Fusion-based Intrusion Detection System for Detecting Zero-day Attacks

Shaik Haseena
Department of CSE
Vignan's Foundation for Science, Technology & Research
Guntur, India
shaseena162@gmail.com

Shaik Mohammed Saif
Department of CSE
Vignan's Foundation for Science, Technology & Research
Guntur, India
saifmohammed07860@gmail.com

Polamarasetti Govind Rao
Department of CSE
Vignan's Foundation for Science, Technology & Research
Guntur, India
pg983081@gmail.com

Dr. Nerella Sameera
Assistant Professor, Department of CSE
Vignan's Foundation for Science, Technology & Research
Guntur, India
sameeracrrit@gmail.com

***Abstract*—Intrusion Detection Systems (IDS) are essential for protecting network infrastructure by identifying malicious activities, unauthorized access, and anomalies in real time. Despite advancements in the field, designing high-performance IDSs capable of addressing both known and emerging threats remains a significant challenge. This research addresses this issue through two primary objectives.**

The first objective focuses on detecting known cyberattacks by leveraging various Machine Learning (ML) and Deep Learning (DL) models, and analyzing which model delivers the best performance for IDS. Performance is measured using standard metrics—accuracy, precision, recall, and F1-score—on the UNSW-NB15 and SDN20 datasets.

The second objective tackles the more complex and critical problem of detecting zero-day attacks, which occur when a security vulnerability is exploited before it is publicly known or patched, posing a severe cybersecurity risk. To address this, a fusion-based intrusion detection system is proposed. Extensive experimentation is conducted using NSL-KDD for training and SDN20 for testing, with evaluation based on accuracy, correctly predicted attacks, missed attacks, and false positive rate.

Through well-designed experiments and comprehensive data analysis, results indicate that the proposed method significantly enhances detection performance for both known and zero-day threats, contributing to the development of a more resilient and adaptive IDS suitable for modern network environments.

***Index Terms*—Intrusion Detection Systems, Zero-Day Attacks, Cybersecurity Threats, Malicious Activities, Security Research, Machine Learning, Deep Learning, Fusion-Based Approaches.**

I. Introduction

The intrusion detection system (ID) has been created a foundational element of modern cybersecurity, tasked with monitoring network activity in real time to identify malicious behavior and unauthorized access. In IDSThere are two types: network-based ID (NID) and host-based ID (HID). while NIDS monitors network traffic in questionable patterns, HIDS focus on system-level indicators such as file changes, user behavior, and log activity. Detection techniques within IDS are generally It is It is divided into signatured based and an abnormal based approaches. The signature-based method is effective at identifying known threats by relying on predefined patterns, but they struggle with zero-day attacks that exploit previously unseen vulnerabilities. Anomaly-based techniques, by contrast, detect deviations from normal behavior, which makes them better suited for uncovering novel threats, though they often suffer from high false positive rates, reducing overall system reliability.

A central challenge in IDS research lies in selecting the most Effective Machine Learning and Deep Learning models, particularly in the absence of definitive benchmarks. Model selection and tuning often rely on trial and error, consuming valuable time and resources while complicating the development of practical IDS solutions. To address this, the present study is guided by two core objectives. The first involves assessing a range of Machine Learning and Deep Learning models of their effectiveness in detecting known cyberattacks and interpreting best solution of IDS, using widely recognized datasets such as UNSW-NB15 and SDN20. Evaluation is based on keyMetrics such as accuracy, recall, F1 score, False Positive Rate were provided well-rounded Views of each model's pros and cons when dealing with established threats.

The second focus of this study is the detection of zero-day attacks—those that take advantage of unknown vulnerabilities and frequently evade conventional defenses. To tackle this, a fusion-based IDS framework is proposed. This approach improves by integrating the strengths of several models the system's generalization capabilities and resilience against novel attacks. Training is conducted using the NSL-KDD dataset, while testing is performed on SDN20, simulating a realistic zero-day scenario. Performance is evaluated based on overall accuracy, the number of correctly detected attacks, missed detections, and false positive rates. By examining the models' ability to generalize across different datasets,

this research aims to enhance IDS robustness in identifying emerging threats.

In summary, this work presents a comparative analysis From various ML and DL model for identified attack detection and zero-day attacks.It proposes a fusion-based strategy that improves generalization and adaptability in intrusion detection systems. Through empirical evaluation using benchmark datasets— UNSW-NB15, SDN20, and This study involves NSL-KDD contributes a practical and optimized framework that can guide future developments in IDS research. The rest of this The paper as constructed as follows: described in Section II as Related work; described in Section III. as Datasets as well as methods used; Section IV details the Methodologies; Section V outlines the Experimental Results and Discussion; Section VI presents the Conclusion.

II. RELATED WORK

Zhang et al. [1] have inquired about Applying deep learning methods for intrusion detection cyber physical systems (CPS); which have been mostly autoencoders, CNNs, and LSTMs. The treatment of the paper brings a focus on the strong The possibilities of deep learning to take on the role of main actor in dealing with an exponentially increasing range of challenges, while, at the same time, it also points out the limitations such as the huge amount of data needed and the necessity of expensive hardware. Importantly, it is noted that the real-time deployment of the models in industrial environments is still a question without a clear answer.

Nura Shifa Musa et al. [2] was the use of variousDeep learning and machine learning approach networks (SDNs) defined in detection of DDOS attacks in software. Their work showed the benefit of usingsupervised, unsupervised and ensemble methods in the field of SDN security. They also mentioned that the real-world application of these methods was the most focused topic, however, the study stressed the importance of integrating the methods into practical environments at the same time.

