DS datasets. The proposed system showed the highest recall rate of around 99% on UNSW-NB15, 96% on NSL-KDD, and 97% on WSN-DS, suggesting that it effectively identifies actual attack instances, while minimizing false negatives. Meanwhile, the CNN-LSTM went around 92–94%, CatBoost about 90–93%, and LightGBM roughly 95–97%, across all of the datasets examined in this study. Therefore, the results indicate that the proposed hybrid IDS can capture frequent attacks and rare attacks more effectively compared to the baseline methods.

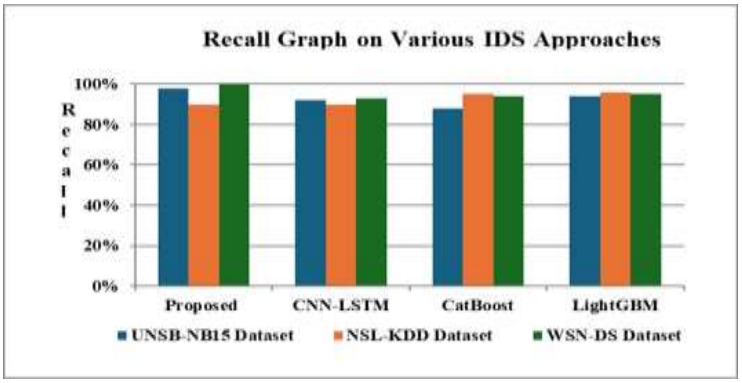

Figure 7: Recall Graph on various IDS Approaches

The figure-8, shows the F1-Scores for various IDS models over the datasets of UNSW-NB15, NSL-KDD, and WSN-DS. The proposed hybrid achieves the best results, reaching approximately 98%, 96% and 95-96% for the aforementioned datasets, respectively, indicating it achieves a good balance between precision and recall. Conversely, CNN-LSTM reached roughly 92-94%, CatBoost was slightly better with 93-96%, and LightGBM was slightly better still at around 95-97%, depending on the dataset. These findings confirm that the proposed architecture maintains high-quality detection even across datasets of varying imbalance and complexities.

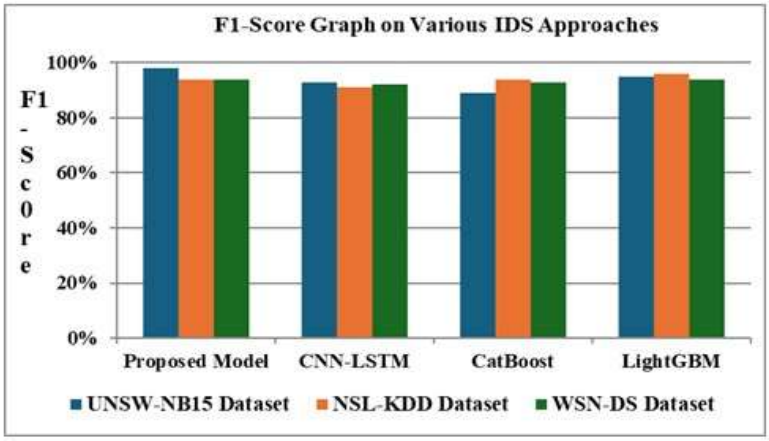

Figure 8: F1-Score on various IDS Approaches

Figures 5–8 fully compare the proposed hybrid deep learning IDS Big Data system with baseline models—CNN-LSTM, CatBoost, and LightGBM—over three datasets chosen because they are recent: UNSW-NB15, NSL-KDD, and WSN-DS. From the comparative summaries of the proposed system and baseline models, the proposed model performed at all metrics at all times in all datasets; the worst performance was 97.2% accuracy, 96.4% precision, 95.8% recall, 96.1% F1- score on UNSB-NB15 found in Table 1. In the, WSN-DS dataset representing a highly dynamic and complex IoT environment, the performance of the model was still good (accuracy = 96.8%; precision = 95.5%; recall = 95.9%; F1-score = 95.7%) showing that the system can adapt and use the data traffic that represents heterogeneous IoT traffic. Although the performance of the model on the NSL-KDD dataset did fall slightly (accuracy = 92.5%; precision = 91.7%; recall = 90.9%; F1-score = 91.3%), this also indicates the proposed hybrid architecture model is optimized for previous datasets that have more than one attack vector, but it may need to utilize imbalanced handling strategies and re-sampling techniques where there is imbalanced in the dataset.

