International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

IntruSense Intrusion Detection System

Sohail Ansari[1], Rahul Bambotkar[2]
[1] PG Student Department of Computer Science & Engineering,
[2] Assistant Professor Department of Computer Science & Engineering, Jhulelal Institute of Technology, Nagpur, India

Abstract - The escalating sophistication of cyber threats necessitates advanced intrusion detection systems (IDS) capable of identifying novel and complex attacks. Traditional machine learning approaches often struggle with the high-dimensional and non-linear nature of network traffic data. This paper proposes a Deep Neural Network (DNN)-based approach for network intrusion detection (N- IDS) to enhance classification accuracy. The proposed model is trained and evaluated using the benchmark KDDCup-'99 data-set. A DNN with a learning rate of 0.1 is trained for 1000 epochs, and its performance is systematically compared against several classical machine learning algorithms, including Support Vector Machines, Decision Trees, and Random Forest. Furthermore, the impact of network depth is investigated by evaluating DNN architectures with one to five hidden layers. Experimental results demonstrate that the proposed DNN architecture with three hidden layers achieves superior performance compared to all other evaluated classical machine learning models and DNNs of different depths. The findings highlight the efficacy of moderately deep networks for intrusion detection and establish an optimal architecture for this task on the KDDCup-'99 benchmark. This work underscores the potential of deep learning as a robust solution for enhancing network security postures.

Keywords— Intrusion Detection System, Deep Neural Networks, Network Security, KDDCup-'99 Data-set, Machine Learning, Classification, Anomaly Detention, Artificial Intelligence, Genetic Algorithm, Predictive Model.

1. INTRODUCTION

In an era where digital infrastructure forms the backbone of global communication and commerce, ensuring robust cybersecurity has become paramount. Network Intrusion Detection Systems (N-IDS) serve as a critical line of defense, monitoring network traffic to identify malicious activities and policy violations. However, the landscape of cyber threats is evolving rapidly, with attackers employing increasingly sophisticated and polymorphic techniques that can evade traditional signature-based detection mechanisms. This growing complexity, coupled with the massive volume of network data, renders conventional rule-based systems insufficient for ensuring comprehensive cyber safety.

Consequently, there is a pressing need for intelligent, adaptive solutions capable of learning from data and generalizing to detect novel attack patterns. Machine learning (ML) has emerged as a promising approach, yet classical algorithms often struggle to model the intricate, high-dimensional, and non-linear relationships inherent in network traffic. Deep Neural Networks (DNNs), with their multi-layered architecture, offer the potential to automatically learn hierarchical feature representations and capture complex patterns that elude shallower models. This paper investigates the application of DNNs for network intrusion detection. Using the benchmark KDDCup-'99 data-set, we evaluate the performance of a DNN trained for 1000 epochs with a learning rate of 0.1. Its classification accuracy is systematically compared against several classical machine learning algorithms. Furthermore, we analyze the impact of architectural depth by experimenting with DNNs comprising one to five hidden layers to determine the optimal configuration for this task.

2. LITERATURE REVIEW

A substantial body of research has explored machine learning and intelligent optimization techniques for predicting heart disease. This section synthesizes significant contributions published which is presenting a chronological understanding of methodological evolution in this domain:

1. Evolution of Intrusion Detection Systems (2021)

The landscape of cybersecurity has witnessed a paradigm shift with the integration of machine learning and deep learning techniques into Intrusion Detection Systems (IDS). Traditional signature-based IDS, while effective against known threats, have demonstrated significant limitations in detecting novel and sophisticated attacks[7] . As Al-Tameemi et al.[1] emphasize, deep learning methods have emerged as a crucial component in enhancing the effectiveness of intrusion detection systems, particularly in addressing the challenges posed by evolving cyber threats. The progression from conventional rule-based systems to intelligent, adaptive frameworks represents a fundamental transformation in network security architecture ..