Zahedi Azam et al. [3] performed a detailed comparative study of IDS models with a focus on decision tree-based methods. The paper discusses several ML and DL-based methods, comparing their performance on public datasets such as NSL-KDD. Decision trees were proved to be effective because they are simple and fast, and hence they can be used for real-time anomaly detection. They also have drawbacks like overfitting and decreased accuracy in dealing with complex, dynamic cyber threats.

Dheyaaldin Alsalman et al. [4] which includes machine learning(ML) models.An important feature of FusionNet is that it was built on the basis of the main principles of the different algorithms forDetecting attacks on IoT networks.The document outlines that the high practical potential of FusionNet in safety-sensitive areas like healthcare and industrial IoT was really emphasized.

Asmaa Halbouni et al. [5] was created by combining deep learning models. which uses CNNS extracts spatial characteristics and LSTM networks for recording time. pattern detection. The approach aims to further improve IDS performance by recognizing both structural and sequential traffic anomalies. This model was evaluated on three widely recognized data records, including CIC-IDS2017 and UNSW-NB15 along with WSN-DS, showing significant accuracy and low false alarm rates. Normalization and dropout layers were used to improve model generalization and stability.

Mirsadeghi et al. [6] dealt with the challange of class disparity in SDN-based intrusion detection by comparing ROS, SMOTE, GANs, and weighted Random Forest (WRF). The results indicated that RF performs better than deep learning algorithms in detecting the minority class, and balancing techniques such as ROS and SMOTE enhance detection but raise false negatives, and thus adaptive learning approaches are necessary.

H. Liao et al. [7] have carried out a survey in the application of Deep learning for IoT intrusion detection, indicating the advantages of CNN, RNN, LSTM, Autoencoders, and DBN in network security. Feature extraction, classification methods, and datasets are compared in their work. Data imbalance, real-time processing, and adaptive learning continue to be critical challenges despite enhancing IoT security with deep learning.

Mohammadi et al. [8] suggested an explainable AI (XAI)-guided method to enhance zero-day attack detection by combining ML models with SMOTE for class balancing. Their research focuses on interpretability and transparency in model decisions, enabling more trust among cybersecurity professionals. SMOTE application enabled overcoming class imbalance, greatly improving the recognition of infrequent unknown attacks.

Mohammad Shurmanet al. [9] introduced a scalable system for detecting and preventing zero-day attacks in Software defined networks (SDNs). Uses SDNS centralized control level, the system is able to do real-time network flow monitoring to identify anomalous behaviors. The research emphasized its flexibility and low latency response to new attack patterns in dynamic network environments.

Patil et al. [10] carried out a comparative case study comparing different methods employed for detecting and preventing zero-day attacks. The study pointed out both the advantages and limitations of these measures with emphasis on implementation issues in actual environments. The study adds value by providing insight into the feasibility of different defense measures in practice and deployment.

III. DATASETS DESCRIPTION

This research utilizes datasets containing comprehensive network traffic data, encompassing both benign and malicious activities. These datasets are essential for the training and evaluation of Intrusion Detection System (IDS) models. By providing a balanced mix of normal and attack traffic, they enable models to learn patterns associated with various cyber threats, thereby improving the ability to effectively identify and classify intrusions.

1) SDN20 Dataset: The SDN20 dataset captures a wide range of network interactions, from standard benign traffic to

various attack patterns, making it highly suitable for evaluating IDS models. It contains 80 features that describe different aspects of network behavior, allowing for a thorough analysis of traffic characteristics. The dataset is categorized into five labels: *BENIGN* (representing normal traffic) and four types of attacks, which include Distributed Denial of Service (DDoS) and web-based attacks such as brute force, cross-site scripting (XSS), and SQL injection. SDN20 is compiled from multiple types of traffic logs, with a particular focus on attack scenarios, making it well-suited for performance analysis of security models. The class distribution reveals that the *BENIGN* class constitutes the majority of the dataset, while DDoS attacks appear as the most dominant attack type among the malicious classes. Offering a broad and practical overview of real-world network traffic and cyberattack types, SDN20 serves as a robust foundation for training and testing IDS models. It significantly contributes to evaluating model accuracy and reliability across various intrusion forms and operational network environments.

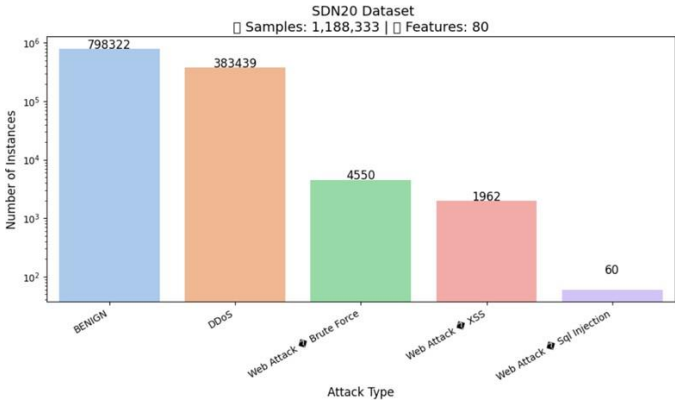


Fig. 1. *Traffic label distribution in the SDN20 dataset..*

2) UNSW-NB15 Dataset:

This dataset is one of the most widely adopted benchmarks in network intrusion detection research. It includes a blend of legitimate and attack-based traffic, making it suitable for IDS model training and validation. Designed by the Australian Centre for Cyber Security (ACCS), this dataset replicates realistic traffic flows and integrates controlled attack behaviors, offering a balanced and representative sample of modern cybersecurity challenges. UNSW-NB15 is especially valuable in machine learning-based security research due to its diversity and feature richness. It captures a wide range of network events, with clear labels distinguishing normal from malicious activity.