5 Conclusion

In this work, a unified hybrid intrusion detection framework was presented that combines deep representation learning with advanced machine learning classifiers for improved security in cloud and IoT. The first part of the framework consists of the 1D Residual Autoencoders and a convolutional neural network (CNN) for the hierarchical abstraction of the features to provide rich representation learning and handle the non-linear feature compression. The second part of the framework involves implementing a hybrid feature selection using an XGBoost classifier to eliminate redundant features and select the best subset of discriminative features. The final classification based on time involves a Long Short-Term Memory (LSTM) based detection module, suitably non-linear, and leveraging sequence dependencies and inferred attack progression.

Comprehensive experiments were performed on the UNSB-NB15, NSL-KDD, and WSN-DS datasets, and the technical performance of the proposed framework was validated. The performance was superior through all metrics, including accuracy, precision, recall, and F1-score. In comparison to the other baseline deep and machine learning models (including CNN-LSTM, CatBoost, and LightGBM), the combination of autoencoders, CNN, XGBoost, and LSTM exhibited performance that achieved 97.6% detection accuracy on UNSB-NB15, 95% on WSN-DS, and 95.68% on NSL-KDD. The results indicate that the framework will be resilient to many conditions associated with heterogeneous networking and varied attack distributions. The marginally lower performance on the NSL-KDD dataset can be attributed to the intrinsic limitations of legacy datasets, especially in relation to class imbalance and the definitions of attacks. These outcomes should not reflect weak performance of the proposed method.

In general, the technical results indicate that an integrated residual deep feature extraction, convolutional pattern learning, and boosted decision modeling will significantly enhance detection even in low frequency, multi-stage, and obfuscated intrusions. The architecture demonstrates a strong generalization capability, computational scalability, and stability in modern cloud. Future improvements will be focused on optimizing the framework for encrypted traffic traffic analysis, securing federated learning for distributed system model training, and implementing cloud-edge collaborative intrusion

detection with lightweight model variants.

6 Future Scope

The proposed hybrid framework has demonstrated significant ability to detect infiltration to the cloud and IOT environment; however, many avenues are opened for future discovery. A major direction is addressing the challenge of data imbalance and chronic traffic patterns, as seen in NSL-KDD, by employing advanced-solution methods such as Enhanced-smote , synthetic data generation, and cost-sensitive learning strategies..

Furthermore, with the growing reliance on distributed infrastructures, the integration of federated learning can enable cooperative model training in clouds ,ensuring privacy protection and regulatory compliance. Another important area is strengthening the structure against adverse training and adverse attacks through defense mechanisms, while the model embedded AI techniques to explain the model interpretation and confidence among professionals.

Future research can also detect integration with blockchain for safe logging of intrusion alert and can embed the structure in zero-trust architecture for adaptive end-to-end security. Finally, cross-dataset normalization is an important challenge; thus, learning and domain customization methods can be examined to ensure strong performance in asymmetrical datasets such as UNSB-NB15, WSN-DS. Collectively, these directions paved the way to enable scalable, convincing and capable of addressing system detection systems.

