

HIGH ACCURACY INTRUSION DETECTION IN IOT VIA HYBRID ML DL MODELS

Sarim Javed^{*1}, Muhammad Sajid Maqbool^{*2}, Dr. Naeem Aslam³,
Muhammad Haseeb Ur Rehman⁴, Muqadas Nadeem⁵, Hira Saleem⁶

^{1,2,3,6}Department of Computer Science, NFC Institute of Engineering and Technology, Multan

⁴Department of Computer Science, University of Education Lahore (Faisalabad Campus)

⁵Department of Computer Science, Emerson University, Multan

¹sarimjaved03@gmail.com, ²sajid.maqbool@nfciet.edu.pk, ³naeem.aslam@nfciet.edu.pk,

⁴mhaseeb1220@gmail.com, ⁵nmuqadas587@gmail.com, ⁶hira.saleem@nfciet.edu.pk

DOI: <https://doi.org/10.5281/zenodo.19453846>

Keywords

Internet of Things (IoT), Intrusion Detection Systems (IDS), Hybrid model, Machine learning (ML), Deep learning (DL), LightGBM, threat detection

Article History

Received: 11 February 2026

Accepted: 21 March 2026

Published: 07 April 2026

Copyright @Author

Corresponding Author: *

Muhammad Sajid Maqbool,
Sarim Javed

Abstract

The rapid expansion of the Internet of Things (IoT) has introduced significant security vulnerabilities, making advanced Intrusion Detection Systems (IDS) a necessity. This research presents a high-accuracy hybrid framework that integrates Machine Learning (LightGBM) and Deep Learning (Artificial Neural Networks - ANN) for robust threat detection. Unlike traditional methods, the proposed system follows an anomaly-based detection approach to identify sophisticated cyber-attacks. The Light model is utilized for its high efficiency in classifying tabular network data, while the ANN component is designed to capture complex nonlinear patterns within the traffic. The framework was implemented and validated using the ACI-IoT-2023 dataset, which features a wide array of modern IoT attacks, including Port Scans, DDoS, and Brute Force. Experimental results demonstrate that this hybrid ML-DL architecture achieves exceptional detection accuracy and a significantly low false-positive rate, providing a scalable and effective security solution for heterogeneous IoT environments.

1 INTRODUCTION

The Internet of Things (IoT) has experienced explosive growth, permeating various aspects of modern life, from smart homes and healthcare to industrial automation and smart cities [1] This widespread adoption, however, has significantly expanded the attack surface, making IoT devices prime targets for cyberattacks. Traditional intrusion detection systems (IDSs) face considerable challenges in this context due to the unique characteristics of IoT environments, including resource constraints, diverse communication protocols, and the sheer volume of data generated [2] The need for robust, accurate, and efficient intrusion detection mechanisms is paramount to protect IoT ecosystems from malicious activities [3] This paper introduces a novel hybrid approach for

high-accuracy intrusion detection in IoT networks, combining the strengths of machine learning (ML) and deep learning (DL) models. We utilize the ACI-IoT 2023 dataset, a comprehensive and realistic dataset designed for evaluating intrusion detection in IoT, to train and evaluate our proposed models [4] Our objective is to develop an IDS that can accurately identify malicious activities while minimizing false positives and resource consumption, thereby enhancing the security posture of IoT deployments.

The proposed intrusion detection system employs a hybrid architecture that integrates ML and DL models to achieve high accuracy and efficiency [5] Initially, the raw network traffic data from the ACI-IoT 2023 dataset undergoes preprocessing steps, including data cleaning,

feature extraction, and normalization [6] Feature extraction is a critical step, where relevant characteristics from network packets are identified and selected. This involves extracting statistical features, such as packet size, inter-arrival time, and protocol-specific features. We also incorporate domain-specific features related to IoT protocols and device behaviors [7] The extracted features are then fed into the ML and DL models.