The dataset is divided into two parts: a training set containing 175,341 records and a test set with 82,332 records, allowing effective evaluation of model performance. Each record includes 49 features, ranging from packet-level metrics to broader flow-based statistics, which help machine learning algorithms detect behavioral anomalies in network traffic. The dataset consists of ten traffic labels, with Normal (BENIGN) representing legitimate activity and the remaining labels corresponding to different types of cyberattacks. This variety makes UNSW-NB15 a valuable resource for building and testing robust IDS models capable of handling diverse threat scenarios.

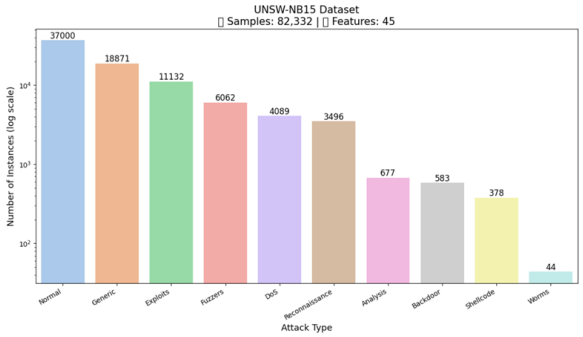


Fig. 2. *Traffic label distribution in the UNSW-NB15 dataset.*

3) NSL-KDD Dataset:

The NSL-KDD dataset is a long-standing standard used to assess the performance of intrusion detection systems. It consists of 41 input features and a target label that classifies each network session as normal or belonging to one of four primary categories of attacks: DoS (Denial of Service), Probe (surveillance and information gathering), R2L (Remote to Local), and U2R (User to Root). Each entry represents a single network session and includes a range of attributes such as protocol type, connection duration, service type, status flags, source and destination byte counts, and other statistical measures.

The test set used in this study contains approximately 22,500 records, offering a relatively balanced mix of normal and malicious traffic. This is a notable improvement over its predecessor, KDD'99, which suffered from class imbalance and data redundancy. NSL-KDD features both categorical and numerical data types, which require preprocessing before being fed into machine learning models. Preprocessing often includes techniques like label encoding or one-hot encoding to convert categorical variables into numerical formats suitable for learning algorithms.With its improved balance and detailed features, the NSL-KDD dataset continues to be a vital and widely trusted resource for advancing IDS performance and exploring complex zero-day intrusion scenarios .

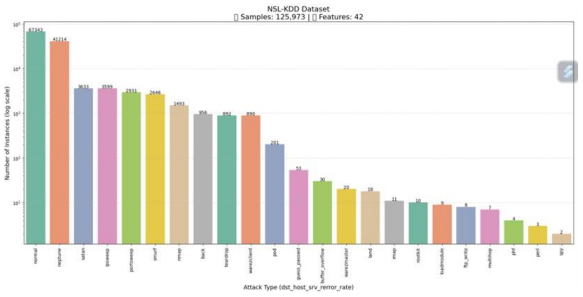


Fig. 3. *Traffic label distribution in the NSL-KDD dataset.*

IV. METHODOLOGIES

This study presents a dual-objective framework aimed at improving how Intrusion Detection Systems (IDS) detect both known and zero-day cyberattacks. The approach blends modern techniques from Machine Learning, Deep Learning, and fusion strategies to build a solution that's not only flexible but also scalable to meet the demands of today's cybersecurity landscape. The methodology is broken down into several key stages, each addressing different aspects of attack detection, ensuring a well-rounded approach to boosting the performance of IDS. The following section outlines the steps and methods used to reach the research goals.

a) ***Objective 1: Interpreting ML and DL models for IDS Known Attacks****:* This study focuses on detecting known cyberattacks and finding the best-performing model using two benchmark datasets: SDN20 and UNSW-NB15. The process begins with preparing the data by encoding categorical features and addressing class imbalances through oversampling and undersampling techniques.

The datasets are divided into training and testing sets, with 80% allocated for training and 20% for testing. Both sets are carefully cleaned to remove any missing values or inconsistencies, ensuring that models learn from high-quality data. To ensure fair contribution from all input variables, normalization is applied across all feature values. Additionally, Principal Component Analysis (PCA) is used for dimensionality reduction, preserving the most critical information while simplifying the dataset and speeding up model training without compromising performance.

b) ***Data Collection****:*

- This study uses two publicly available datasets: SDN20 and UNSW-NB15. Both datasets consist of labeled network traffic data, containing a mix of normal and malicious activity. These datasets provide a solid foundation for developing and evaluating intrusion detection models.

c) ***Preprocessing****:*

- The pre-processing step encompassed various essential processes to construct suitable training data for the models. Categorical features were first converted into numerical values to make them compatible with Machine Learning algorithms. To handle class imbalance and ensure that all attack types were fairly represented, resampling techniques were applied. The data records were then split with an 80:20 ratio for training and testing. Data cleaning was performed to remove any missing, incorrect, or inconsistent values. To balance the contribution of all features throughout the learning process, normalization was applied across all feature values. Finally, dimensionality reduction was performed using Principal Component Analysis (PCA) while retaining most of the meaningful variance in the data.