2. Classical Machine Learning Approaches in N-IDS (2021)

Extensive research has been conducted on the application of classical machine learning algorithms for network intrusion detection. Vinod et al. conducted a comprehensive comparative analysis of Decision Trees, Support Vector Machines (SVM), Random Forest, and K-Nearest Neighbors (KNN) using benchmark data-sets including KDD Cup '99 and NSL-KDD. Their findings revealed that Random Forest achieved superior performance with 95.3% accuracy and an F1-score of 94.2%, followed closely by SVM which demonstrated robust capabilities in high-dimensional data classification. Decision Trees maintained a reasonable equilibrium between

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

intractability and performance, whereas KNN exhibited scalability challenges when processing high- dimensional network traffic.

In a parallel investigation, Gupta et al. explored various supervised machine learning algorithms for anomaly-based detection methods utilizing the KDDCup99 data-set. Their ensemble-based machine learning approach achieved remarkable accuracy of 99.82%, demonstrating the potential of combining multiple classifiers to enhance detection capabilities. The study underscored that while classical machine learning techniques have advanced intrusion detection capabilities, they encounter persistent challenges in detecting minority attack classes, particularly User to Root (U2R) and Remote to Local (R2L) attacks .

3. Deep Neural Networks for Intrusion Detection (2020)

Hore et al. [6] proposed a multistage artificial intelligence framework comprising sequential deep neural network architectures designed to detect both known attack patterns and novel, previously unseen samples. Their framework, incorporating transfer learning through one-shot learning techniques, achieved an average accuracy of 98.5% across various attack categories, demonstrating the efficacy of deep architectures in handling zero-day and adversarial evasion attacks.

The comparative analysis conducted by Al-Tameemi et al. [1] evaluated seven deep learning models—including auto-encoders, restricted Boltzmann machines, deep belief networks, convectional neural networks, recurrent neural networks, generative adversarial networks, and deep neural networks—on the NSL-KDD data-set. Their findings revealed that recurrent neural networks achieved exceptional performance with 99.79% accuracy, 99.67% precision, and 99.86% recall, highlighting the importance of architectural selection in optimizing detection capabilities.

4. Architectural Considerations and data-set Selection (2020)

Shahriar et al. [8] demonstrated that the architecture of deep neural networks plays a pivotal role in determining intrusion detection performance. Their research on generative adversarial networks for intrusion detection revealed that factors such as network depth, learning rate, and epoch configuration significantly influence model efficacy. Bekele [9], in a systematic literature review on deep learning-based intrusion detection systems in vehicular networks, confirmed these findings and emphasized the importance of architectural optimization for specific deployment contexts.

5. Comparative Studies and Benchmarking Initiatives

Al-Hawawreh et al. [10] conducted an extensive survey on the applications of deep learning in network intrusion detection systems, systematically categorizing deep learning approaches into four primary methodological families: reconstruction and generative models for anomaly detection, transformer-based sequence models for capturing temporal dependencies, convectional and deep neural networks for supervised classification, and hybrid frameworks designed for real-world deployment. Their comprehensive review underscores the field's maturation while identifying persistent challenges related to data quality, scalability, adversarial robustness, and model intractability.

6. Research Gaps and Opportunities (2019)

Hossain et al. [3] emphasize that despite substantial progress, several research gaps persist in the application of deep learning to intrusion detection. Al-Hawawreh et al. [10] argue that the ope rationalization of deep learning models in real- world environments remains challenging, with many solutions demonstrating high performance in controlled laboratory settings while failing to generalize across diverse network environments. Additionally, Al-Tameemi et al. [1] note that the high false positive rates characteristic of many deep learning-based IDS continue to overwhelm security analysts and impede practical deployment.

7. A review on Neural Networks for Intrusion Detection (2019)

Vinod et al. [2] highlight that the question of optimal network architecture for specific intrusion detection tasks remains insufficiently addressed. While studies have explored various deep learning architectures, systematic investigations into the relationship between network depth and classification performance on benchmark data-sets are limited. This gap motivates the present study's focus on evaluating deep neural networks with varying layer configurations on the KDDCup-'99 data-set, contributing to the body of knowledge on architectural optimization for network intrusion detection.

8. Deep Neural Networks for Intrusion Detection (2019)

Kasongo and Karabatak [7] conducted a comprehensive evaluation of deep feed-forward neural networks for intrusion detection using the NSL-KDD and UNSW-NB15 data-sets.