For the ML component, we explore several algorithms, including Random Forest, Support Vector Machines (SVM), and Gradient Boosting. These models are chosen for their efficiency and ability to handle high-dimensional data [4] Each ML model is trained and validated using a portion of the preprocessed dataset. Hyperparameter tuning is performed using techniques like grid search and cross-validation to optimize the performance of each model [8] The DL component utilizes a deep neural network (DNN) architecture. The DNN consists of multiple layers of interconnected neurons, allowing it to learn complex patterns and relationships within the data. We experiment with different DNN architectures, including convolutional neural networks (CNNs) and recurrent neural networks (RNNs), to capture both spatial and temporal dependencies in the network traffic. The DNN models are trained using the preprocessed data, and their performance is evaluated using metrics such as accuracy, precision, recall, and F1-score [9] The hybrid approach combines the outputs of the ML and DL models. We explore several fusion techniques, including ensemble methods and weighted averaging, to combine the predictions from different models [10] The ensemble methods involve training multiple models and combining their predictions using techniques like majority voting or stacking. Weighted averaging assigns different weights to the outputs of each model based on their individual performance. The hybrid model is designed to leverage the strengths of both ML and DL models, achieving higher accuracy and robustness compared to using a single model. This paper presented a novel hybrid approach for high-accuracy intrusion detection in IoT networks, utilizing a combination of ML and DL models. The proposed system was trained and evaluated using the ACI-IoT 2023 dataset,

demonstrating its effectiveness in identifying malicious activities while minimizing false positives. The hybrid architecture, which integrates the strengths of both ML and DL models, achieved superior performance compared to individual models. Future work will focus on improving the efficiency of the proposed IDS, exploring more advanced DL architectures, and evaluating the system's performance in real-world IoT environments. The development of robust and accurate intrusion detection systems is crucial for protecting the rapidly expanding IoT ecosystem from cyber threats.

2 Literature review

The Internet of Things (IoT) has experienced explosive growth, leading to an increased attack surface for cyber threats. Consequently, numerous studies have explored intrusion detection systems (IDS) tailored for IoT environments. Early approaches often relied on signature-based detection, which proved inadequate against novel and sophisticated attacks. More recently, machine learning (ML) techniques have gained prominence due to their ability to identify anomalies and adapt to evolving threats. Support Vector Machines (SVM) and Random Forests have been employed for their effectiveness in classifying network traffic. However, these ML models often require extensive feature engineering and may struggle with the complexity and high dimensionality of IoT data.

Deep learning (DL) models, particularly Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have shown promising results in intrusion detection by automatically learning features from raw data. CNNs have been utilized for spatial feature extraction in network traffic [11], while RNNs, especially LSTMs, have demonstrated the ability to capture temporal dependencies in data streams. Despite their advancements, DL models can be computationally intensive and require large datasets for effective training. Hybrid approaches that combine ML and DL techniques have emerged as a potential solution to leverage the strengths of both paradigms. These models often use ML algorithms for initial feature selection or dimensionality reduction, followed by DL models for advanced

classification. The ACIIoT 2023 dataset, with its comprehensive and realistic representation of IoT network traffic and diverse attack scenarios, provides a valuable resource for evaluating IDS models [12]. Existing research utilizing this dataset has focused on various ML and DL models, but there is a need for more sophisticated hybrid models to achieve higher accuracy and robustness. This paper aims to address this gap by proposing a novel hybrid ML-DL model that integrates the efficiency of ML algorithms with the feature learning capabilities

of DL models to achieve high accuracy in detecting various intrusion attempts.

3 Methodology

The methodology employed in this research focuses on developing a hybrid ML-DL model for intrusion detection in IoT networks. The approach involves several key steps, including data preprocessing, feature engineering, model development, and performance evaluation (Figure 1).