  In addition to these steps, correlation analysis was performed to identify and eliminate highly redundant features. By minimizing multicollinearity, the datasets became more robust and improved the interpretability of the models.

d) ***Model Training****:*

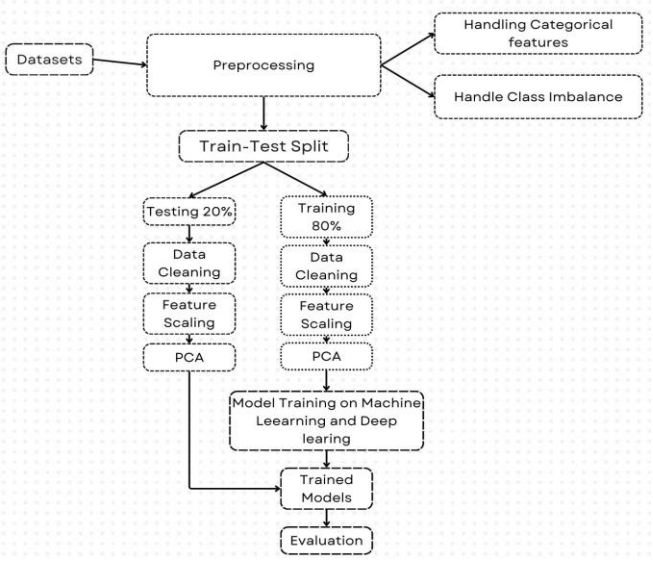
- The choice of models was guided by their proven effectiveness in prior intrusion detection studies and their ability to generalize across different types of network behaviors.

  A variety of Machine Learning and Deep Learning models were trained using the PCA-transformed, preprocessed data. These models aimed to learn distinctive patterns in network traffic that differentiate normal behavior from known attack signatures. Specifically, the models used included Decision Trees (DT), Random Forest, Na¨ıve Bayes, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Gated Recurrent Units (GRU), Artificial Neural Networks (ANN), CNN-RNN hybrid models, GRU-CNN hybrid models, and ensemble approaches combining Decision Trees and Random Forests.

e) ***Model Evaluation****:*

- After training, the models were evaluated using the testing set. To evaluate model performance, a confusion matrix was used to visualize prediction outcomes, while a classification report provided comprehensive metrics including Accuracy, Precision, Recall, F1-Score, and False Positive Rate.

  Additionally, Receiver Operating Characteristic (ROC) curves were generated to visually assess the trade-off between detection rates and false alarms. Such comprehensive evaluation helps in identifying the most practical and deployable models for real-world IDS environments.



Objective 1

Fig. 4. Model Architecture

Objective 2: *Fusion-Based Intrusion Detection for Zero-Day Attacks*

In this part of the study, the focus is on detecting zero-day attacks — threats that have not been encountered before and do not match any known patterns. To simulate and study such attacks, two widely used datasets were selected: **SDN20** and **NSL-KDD**.

The process starts with preparing the data. Both datasets are cleaned by removing any incorrect, missing, or inconsistent entries. Column names are standardized for clarity, and labels are renamed to maintain consistency. Categorical (non-numeric) values are converted into numerical form, and numerical features are scaled to ensure all data falls within a similar range. These steps prepare the data for effective training of machine learning models.

Once the data is clean and uniform, Principal Component Analysis (PCA) is applied to reduce the number of features while keeping the most important information intact. To gain deeper insight into the structure of the data, a technique called **T-distributed Stochastic Neighbor Embedding (t-SNE)** is used to create a 2D visual representation of the records. This helps in better understanding the distribution of normal and abnormal traffic patterns.

The next step involves simulating a realistic zero-day attack scenario. To do this, only 10% of the normal data from the **NSL-KDD** dataset is combined with the **SDN20** dataset containing limited information about attacks. This small sample mimics real-world conditions where systems have very little prior exposure to emerging threats. Meanwhile, the remaining portion of the **NSL-KDD** dataset, containing both normal and attack records, is used to train the model.

After the data records are combined and properly labeled, they are split into input features and output labels. The model is then trained to recognize whether incoming traffic is normal or potentially harmful. Once training is complete, the system can monitor new traffic in real-time, flagging any suspicious activity that may indicate a zero-day attack. This fusion-based approach enhances early detection of emerging threats, even when historical data is limited.

*f) **Data Collection**:*

- The first step involved selecting and gathering data from the **SDN20** and **UNSW-NB15** datasets. Both datasets include detailed network activity records classified as either normal or malicious. Their comprehensive and diverse nature made them suitable for building a detection system capable of identifying unknown threats.

*g) **Preprocessing**:* Several preprocessing steps were applied to prepare the datasets for modeling:

- Column names were standardized across both datasets to maintain a uniform structure.
- Target labels and column names were adjusted for consistency between SDN20 and NSL-KDD formats.
- Categorical data (such as protocol types and service names) was converted into numerical values to be interpretable by machine learning models.
- Missing or inconsistent data entries were either corrected or removed to prevent errors during training.
- All numerical features were scaled using normalization techniques such as **MinMaxScaler** or **StandardScaler** to bring values into a consistent range.
- Class labels were encoded, with normal traffic labeled as **0** and attack traffic labeled as **1**.

*h) **Dimensionality Reduction with PCA**:*

- Once preprocessing was complete, Principal Component Analysis (PCA) was applied to reduce the number of features while preserving the most significant information. This dimensionality reduction step helped speed up the processing and made the modeling task more efficient without losing important patterns in the data.

*i) **2D Transformation using t-SNE**:*

- After PCA, a technique called **t-distributed Stochastic Neighbor Embedding (t-SNE)** was applied to further simplify the data. t-SNE is a non-linear dimensionality reduction method that focuses on preserving local structures in high-dimensional data.
- It converts similarities between data points into joint probabilities and seeks a lower-dimensional representation that maintains these relationships.
- Unlike PCA, which is linear and focuses on global variance, t-SNE captures the neighborhood information, making it ideal for visualizing complex patterns, such as subtle differences between normal and attack traffic.

t-SNE operates through the following steps :

- Start with high-dimensional data.
- Calculate similarities between data points.
- Map the data points into a 2D space.
- A distinct 2D mapping of the dataset.