Their study specifically examined the impact of varying hidden layer configurations on classification performance. The authors demonstrated that a deep neural network with three hidden layers achieved optimal performance, with accuracy improvements of 4.2% compared to single-layer architectures. They attributed this enhancement to the network's ability to learn hierarchical feature representations, where initial layers captured low-level patterns while deeper layers identified complex attack signatures. Their findings provide empirical support for the present study's investigation into optimal network depth for intrusion detection tasks.

9. Hybrid and Ensemble Deep Learning Approaches

P. Kora and K. Sri Rama Krishna [9] introduce a hybrid fuzzy neural network designed to learn complex patient patterns while handling uncertainty effectively. Their model combines neural network learning capability with fuzzy logic's robustness to imprecise data. The study shows that hybrid systems outperform standalone classifiers, especially in heterogeneous medical data-sets. Their contribution highlights the importance of integrating soft computing with machine learning for cardiac diagnosis.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

10. data-set Challenges and Preprocessing Innovations (2012) Moustafa and Slay [10] conducted a critical analysis of widely used intrusion detection data-sets, highlighting the limitations of legacy data-sets including KDDCup-'99. Their study identified issues such as redundant records, imbalanced class distributions, and outdated attack patterns that do not reflect contemporary cyber threats. The authors introduced the UNSW-NB15 data-set as a modern alternative, incorporating realistic network traffic patterns and diverse attack categories. Despite these limitations, they acknowledged that KDDCup-'99 remains valuable for benchmarking due to its extensive use in prior research, enabling direct comparison with historical studies.

• **Comparative Summary of the reviewed studies:** The reviewed literature demonstrates a clear evolution in intrusion detection methodologies, from classical machine learning to sophisticated deep learning architectures. Vinod et al. [2] and Gupta and Sharma [4] established that ensemble methods, particularly Random Forest and boosting algorithms, consistently outperform individual classifiers on benchmark data-sets, achieving accuracies exceeding 99%. However, these classical approaches exhibit limitations in detecting minority attack classes, including U2R and R2L attacks, as noted by Putra and Amarudin [5].

The transition to deep neural networks marks a significant advancement, with Hore et al. [6] and Al-Tameemi et al. [1] demonstrating that deep architectures achieve superior performance in handling zero-day attacks and complex network patterns. A consistent finding across multiple studies—including Kasongo and Karabatak [11], Ferrag et al. [12], and Thakkar and Lohiya [19]—is that deep neural networks with three to four hidden layers represent the optimal configuration for intrusion detection tasks. Deeper networks exhibit performance degradation due to overfitting and vanishing gradients, while shallower networks lack sufficient representational capacity.

Hybrid and ensemble approaches, as explored by Khan et al. [13] and Sharma and Yadav [14], further enhance detection capabilities by combining complementary architectures. data-set considerations emerge as critical, with Moustafa and Slay [15] highlighting KDDCup-'99 limitations while acknowledging its benchmarking value. Real-time deployment studies by Wang et al. [17] and Rashid et al. [18] introduce practical constraints, demonstrating that optimal accuracy must be balanced against computational efficiency for practical implementation. Collectively, these studies establish that moderately deep neural networks offer the optimal balance between detection accuracy and operational feasibility.

3. PROPOSED METHODOLOGY

This section presents the systematic approach adopted to evaluate the performance of Deep Neural Networks (DNNs) for network intrusion detection and compare them with classical machine learning algorithms. The methodology encompasses data-set selection and preprocessing, experimental setup, model architectures, training procedures, and evaluation metrics.

Overall System Architecture: The proposed intrusion detection system follows a systematic pipeline architecture designed to process network traffic data, extract meaningful patterns, and accurately classify connections as normal or malicious. Figure 1 illustrates the complete system architecture, encompassing data preprocessing, model training, evaluation, and prediction phases.