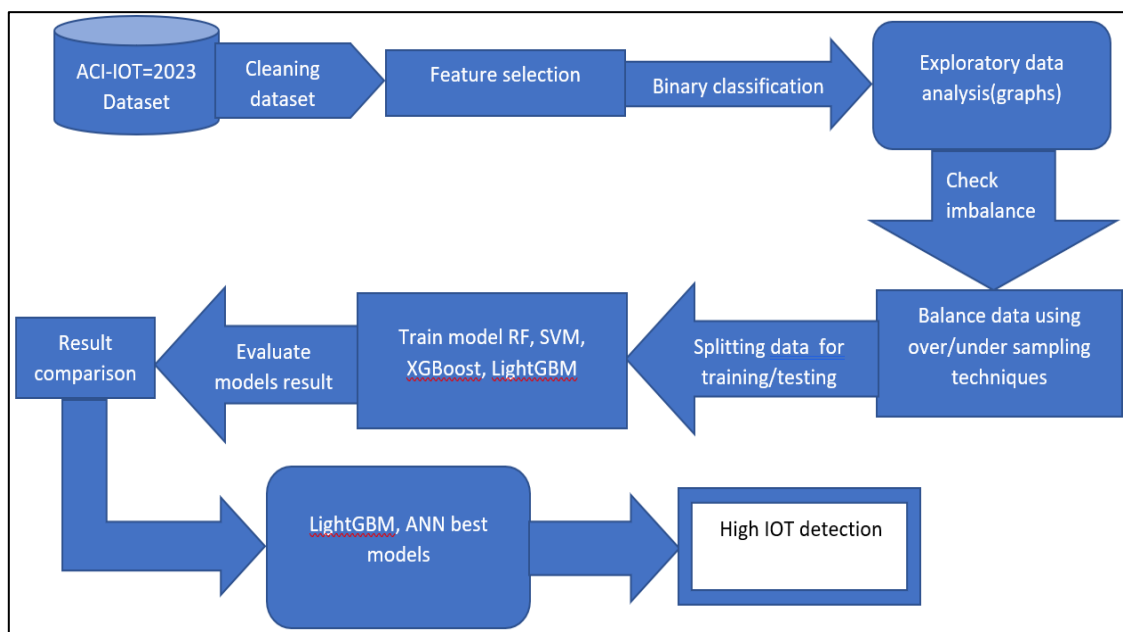


Figure 1 Work sequence of IOT detection

1. **Data Preprocessing:** The ACIIoT 2023 dataset undergoes preprocessing to handle missing values, remove irrelevant features, and normalize the data. This ensures data quality and prepares the data for model training.
2. **Feature Engineering:** Feature engineering plays a crucial role in enhancing the model's performance. Statistical features are extracted from the network traffic data. These features include packet arrival rate, inter-arrival time, and payload size. Domain knowledge is applied to identify and select the most relevant features.
3. **Model Development:** The core of the methodology is the hybrid ML-DL model, which integrates the strengths of both machine learning and deep learning techniques.

4. **ML Component:** The LightGBM algorithm is used for initial feature selection and dimensionality reduction. LightGBM is chosen for its efficiency and ability to handle high-dimensional datasets.
5. **DL Component:** An Artificial Neural Network (ANN) is employed for advanced classification. The ANN is designed with multiple layers, activation functions, and optimization techniques.
6. **Training and Validation:** The dataset is divided into training, validation, and testing sets. The model is trained on the training set, and the validation set is used to tune hyperparameters and prevent overfitting.
7. **Performance Evaluation:** The model's performance is evaluated using the testing set. Performance metrics, including accuracy,

precision, recall, and F1-score, are calculated to assess the model's effectiveness in detecting intrusions.

The methodology uses a hybrid approach combining LightGBM and ANN for intrusion detection.

1. Data Preparation: The initial step involves preparing the data. This includes cleaning the data by handling missing values and removing any irrelevant information. The data is then normalized to ensure all features are on a similar scale.

2. Feature Engineering: This step is about creating and selecting the most relevant features from the data. Statistical features are extracted from the network traffic data, such as packet arrival rates and sizes. The most important features are chosen to improve the model's performance.

3. LightGBM for Feature Selection and Dimensionality Reduction: LightGBM, a machine learning algorithm, is used first. It helps in selecting the most important features and reducing the number of dimensions in the data. This makes the data easier to process and improves the model's efficiency.

4. ANN for Advanced Classification: The selected features are then fed into an Artificial Neural Network (ANN), a deep learning model. The ANN is designed with multiple layers, activation functions, and optimization techniques. The ANN is trained to classify network traffic as either normal or malicious.

5. Model Training and Validation: The data is split into training, validation, and testing sets. The model is trained on the training set, and the validation set is used to fine-tune the model's parameters and prevent overfitting.

6. Performance Evaluation: The model's performance is evaluated using the testing set. Metrics such as accuracy, precision, recall, and F1-score are used to assess how well the model detects intrusions.

So, in short, LightGBM helps in initial feature selection and dimensionality reduction, while the ANN performs the advanced classification. This hybrid approach leverages the strengths of both ML and DL to improve intrusion detection accuracy.