This conversion made it easier to visually interpret the datasets and helped in clearly separating normal and attack traffic.

*i) **Label Standardization**:*

- Class labels were standardized across both datasets, with 'normal' traffic labeled as **0** and 'attack' traffic labeled as **1**.

*k) **Feature-Level Fusion and Data Combination**:*

- A feature-level fusion approach was used to combine the datasets.
- Only 10% of the normal samples from the SDN20 dataset were selected.
- These selected samples were combined with the entire 2D-transformed NSL-KDD dataset (including both attack and normal samples).
- The resulting training set consisted mainly of NSL-KDD samples, with a small portion of SDN20 normal samples incorporated.

*l) **Feature-Label Split**:*

- The combined dataset was then split into input features (functions) and output labels (targets).

*m) **Model Training**:*

- Machine learning models such as **Decision Trees**, **Random Forests**, and **Support Vector Machines (SVMs)** were trained on the composite dataset.

*n) **Classification**:*

- After visualization and transformation, the refined data was fed into the machine learning models. These models were trained and evaluated using test data to classify whether each sample represented normal behavior or an attack.

*o) **Attack Detection Output**:*

- Based on the model predictions, the system could identify and flag suspicious traffic. This final step enabled the detection of potential zero-day attacks — unknown threats that had not been previously encountered — thereby improving overall network security.

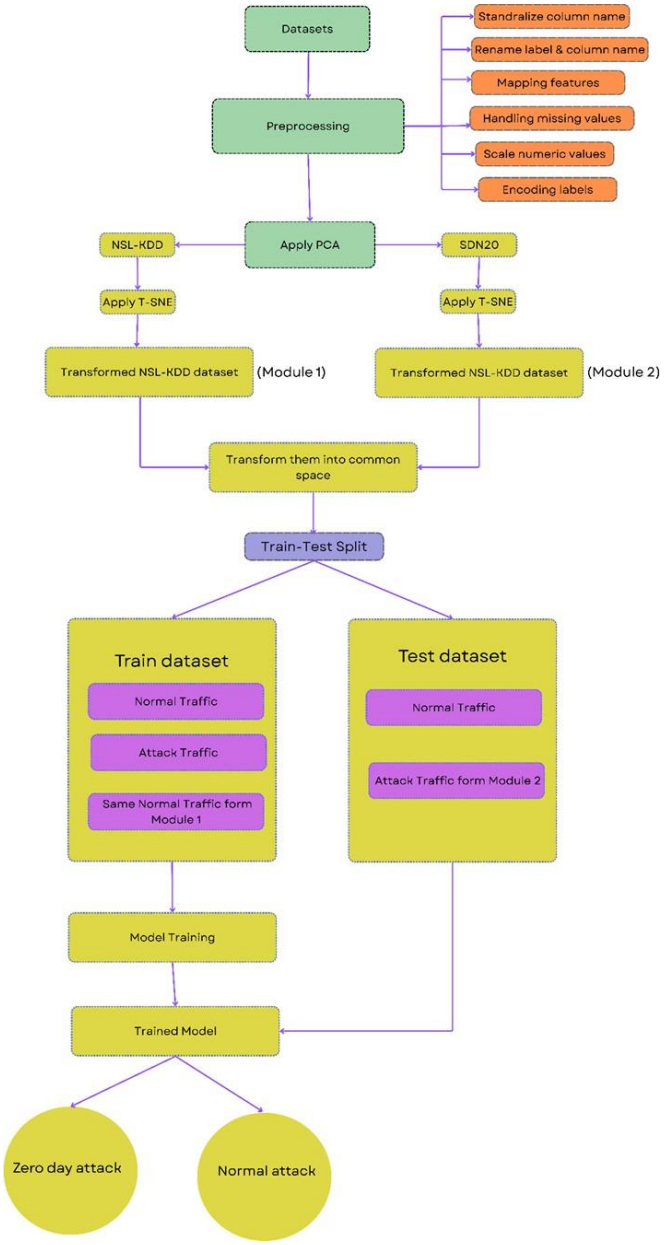


Fig. 5. Objective 2 Model Architecture

V. Experimentation, Results, and Discussion

The experiments were conducted on Google Colab, focusing on the performance evaluation of various machine learning and deep learning models in detecting both known and zero-day attacks. The evaluation utilized three benchmark datasets: **SDN20**, **UNSW-NB15**, and **NSL-KDD**. Analysis was performed under both binary-class and multi-class classification settings during known attack scenarios.

Objective 1 Result: A comparative analysis of accuracy and false positive rate (FPR) was carried out for each model. It was observed that ensemble models, which combine predictions from multiple models, consistently outperformed individual models. The models' performance was assessed using the following evaluation metrics:

Evaluation Metrics

1) **Precision**: Measures the accuracy of positive predictions.

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

2) **Recall**: Reflects the model's ability to correctly identify all positive instances.

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

3) **F1-score**: Represents the harmonic mean of Precision and Recall, providing a balanced measure.

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (3)$$

4) **Accuracy**: Indicates the proportion of correctly predicted instances among the total number of instances.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

where:

- TP = True Positives
- TN = True Negatives
- FP = False Positives
- FN = False Negatives

A. Data Records and Classification Approach

The following datasets were employed for experimentation:

- **SDN20 Dataset**: This dataset focuses on network traffic from Software Defined Networks (SDNs), designed to test and benchmark IDS systems against attacks that are specific to SDN environments.
- **UNSW-NB15 Dataset**: This dataset includes real-world network traffic data, capturing a broad range of attack types, including Denial of Service (DoS), probing, and exploits, providing a challenging benchmark for IDS evaluation.