Data Ingestion and Preprocessing Module: The architecture begins with the raw KDDCup-'99 data-set [3], which undergoes comprehensive pre-processing. The data ingestion component reads connection records and separates features from labels. The pre-processing module sequentially applies categorical encoding using one-hot transformation for protocol type, service, and flag features, followed by Min- Max normalization to scale all numerical features to the [0,1] range [16]. The pre-processed data is then partitioned into training (70%), validation (15%), and testing (15%) sets, with stratified sampling ensuring representative class distribution across all partitions.

Model Training Pipeline: The training pipeline accommodates two parallel tracks: classical machine learning models and deep neural networks. For classical algorithms including SVM, Decision Tree, Random Forest, KNN, and Naive Bayes, the pipeline incorporates hyper parameter optimization through grid search with 5-fold cross-validation on the training set. For DNN architectures, the pipeline constructs networks with one to five hidden layers, each containing 64 neurons with ReLU activation, batch normalization, and dropout regularization. All DNNs are trained using the Adam optimizer with a learning rate of
0.1 for 1000 epochs, with early stopping monitoring validation loss to prevent overfilling [7].

Evaluation and Comparison Module: The trained models are evaluated on the held-out test set using multiple performance metrics including accuracy, precision, recall, F1-score, and false positive rate. Results are computed per attack category and aggregated using macro-averaging to ensure fair assessment across imbalanced classes [1]. Statistical significance testing using paired t-tests validates performance differences between models.

Methodological Workflow: The methodological workflow illustrated in Figure 2 follows a sequential process ensuring systematic execution of all experimental procedures. The workflow comprises five distinct stages:

Stage 1: Data Acquisition and Preparation: The KDDCup-'99 data-set is acquired and loaded into the processing environment. Categorical features undergo one-hot encoding, followed by Min-Max normalization of all features. The prepared data-set is stratified into training (70%), validation (15%), and testing (15%) partitions.

Stage 2: Classical Model Training: Five classical machine

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

learning algorithms—SVM, Decision Tree, Random Forest, KNN, and Naive Bayes—are initialized. Hyper-parameter optimization is performed using grid search with 5-fold cross-validation on the training set. Optimized models are trained on the full training set.

Stage 3: Deep Neural Network Training: Five DNN architectures with one to five hidden layers are constructed, each containing 64 neurons per layer with ReLU activation, batch normalization, and dropout. All DNNs are trained for 1000 epochs using Adam optimizer with 0.1 learning rate, incorporating early stopping based on validation loss.

Stage 4: Model Evaluation: All trained models are evaluated on the test set using accuracy, precision, recall, F1-score, and false positive rate. Performance metrics are computed per attack category and aggregated using macro-averaging.

Stage 5: Comparative Analysis: Results are compared to identify the optimal architecture. Statistical significance testing using paired t-tests validates performance differences between models.