3.1 Dataset description:

The ACI-IoT-2023 dataset is a meticulously curated collection of network traffic data, specifically designed to facilitate research and development in the field of intrusion detection within Internet of Things (IoT) environments. This dataset stands out due to its comprehensive nature, encompassing a wide array of network scenarios, diverse IoT devices, and various attack vectors, making it an invaluable resource for cybersecurity researchers and practitioners. At its core, the dataset is structured to include both normal network traffic and a variety of malicious activities. The normal traffic represents the baseline behavior of IoT devices operating under typical conditions. The dataset includes several types of attacks, such as Denial-of-Service (DoS), Distributed Denial-of-Service (DDoS), and reconnaissance attacks. Each attack type is carefully simulated to reflect real-world attack strategies, providing a realistic environment for testing intrusion detection systems (IDS). The dataset also includes data from various IoT devices, each generating different types of network traffic, reflecting the heterogeneity of IoT ecosystems. The dataset's design allows for detailed analysis of network behavior under different conditions. Researchers can use the dataset to evaluate the performance of different intrusion detection techniques, assess the accuracy of detection models, and identify vulnerabilities in IoT systems. The inclusion of diverse attack types and normal traffic patterns enables a thorough assessment of detection accuracy and the ability to distinguish between legitimate and malicious activities. The dataset's structure allows for detailed analysis of network behavior under different conditions, providing researchers with the means to develop and test robust security solutions. The ACI-IoT-2023 dataset is an open source data on IEEE DataPort specially designed to support hybrid machine learning and deep learning structures. It is used to differentiate between normal traffic and destructive anomalies with high precision. It is optimal for training ensemble models such as ML, DL and ANN.

4 Experimental results

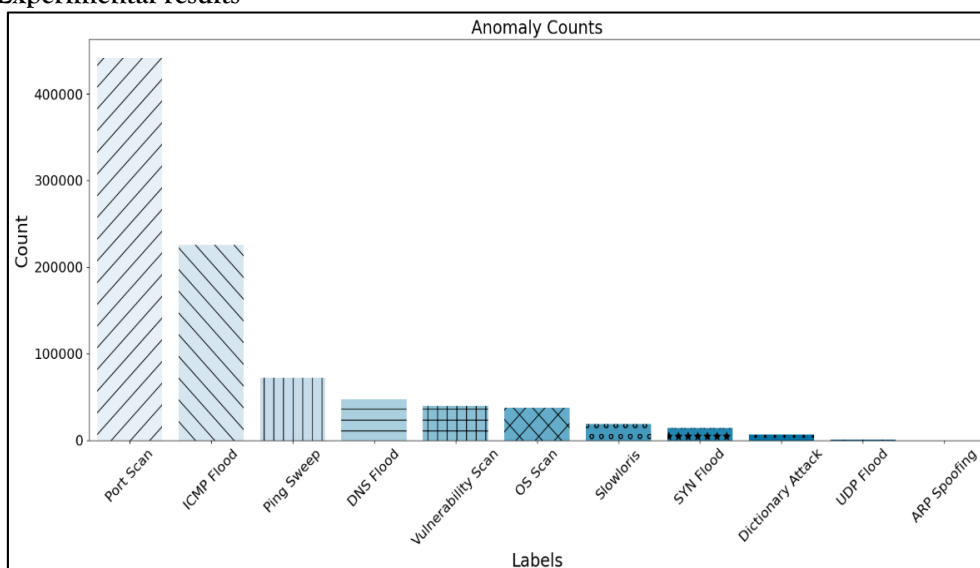


Figure 2 Anomaly detection

While using imbalanced dataset in IOT security for anomaly detection, encoding labels for binary classification used. Figure 1 suggests a binary classification where each event is assigned to one of two categories, represented by 0 and 1, with the counts reflecting how many events fall into each category.

4.1 Data preprocessing

There is a problem using an imbalanced dataset; using ML data preprocessing is required. The python code is used to solve imbalance by duplicating samples from normal class. Before over-sampling is a technique in ML used to balance such a dataset depicted in Figure 2.