Each dataset underwent:

- **Binary Classification**: Predicting one of two possible outcomes like normal or attack, offering a simpler approach to identify the presence of an intrusion.

- **Multi-Class Classification**: Predicting one among three or more distinct categories, which allows the model to differentiate between various attack types and normal traffic.

.

Using the SDN20 dataset to classify binary classes:

Algorithm	Accuracy (%)	FPR (%)
Decision Trees (DT)	98.25	2.98
Random Forest	99.44	0.90
ANN	99.22	0.53
Naïve Bayes	76.27	3.88
CNN	99.56	3.94
RNN	98.17	4.12
GRU	99.90	5.03
DT & Random Forest	99.12	3.14
GRU & CNN	98.20	5.17

TABLE I

Performance of Various Algorithms for SDN20 Binary Classification

Using the SDN20 dataset, multi-class classification:

Algorithm	Accuracy (%)	FPR (%)
Decision Trees (DT)	98.25	2.98
Random Forest	99.44	0.90
ANN	99.22	0.53
Naïve Bayes	76.27	3.88
CNN	99.56	3.94
RNN	98.17	4.12
GRU	99.90	5.03
DT & Random Forest	99.12	3.14
GRU & CNN	98.20	5.17

TABLE II

Performance of Various Algorithms for SDN20 Multi-Class Classification

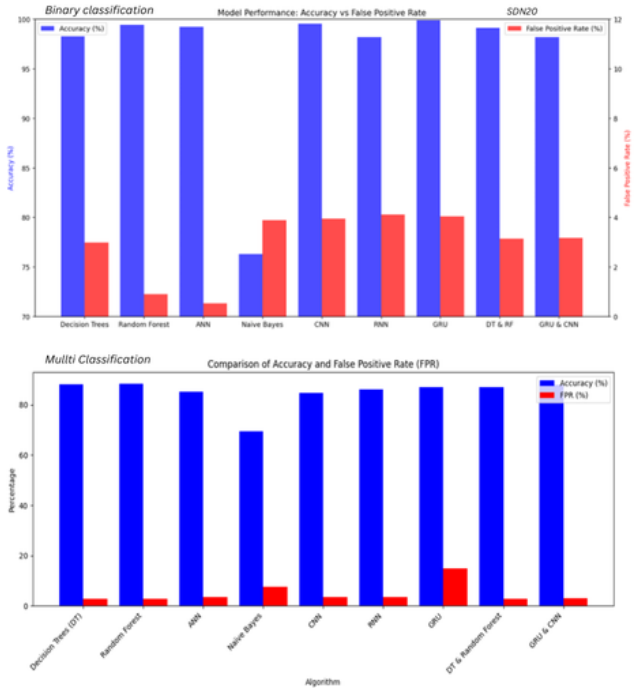


Fig. 6. *Graphical representation of the results for SDN20 Binary and Multi-Class Classification*

Using the UNSW-NB15 dataset, binary-class classification:

Algorithm	Accuracy (%)	FPR (%)
Decision Trees (DT)	97.29	1.64
Random Forest	95.87	2.51
Naïve Bayes	82.21	20.39
CNN	97.34	1.76
RNN	97.09	2.09
GRU	97.19	1.24
ANN	97.31	1.91
CNN-RNN	98.08	1.45
GRU & CNN	97.56	2.26

TABLE III

Performance of Various Algorithms for UNSW-NB15 Binary-Class Classification

Using the UNSW-NB15 dataset, multi-class classification:

Algorithm	Accuracy (%)	FPR (%)
Decision Tree	73.32	2.96
Random Forest	74.25	2.86
KNN	94.98	0.56
Naive Bayes	48.41	5.73
CNN	74.05	2.88
GRU	72.28	3.06
RNN	69.97	3.33

TABLE IV

Performance of Various Algorithms for UNSW-NB15 Multi-Class Classification

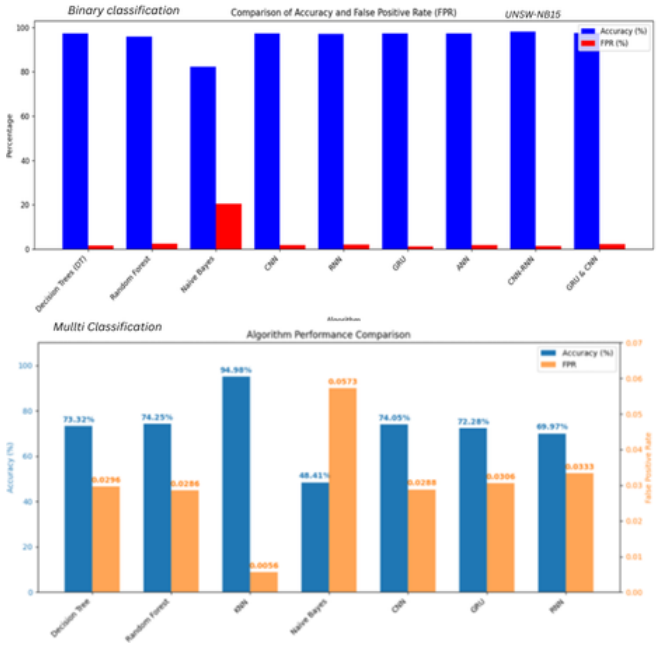


Fig. 7. *Graphical representation of the results for UNSW-NB15 Binary and Multi-Class Classification*