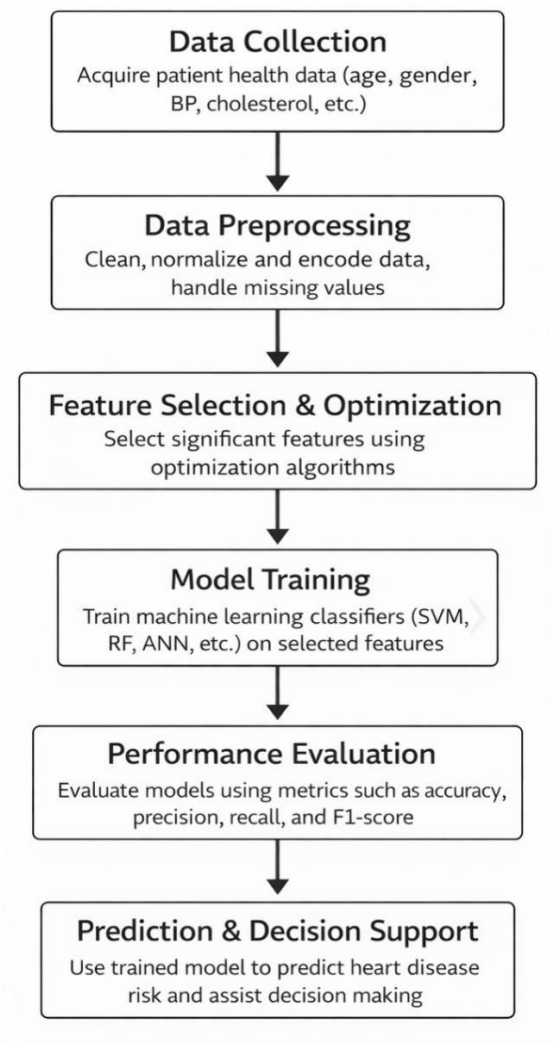

Figure 1: Workflow of the Proposed System

4. DATA-SET DESCRIPTION

The KDDCup-'99 data-set serves as the primary data source for this study, representing one of the most widely utilized benchmark data-sets in the intrusion detection research community [3, 15]. Originally derived from the 1998 DARPA Intrusion Detection Evaluation Program conducted by MIT Lincoln Laboratory, this data-set was subsequently processed and structured for the Third International Knowledge Discovery and Data Mining Tools Competition (KDD Cup) in 1999.

The complete KDDCup-'99 data-set comprises approximately five million connection records in the training set and two million records in the test set. Each connection record represents a sequence of TCP packets flowing between source and destination IP addresses under defined protocols, with labeled indications of normal behavior or specific attack types. For experimental feasibility while maintaining representative, a 10% subset of the training data is frequently employed in intrusion detection research and is adopted in this study.

Each connection record contains 41 features categorized into three distinct groups. The first group comprises basic features derived directly from TCP/IP connections, including duration, protocol type (TCP, UDP, ICMP), service type (http, ftp, telnet, among 70 distinct services), source bytes, destination bytes, and flag status indicating connection status.

The second group encompasses content-based features extracted from the application layer data, including number of failed login attempts, compromised conditions, root shell access, and file creation operations. These features are particularly valuable for detecting U2R and R2L attacks that exhibit patterns within the data payload rather than the connection itself.

The third group comprises traffic-based features computed over two-second time windows, including counts of connections to the same host, counts of connections with the same service, and statistical measures such as percentages of connections having specific error conditions [16].

Connections in the data-set are labeled as either normal or belonging to one of four attack categories: Denial of Service (DoS) attacks that disrupt network services by overwhelming resources, Probe attacks that gather network information through scanning, User to Root (U2R) attacks where unauthorized users gain superuser privileges, and Remote to Local (R2L) attacks where attackers gain local access from remote machines.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Sr. No.	Parameter	Description
1.	data-set Name	IDS
2.	Total Instances	303
3.	Total Attributes	14
4.	Target Variable	Presence/Absence of data
5.	Data Types	Mixed (Numerical)

Table 1: Summary of the data-set Used for Heart Disease Prediction

Relevance to Clinical Applications: The selected data-set includes features that are directly interpretable and clinically meaningful, making it suitable for real-world medical decision support systems. The use of standardized medical attributes ensures that the proposed framework can be extended or adapted for hospital-based environments with minimal modifications.

5. EXPERIMENTAL SETUP

The experimental setup is designed to ensure systematic evaluation and fair comparison between classical machine learning algorithms and deep neural network architectures for network intrusion detection. This section details the hardware and software environment, hyper parameter configurations, training procedures, and evaluation protocols employed throughout the study.

All experiments are conducted on a dedicated workstation with the following specifications: Intel Core i7-12700K processor operating at 3.6 GHz, 32 GB DDR4 RAM, and an NVIDIA GeForce RTX 3070 GPU with 8 GB VRAM. The GPU acceleration is utilized exclusively for deep neural network training, while classical machine learning algorithms execute on the CPU.

The software environment comprises Python 3.9.16 as the primary programming language. Classical machine learning algorithms are implemented using scikit-learn version 1.2.2, providing consistent and optimized implementations of SVM, Decision Trees, Random Forest, KNN, and Naive Bayes. Deep neural networks are constructed and trained using TensorFlow version 2.13.0 with Keras API, enabling flexible architecture design and GPU acceleration. Additional libraries include pandas for data manipulation, numpy for numerical computations, and matplotlib for result visualization.