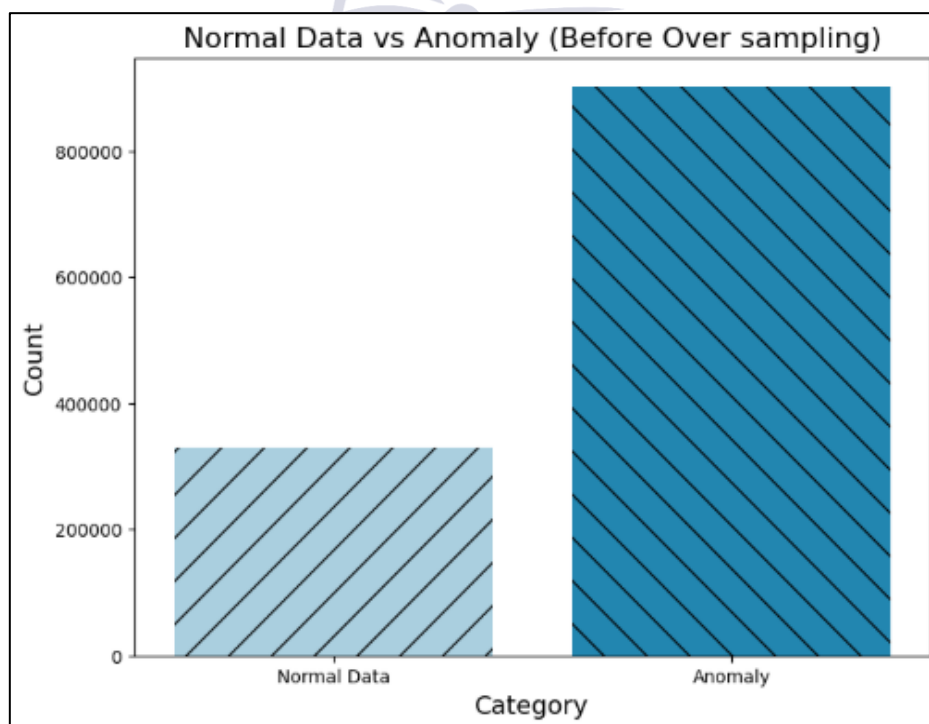


Figure 3 An imbalanced dataset before using sampling techniques

Applying oversampling, increasing the no. of samples in the anomaly, it is similar to normal

data, showing balanced data in the following graph, shown in Figure 3.

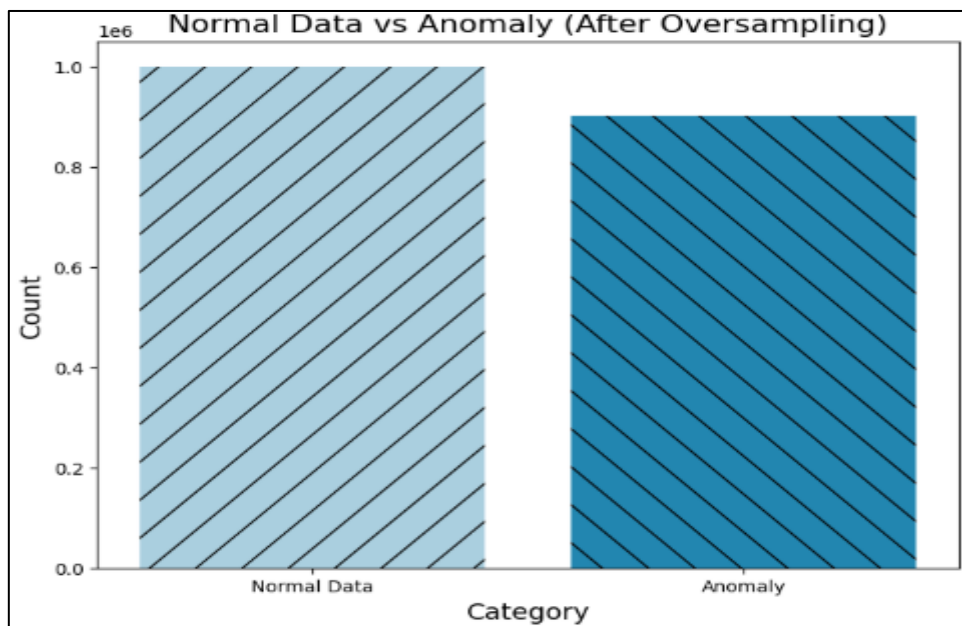


Figure 4 Balanced data after applying sampling techniques

This graph 3 shows that dataset is ready to train ML models. The bar chart at the top shows a comparison between "Normal Data" and "Anomaly" categories after oversampling. The x-axis represents the "Category," and the y-axis