This study presents a detailed comparison of intrusion detection performance using two prominent.cybersecurity datasets—SDN20 and UNSW-NB15. Both datasets were evaluated through binary and multi-class classification approaches to better understand how well machine learning models could identify and categorize network attacks. The analysis begins

with the SDN20 dataset. First, using binary classification, between normal and attack traffic. The results were shared through both tables and visual graphs, helping to clearly show how accurately the models could detect threats. Following this, the study explored multi-class classification on the same dataset, breaking down the attacks into different categories. This provided a more in-depth view of how the models performed when facing more complex and varied intrusion types. Throughout the analysis, visual tools like graphs and tabular data played a key role in highlighting Performance metrics such as accuracy, recall, F1-score. Those metrics have a comprehensive understanding of the advantages and disadvantages of individual models, especially when used in different types of attacks. The paper then shifts focus to the UNSW-NB15 dataset, applying the same binary and multi-class classification techniques. Just like with the SDN20 dataset, the results were presented visually and numerically, allowing for an easy comparison of model effectiveness across datasets. One minor issue was identified during this process: a graph showing results for the UNSW-NB15 dataset was mistakenly labeled as SDN20. While this did not affect the findings, it serves as a reminder of how important accuracy in documentation and labeling can be—especially when presenting technical results. In conclusion, the study not only compares the capabilities of the two datasets but also offers practical recommendations on which classification methods work best for different intrusion detection tasks. These insights could be particularly valuable for researchers and cybersecurity professionals looking Improved detection of known and new threats.

Objective 2 Results:

Model	Accuracy (%)	Total	Correct	Missed	FPR (%)
DT	82.1	5796	4528	1268	15.88
RF	82.2	5796	4492	1304	15.52
SVM	42.1	5796	2564	3232	88.87
DT + RF	82.1	5796	4531	1265	15.95

TABLE V
Tabular representation of zero-day attack detection results.

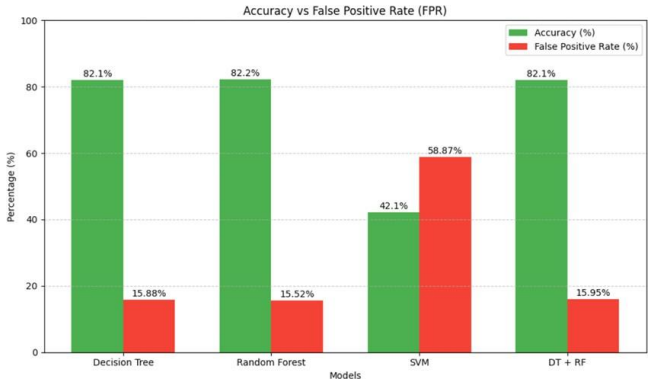


Fig. 8. *Graphical representation of results of accuracies and false positive rate*

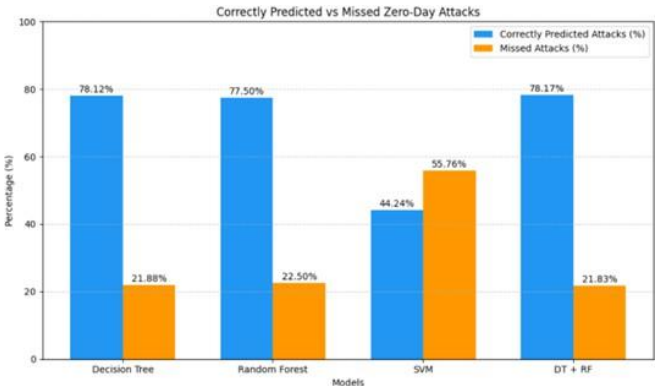


Fig. 9. *Graphical representation of correctly predicted vs missed zero-day attacks*

This study presents a performance evaluation of several A Machine Learning model such includes decision trees, random forests, Support Vector Machines (SVMs), and hybrid models that combine decision trees and random forests. Important Issues with Intrusion Detection Systems (IDS) task was reference to zero-day attack detection. Furthermore, increasing the number of attack samples did not result in improved model performance. Among the evaluated models, the Random Forest The classifier showed the highest overall accuracy and achieved 82.2%. It was able to detect 77.50% of attack instances, with a false positive rate limited to 15.52%. While both the Decision Tree and the combined Decision Tree + Random Forest models yielded similar accuracy scores (82.1%), the hybrid model slightly outperformed in terms of attack detection, achieving 78.17 % detection rate and false positive rate. 15.9 %. In contrast,Support Vector Machine (SVM) showed a significant performance drop in attack detection rate of only 42.1% and a high False positive Rate of 58.87%. Consequently, SVM failed to detect over half (55.76%) of the attacks, rendering it less suitable for intrusion detection in this context.

VI. Conclusion

This research advances Intrusion Detection Systems (IDS) by developing a dual-objective framework to detect both known and zero-day cyberattacks using Machine Learning (ML), Deep Learning (DL), and fusion-based methodologies. The first objective evaluated ML and DL models on UNSW-NB15 and SDN20 datasets, revealing that while Decision Trees and Random Forests performed reliably, DL models like CNNs, RNNs, and GRUs excelled in accuracy, precision, and recall, especially in complex multi-class classification tasks. The second, more critical objective tackled zero-day attack detection through a fusion-based IDS, integrating NSL-KDD for training and SDN20 for testing. By employing PCA and t-SNE for dimensionality reduction and feature-level dataset fusion, the system effectively identified unseen threats. Random Forest emerged as the standout model, achieving superior accuracy and detection rates with a low false positive rate. This work demonstrates that combining diverse datasets with

ensemble and DL techniques creates adaptive, intelligent IDS frameworks capable of addressing evolving cyber threats. In real-time applications, this solution strengthens cybersecurity in dynamic environments like Software-Defined Networks and IoT systems, enabling rapid and reliable threat detection. By offering a scalable approach to zero-day attack mitigation, this research significantly contributes to the field, providing a foundation for developing next-generation IDS that can proactively counter sophisticated cyberattacks and enhance network security.