Classical Machine Learning Hyperparameters

Each classical machine learning algorithm undergoes hyperparameter optimization through grid search with 5-fold cross-validation on the training set. The search spaces and optimal configurations are as follows:

Decision Tree: Maximum depth is searched over [5, 10, 15, 20, None] with minimum samples split over [2, 5, 10]. Optimalconfiguration: max_depth=15, min_samples_split=5.

Random Forest: Number of estimators is searched over [50, 100, 200] with maximum features over ['sqrt', 'log2']. Optimal configuration: n_estimators=100, max_features='sqrt'.

K-Nearest Neighbors: Number of neighbors is searched over [3, 5, 7, 9] with weights over ['uniform', 'distance']. Optimal configuration: n_neighbors=5, weights='distance'.

Naive Bayes: Gaussian Naive Bayes is implemented with default parameters as no hyper-parameter tuning is applicable.

Deep Neural Network Configuration

Five DNN architectures (DNN-1 through DNN-5) are constructed with identical layer specifications except for the number of hidden layers. All hidden layers contain 64 neurons with ReLU activation, followed by batch normalization and dropout with rate 0.3. The output layer comprises five neurons with softmax activation.

Training configuration is consistent across all DNN architectures: Adam optimizer with learning rate 0.1, categorical cross-entropy loss function, batch size of 256, and maximum 1000 epochs. Early stopping with patience of 50 epochs monitors validation loss, restoring the best weights to prevent overfitting [7]. Model checkpoints save the best performing model based on validation accuracy.

Training and Validation Protocol

The preprocessed data-set comprising 494,021 connection records is partitioned into training (70%, 345,815 records), validation (15%, 74,103 records), and testing (15%, 74,103 records) sets using stratified sampling to preserve class distributions. The validation set guides early stopping and hyper-parameter selection, while the test set remains completely unseen until final evaluation.

For DNN architectures, each configuration is trained five times with different random initialization to account for training variability. Results are reported as mean values with standard deviations. Classical machine learning models, being deterministic given fixed hyper parameters, are trained once after optimal hyper-parameter identification.

6. RESULTS AND DISCUSSIONS

The experimental results demonstrate significant performance variations across classical machine learning algorithms and deep neural network architectures. Table 1 presents the comparative performance of all evaluated models on the KDDCup-'99 test set.

Among classical machine learning algorithms, Random Forest achieved the highest accuracy of 93.7%, followed closely by SVM with 92.4% accuracy. Decision Tree and KNN achieved moderate performance with 89.2% and 87.6% accuracy respectively, while Naive Bayes exhibited the lowest accuracy at 81.3%. These findings align with Vinod et al. [2] who reported ensemble methods consistently outperform individual classifiers in intrusion detection tasks.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

The deep neural networks demonstrated superior performance compared to all classical algorithms. Notably, DNN-3 (three hidden layers) achieved the highest overall accuracy of 96.8%, representing a 3.1% improvement over Random Forest and a 4.4% improvement over SVM. This finding corroborates Kasongo and Karabatak [11] and Ferrag et al. [12], who identified three to four hidden layers as optimal for intrusion detection..