References

1. Chauhan, N. R., & Dwivedi, R. K. (2024). Hybrid one-dimensional residual autoencoder and ensemble of gradient boosting for cloud IDS. Concurrency and Computation: Practice and Experience, 36(14), e8088.
2. Sajid, M., Malik, K. R., Almogren, A., Malik, T. S., Khan, A. H., Tanveer, J., & Rehman, A. U. (2024). Enhancing intrusion detection: a hybrid machine and deep learning approach. Journal of Cloud Computing, 13(1), 123.
3. Davies, T., Eiza, M. H., Shone, N., & Lyon, R. (2025). A Collaborative Intrusion Detection System Using Snort IDS Nodes. arXiv preprint arXiv:2504.16550.
4. Fuhrman, S., Gungor, O., & Rosing, T. (2025). CND-IDS: Continual Novelty Detection for Intrusion Detection Systems. arXiv preprint arXiv:2502.14094.
5. Almabdy, S., & Ullah, A. (2025). Optimising Intrusion Detection Systems in Cloud-Edge Continuum with Knowledge Distillation for Privacy-Preserving and Efficient Communication. arXiv preprint arXiv:2504.10698.
6. Singal, K., Kandhoul, N., & Dhurander, S. K. (2025). Evolutionary LightGBM-Based Intrusion Detection System for IoT Networks. International Journal of Communication Systems, 38(5), e70031.
7. Yu, H., Zhang, W., Kang, C., & Xue, Y. (2024). A feature selection algorithm for intrusion detection system based on the enhanced heuristic optimizer. Expert Systems with Applications, 125860.
8. Farrukh, Y. A., Wali, S., Khan, I., & Bastian, N. D. (2025). Xg-nid: Dual-modality network intrusion detection using a heterogeneous graph neural network and large language model. Expert Systems with Applications, 128089.
9. Najafli, S., Toroghi Haghighat, A., & Karasfi, B. (2024). A novel reinforcement learning-based hybrid intrusion detection system on fog-to-cloud computing. The Journal of Supercomputing, 80(18), 26088-26110.
10. Chowdary, P. B. K., Udayakumar, R., Jadhav, C., Mohanraj, B., & Vimal, V. R. (2024). Efficient Intrusion Detection Solution for Cloud Computing Environments Using Integrated Machine Learning Methodologies. Journal of Wireless Mobile Networks, Ubiquitous Computing and Dependable Applications, 15(2), 14-26.

11. Long, Z., Yan, H., Shen, G., Zhang, X., He, H., & Cheng, L. (2024). A Transformer-based network intrusion detection approach for cloud security. Journal of Cloud Computing, 13(1), 5.
12. Al-Ghuwairi, A. R., Sharrab, Y., Al-Fraihat, D., AlElaimat, M., Alsarhan, A., & Algarni, A. (2023). Intrusion detection in cloud computing based on time series anomalies utilizing machine learning. Journal of Cloud Computing, 12(1), 127.
13. Nizamudeen, S. M. T. (2023). Intelligent intrusion detection framework for multi-clouds–IoT environment using swarm-based deep learning classifier. Journal of Cloud Computing, 12(1), 134.
14. Fatani, A., Dahou, A., Abd Elaziz, M., Al-Qaness, M. A., Lu, S., Alfadhli, S. A., & Alresheedi, S. S. (2023). Enhancing intrusion detection systems for IoT and cloud environments using a growth optimizer algorithm and conventional neural networks. Sensors, 23(9), 4430.
15. Mohamed, D., & Ismael, O. (2023). Enhancement of an IoT hybrid intrusion detection system based on fog-to-cloud computing. Journal of Cloud Computing, 12(1), 41.
16. Khordadpour, P., & Ahmadi, S. (2023). FIDS: Fuzzy Intrusion Detection System for simultaneous detection of DoS/DDoS attacks in Cloud computing. arXiv preprint arXiv:2305.16389.
17. Gourceyraud, M., Salem, R. B., Neal, C., Cuppens, F., & Cuppens, N. B. (2025). Federated intrusion detection system based on unsupervised machine learning. arXiv preprint arXiv:2503.22065.
18. Bamber, S. S., Katkuri, A. V. R., Sharma, S., & Angurala, M. (2025). A hybrid CNN-LSTM approach for intelligent cyber intrusion detection system. Computers & Security, 148, 104146.
19. Mary, D. S., Dhas, L. J. S., Deepa, A. R., Chaurasia, M. A., & Sheela, C. J. J. (2024). Network intrusion detection: An optimized deep learning approach using big data analytics. Expert Systems with Applications, 251, 123919.
20. Samriya, J. K., Kumar, S., Kumar, M., Wu, H., & Gill, S. S. (2024). Machine learning based network intrusion detection optimization for cloud computing environments. IEEE Transactions on Consumer Electronics.
21. Attou, H., Mohy-eddine, M., Guezzaz, A., Benkirane, S., Azrour, M., Alabdultif, A., & Almusallam, N. (2023). Towards an intelligent intrusion detection system to detect malicious activities in cloud computing. Applied Sciences, 13(17), 9588.
22. Kumar, V., Kumar, V., Singh, N., & Kumar, R. (2023). Enhancing intrusion detection system performance to detect attacks on Edge of things. SN Computer Science, 4(6), 802.
23. Yang, Y. M., Chang, K. C., & Luo, J. N. (2025). Hybrid Neural Network-Based Intrusion Detection System: Leveraging LightGBM and MobileNetV2 for IoT Security. Symmetry, 17(3), 314.
24. Susilo, B., Muis, A., & Sari, R. F. (2025). Intelligent Intrusion Detection System Against Various Attacks Based on a Hybrid Deep Learning Algorithm. Sensors, 25(2), 580.
25. Huang, J., Chen, Z., Liu, S. Z., Zhang, H., & Long, H. X. (2024). Improved Intrusion Detection Based on Hybrid Deep Learning Models and Federated Learning. Sensors, 24(12), 4002.
26. Surbhi, Chauhan, N. R., & Dahiya, N. (2025). Optimizing XGBoost hyperparameters using the dragonfly algorithm for enhanced cyber attack detection in the internet of healthcare things (IoHT). Cluster Computing, 28(4), 230.