REFERENCES

[1] Jun Zhang, Lei Pan, Qing-Long Han, Chao Chen, Sheng Wen, and Yang Xiang, "Deep Learning Based Attack Detection for Cyber-Physical System Cybersecurity: A Survey," *IEEE Transactions on Industrial Informatics*, vol. XX, no. XX, pp. 1–XX, 2020.

[2] Ahmed H. Janabi, Mark Johnson, and Triantafyllos Kanakis, "Survey: Intrusion Detection System in Software-Defined Networking," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[3] Nura Shifa Musa, Amira Mahamat Abdallah, Nada Masood Mirza, Saida Hafsa Rafique, and Thangavel Murugan, "Machine Learning and Deep Learning Techniques for Distributed Denial of Service Anomaly Detection in Software Defined Networks—Current Research Solutions," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[4] Hanadi Hakami, Muhammad Faheem, and Majid Bashir Ahmad, "Machine Learning Techniques for Enhanced Intrusion Detection in IoT Security," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[5] Tristan Bilot, Nour El Madhoun, Khaldoun Al Agha, and Anis Zouaoui, "Graph Neural Networks for Intrusion Detection: A Survey," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[6] Zahedi Azam, Md. Motaharul Islam, and Mohammad Nurul Huda, "Comparative Analysis of Intrusion Detection Systems and Machine Learning-Based Model Analysis Through Decision Tree," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[7] Dheyaaldin Alsalman, "A Comparative Study of Anomaly Detection Techniques for IoT Security Using Adaptive Machine Learning for IoT Threats," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[8] Alejandro Dom´ınguez Campos, Jose´-Luis Gonza´lez-Sa´nchez, Felipe Lemus-Prieto, and Andre´s Carolindo, "Intrusion Detection for IoT Environments Through Side-Channel and Machine Learning Techniques," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[9] Asmaa Halbouni, Teddy Surya Gunawan, Mohamed Hadi Habaebi, Murad Halbouni, Mira Kartiwi, and Robiah Ahmad, "CNN-LSTM: Hybrid Deep Neural Network for Network Intrusion Detection System," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[10] Saqib Ali, Mokhtar Mohammadi, Jan Lansky, Sarkhel H. Taher Karim, Shima Rashidi, Mehdi Hosseinzadeh, Mohammed Kamal Majeed, and Amir Masoud Rahmani, "Deep Learning-Based Intrusion Detection Systems: A Systematic Review," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[11] William Villegas-Ch, Rommel Gutierrez, Jaime Govea, Alexandra Maldonado Navarro, and Aracely Mera-Navarrete, "Effectiveness of an Adaptive Deep Learning-Based Intrusion Detection System," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[12] Jiawei Du, Kai Yang, Yanjing Hu, and Lingjie Jiang, "NIDS-CNNLSTM: Network Intrusion Detection Classification Model Based on Deep Learning," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[13] Seyed Mohammadhadi Mirsadeghi, Risto Vaarandi, Hayretdin Bahsi, and Wisse Minoubi, "Learning From Few Cyber-Attacks: Addressing the Class Imbalance Problem in Machine Learning-Based Intrusion Detection in Software-Defined Networking," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[14] Jitendra Kumar Samriya, Surendra Kumar, Huaming Wu, Mohit Kumar, and Sukhpal Singh Gil, "Machine Learning-Based Network Intrusion Detection Optimization for Cloud Computing Environments," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[15] Han Liao, Mohd Zamri Murah, Mohammad Kamrul Hasan, Azana Hafizah Mohd Aman, Jin Fang, Xuting Hu, and Atta Ur Rehman Khan, "A Survey of Deep Learning Technologies for Intrusion Detection in Internet of Things," *IEEE Access*, vol. XX, no. XX, pp. XX–XX, 2020.

[16] Sruthi C.K., Aswathy Ravikumar, and Harini Sriraman, "Enhancing Zero-Day Attack Detection with XAI-Driven ML Models and SMOTE Analysis," in *Proceedings of the 2024 3rd IEEE International Conference on Artificial Intelligence for Internet of Things (AIIoT)*, pp. XX–XX, 2024.

[17] Sowmiya R., R. A. J. Kumar, and Arputharaj K., "Zero-Day Attack Detection Using Autoencoder Based Feature-Level Fusion of SDN20 and NSL-KDD Datasets," in *Proceedings of the 2024 3rd IEEE International Conference on Artificial Intelligence for Internet of Things (AIIoT)*, pp. XX–XX, 2024.

[18] Huthifh Al-Rushdan, Mohammad Shurman, Sharhabeel H. Alnabelsi, and Qutaibah Althebyan, "Zero-Day Attack Detection and Prevention in Software-Defined Networks," *Journal/Conference Name*, vol. XX, no. XX, pp. XX–XX, 2024.

[19] Harini Sriraman, Aswathy Ravikumar, and Sruthi C.K., "Deep Learning-Based Intrusion Detection: A Comparative Study of CNN, RNN, and Hybrid Models for Zero-Day Threat Detection," in *Proceedings of the 2024 3rd IEEE International Conference on Artificial Intelligence for Internet of Things (AIIoT)*, pp. XX–XX, 2024.

[20] Sruthi C.K., Harini Sriraman, and Aswathy Ravikumar, "GRU-CNN Based Zero-Day Attack Detection on SDN20 and NSL-KDD Datasets," in *Proceedings of the 2024 3rd IEEE International Conference on Artificial Intelligence for Internet of Things (AIIoT)*, pp. XX–XX, 2024.