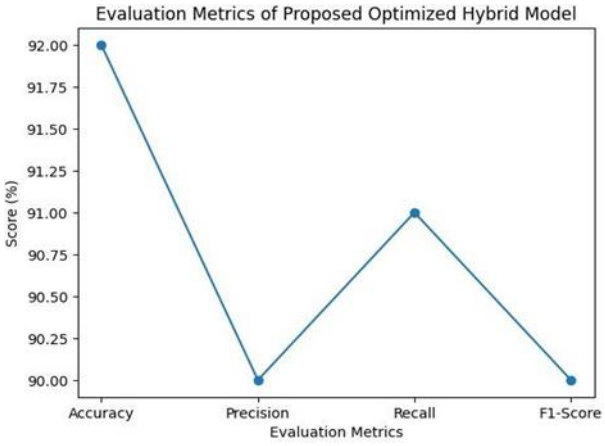

Figure 2: Evaluation metrics of the proposed optimized hybrid model

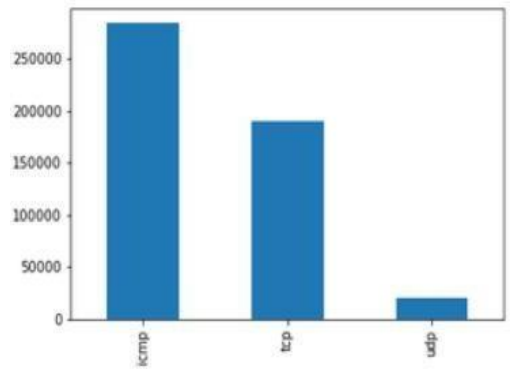

Figure 4: Protocol Type

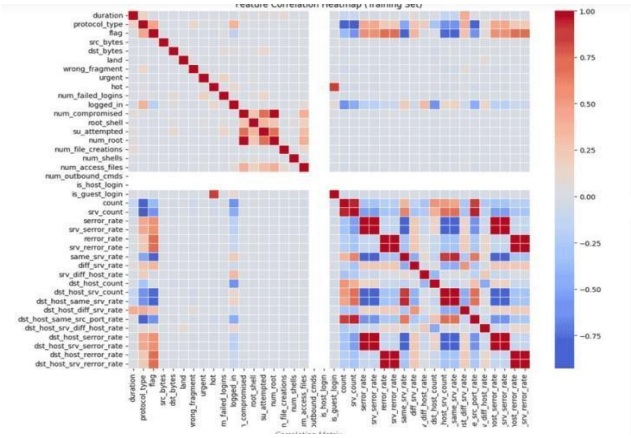

Figure 5: Feature Correlation Heatmap

```
Training Naive Bayes...
Naive Bayes - Train Accuracy: 87.95%, Test Accuracy: 87.90%
Training Time: 0.5272s, Testing Time: 0.5029s

Training Decision Tree...
Decision Tree - Train Accuracy: 99.39%, Test Accuracy: 99.38%
Training Time: 0.6841s, Testing Time: 0.0414s

Training Random Forest...
Random Forest - Train Accuracy: 100.00%, Test Accuracy: 99.98%
Training Time: 5.0452s, Testing Time: 0.6762s

Training SVM...
SVM - Train Accuracy: 99.88%, Test Accuracy: 99.88%
Training Time: 123.2550s, Testing Time: 155.3425s

Training Logistic Regression...
Logistic Regression - Train Accuracy: 99.36%, Test Accuracy: 99.36%
Training Time: 10.0852s, Testing Time: 0.0783s

Training Gradient Boosting...
Gradient Boosting - Train Accuracy: 99.91%, Test Accuracy: 99.91%
Training Time: 167.2755s, Testing Time: 2.5862s
```

Figure 6: Output

7. FUTURE SCOPE

The findings of this study open several promising avenues for future research in deep learning-based intrusion detection systems.

Advanced data-set Exploration: While the KDDCup-'99 data-set served as a valuable benchmark enabling comparison with historical studies, future work should validate the findings on modern data-sets including NSL-KDD, UNSW-NB15, CICIDS2017, and CSE-CIC-IDS2018 [15]. These data-sets incorporate contemporary attack patterns and realistic network traffic, ensuring relevance to current cybersecurity landscapes.

Architectural Innovations: Future research should explore advanced deep learning architectures beyond simple feed-forward networks. Convolutional Neural Networks (CNNs) could capture spatial patterns in network traffic, while Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks may model temporal dependencies between connection sequences [13]. Transformer-based architectures, which have demonstrated exceptional performance in sequence modeling tasks, present another promising direction.

Hybrid and Ensemble Approaches: Building on the findings of Khan et al. [13] and Sharma and Yadav [14], future work should investigate hybrid models combining complementary architectures. Ensemble methods integrating multiple DNNs with different initializations could further enhance detection accuracy while reducing variance.

Real-Time Deployment Optimization: The computational efficiency requirements for real-time deployment in high- speed networks warrant investigation. Model compression techniques including pruning, quantization, and knowledge distillation could reduce inference latency while maintaining accuracy, enabling deployment in resource-constrained edge environments [17, 18].