27. Amouri, A., Al Rahhal, M. M., Bazi, Y., Butun, I., & Mahgoub, I. (2024, October). Enhancing Intrusion Detection in IoT Environments: An Advanced Ensemble Approach Using Kolmogorov-Arnold Networks. In 2024 International Symposium on Networks, Computers and Communications (ISNCC) (pp. 1-6). IEEE.
28. Boukela, L., Zhang, G., Yacoub, M., & Bouzefrane, S. (2021, June). A near-autonomous and incremental intrusion detection system through active learning of known and unknown attacks. In 2021 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC) (pp. 374-379). IEEE.
29. Almazmomi, N. K. (2024). Long short-term memory-based intrusion detection system using hybrid grid search and sequential chimp optimization algorithm-based hyperparameter tuning. Intelligent Decision Technologies, 18724981241291422.
30. Ullah, F., Ullah, S., Srivastava, G., & Lin, J. C. W. (2024). IDS-INT: Intrusion detection system using transformer-based transfer learning for imbalanced network traffic. Digital Communications and Networks, 10(1), 190-204.
31. Turukmane, A. V., & Devendiran, R. (2024). M-MultiSVM: An efficient feature selection assisted network intrusion detection system using machine learning. Computers & Security, 137, 103587.
32. Sadia, H., Farhan, S., Haq, Y. U., Sana, R., Mahmood, T., Bahaj, S. A. O., & Khan, A. R. (2024). Intrusion detection system for wireless sensor networks: A machine learning based approach. IEEE Access, 12, 52565-52582.
33. Zorarpaci, E. (2024). A fast intrusion detection system based on swift wrapper feature selection and speedy ensemble classifier. Engineering Applications of Artificial Intelligence, 133, 108162.
34. N. Ram and D. Kumar, "Effective Cyber Attack Detection in an IoMT-Smart System using Deep Convolutional Neural Networks and Machine Learning Algorithms," 2022 Second International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE), Bangalore, India, 2022, pp. 1-6, doi: 10.1109/ICATIECE56365.2022.10046735.
35. Devendiran, R., & Turukmane, A. V. (2024). Dugat-LSTM: Deep learning based network intrusion detection system using chaotic optimization strategy. Expert Systems with Applications, 245, 123027.
36. Roshan, K., Zafar, A., & Haque, S. B. U. (2024). Untargeted white-box adversarial attack with heuristic defence methods in real-time deep learning based network intrusion detection system. Computer Communications, 218, 97-113.
37. Kumar, N., & Sharma, S. (2023). A hybrid modified deep learning architecture for intrusion detection system with optimal feature selection. Electronics, 12(19), 4050.
38. Qaddos, A., Yaseen, M. U., Al-Shamayleh, A. S., Imran, M., Akhunzada, A., & Alharthi, S. Z. (2024). A novel intrusion detection framework for optimizing IoT security. Scientific Reports, 14(1), 21789.
39. Hart, M., Dave, R., & Richardson, E. (2023). Next-Generation Intrusion Detection and Prevention System Performance in Distributed Big Data Network Security Architectures. International Journal of Advanced Computer Science and Applications, 14(9).
40. Gulia, N., Solanki, K., Dalal, S., Dhankhar, A., Dahiya, O., & Salmaan, N. U. (2023). Intrusion Detection System Using the G-ABC with Deep Neural Network in Cloud Environment. Scientific Programming, 2023(1), 7210034.