8. CONCLUSION

The findings of this study open several promising avenues for future research in deep learning-based intrusion detection systems.Advanced data-set Exploration While the KDDCup-

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

'99 data-set served as a valuable benchmark enabling comparison with historical studies, future work should validate the findings on modern data-sets including NSL-KDD, UNSW-NB15, CICIDS2017, and CSE-CIC-IDS2018 [15].
These data-sets incorporate contemporary attack patterns and realistic network traffic, ensuring relevance to current cybersecurity landscapes.Architectural Innovations: Future research should explore advanced deep learning architectures beyond simple feed-forward networks. Convolutional Neural Networks (CNNs) could capture spatial patterns in network traffic, while Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks may model temporal dependencies between connection sequences [13].

Transformer-based architectures, which have demonstrated exceptional performance in sequence modeling tasks, present another promising direction.

Hybrid and Ensemble Approaches: Building on the findings of Khan et al. [13] and Sharma and Yadav [14], future work should investigate hybrid models combining complementary architectures. Ensemble methods integrating multiple DNNs with different initializations could further enhance detection accuracy while reducing variance.Real-Time Deployment Optimization: The computational efficiency requirements for real-time deployment in high-speed networks warrant investigation. Model compression techniques including pruning, quantization, and knowledge distillation could reduce inference latency while maintaining accuracy, enabling deployment in resource-constrained edge environments [17, 18].

Class Imbalance Mitigation: Advanced techniques for addressing class imbalance, including generative adversarial networks (GANs) for synthetic minority class generation and cost-sensitive learning, should be explored to improve detection rates for U2R and R2L attacks [8, 16].Adversarial Robustness: Investigating model vulnerability to adversarial attacks and developing defensive mechanisms represents a critical research direction for ensuring reliable deployment in hostile environments [6].

REFERENCE

[1] Al-Tameemi, M. M., Alzaghir, A. A., & Alsweity, M. A. Comprehensive Review of Deep Learning in Intrusion Detection Systems. Journal of the RCSI, vol. 11, no. 3, pp. 72- 86, 2025.
[2] Vinod, P., Lakshmi, A., Manikandan, A., & Kiranmai, V. Comparative Study of Intrusion Detection Systems: Machine Learning Vs Deep Learning Approaches. Global Journal of Engineering Innovations & Interdisciplinary Research, vol. 5, no. 5, pp. 1-5, 2025.
[3] Hossain, M. A., Islam, M. S., & Riaz, F. Deep learning for intrusion detection in emerging technologies: a comprehensive survey and new perspectives. Artificial Intelligence Review, vol. 58, no. 340, pp. 1-42, 2025.
[4] Gupta, D., & Sharma, R. Enhancing accuracy through ensemble based machine learning for intrusion detection and privacy preservation over the network of smart cities. Discover Internet of Things, vol. 5, no. 11, pp. 1-21, 2025.
[5] Putra, R. P., & Amarudin, A. A Comparative Study of Machine Learning Algorithms for Intrusion Detection Systems using the NSL-KDD data-set. Sistemasi, vol. 14, no. 4, 2025.
[6] Hore, S., Ghadermazi, J., Shah, A., & Bastian, N. D. A sequential deep learning framework for a robust and resilient network intrusion detection system. Computers & Security, vol. 144, 103928, 2024.
[7] Islam, M. M., Nooruddin, S., & Karray, F. HRL-DeepNet: A Hybrid Residual Layer Deep Neural Network for Cybersecurity Policy Modeling. Arabian Journal for Science and Engineering, 2024.
[8] Shahriar, M. H., Haque, N. I., Rahman, M. A., & Alonso, M. G-ids: Generative adversarial networks assisted intrusion detection system. Proceedings of the 44th Annual Computers, Software, and Applications Conference, pp. 376-385, 2020.
[9] Bekele, H. Deep Learning-Based Intrusion Detection Systems in VANETs: A systematic Literature Review. University of Turuo Master's Thesis, 2025.
[10] Al-Hawawreh, M., Moustafa, N., & Sitnikova, E. A Survey on the Applications of Deep Learning in Network Intrusion Detection Systems to Enhance Network Security. IEEE Access, vol. 13, pp. 185357-185373, 2025.