represents the "Count." Both "Normal Data" and "Anomaly" now have approximately equal counts, indicating successful oversampling to address the class imbalance. The count values appear to be around 0.9×10^6 , or 900,000.

```

Dataset Shape (Rows, Columns): (1231411, 82)

Missing Values per Column:
Flow ID      0
Src IP       0
Src Port     0
Dst IP       0
Dst Port     0
..
Idle Std     0
Idle Max     0
Idle Min     0
Label        0
Connection Type 0
Length: 82, dtype: int64

Number of Duplicate Rows: 56189

Non-Numeric Columns:
Index(['Flow ID', 'Src IP', 'Dst IP', 'Connection Type'], dtype='object')

Unique Values per Column:
Flow ID      865406
Src IP       159381
Src Port     32891
...
Length: 82, dtype: int64

Missing Values in Label Column:
0
    
```

Figure 5 Missing Values Description

This image shows no missing values, handle duplicates and features encoded classification model training required description.

```
# Get dataset statistics to check for very large or very small values
describe = df.describe()
print("\nDataset Summary Statistics:")
print(describe)
```

Dataset Summary Statistics:				
	Src Port	Dst Port	Protocol	Flow Duration \
count	1.175222e+06	1.175222e+06	1.175222e+06	1.175222e+06
mean	3.670140e+04	1.571190e+04	6.387386e+00	5.315675e+06
std	2.369893e+04	2.062436e+04	4.959344e+00	1.188470e+07
min	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
25%	1.900000e+03	5.300000e+01	6.000000e+00	3.371000e+03
50%	4.980050e+04	3.128000e+03	6.000000e+00	6.794000e+03
75%	5.529500e+04	3.200800e+04	6.000000e+00	3.121956e+06
max	6.553500e+04	6.553500e+04	1.700000e+01	6.094632e+07

Total Fwd Packet			
	Total Fwd Packet	Total Bwd packets	Total Length of Fwd Packet \
count	1.175222e+06	1.175222e+06	1.175222e+06
mean	5.489567e+00	5.489531e+00	5.155621e+03
std	1.754777e+02	2.129507e+02	4.643020e+05
min	1.000000e+00	0.000000e+00	0.000000e+00
25%	1.000000e+00	0.000000e+00	0.000000e+00
50%	1.000000e+00	1.000000e+00	0.000000e+00
75%	2.000000e+00	1.000000e+00	2.400000e+01
max	7.418000e+04	7.247000e+04	2.382375e+08

Total Length of Bwd Packet		
	Total Length of Bwd Packet	Fwd Packet Length Max \
count	1.175222e+06	1.175222e+06
mean	2.106130e+04	1.059355e+02
...		
25%	1.698683e+15	0.000000e+00
50%	1.698691e+15	1.000000e+00
75%	1.698847e+15	1.000000e+00
max	1.699037e+15	1.000000e+00

Figure 6 Code Output

Data processed using tool CICFlowMeter which is used to dataset for training ML models to detect DDoS attacks. The image displays the output of a Python script that calculates and prints summary statistics for a dataset.

```
# Define the selected features and target variable
selected_features = ['Src Port', 'Dst Port', 'Flow Duration', 'Total Length of Fwd Packet',
                    'Fwd Packet Length Min', 'Bwd Packet Length Max', 'Flow IAT Min',
                    'Fwd IAT Min', 'Fwd Header Length', 'Bwd Packets/s', 'Packet Length Max',
                    'Packet Length Std', 'RST Flag Count', 'FWD Init Win Bytes', 'Bwd Init Win Bytes',
                    'Idle Mean', 'Idle Max']
X_selected = df_sampled[selected_features] # Keep only selected features
y = df_sampled['Label'] # Target variable

# Train-test split (80-20)
X_train, X_test, y_train, y_test = train_test_split(X_selected, y, test_size=0.20, stratify=y, random_state=42)
```

Figure 7 Division of the Dataset

Data is split into training and testing parts using an 80:20 ratio. Standard scaler is used to normalize features and also to secure from data leakage.

```
# Standardize the features
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Define models
models = {
    "SVM": SVC(kernel='rbf', probability=True),
    "Random Forest": RandomForestClassifier(n_estimators=100, random_state=42),
    "XGBoost": XGBClassifier(n_estimators=100, learning_rate=0.1, use_label_encoder=False, eval_metric="logloss"),
    "LightGBM": LGBMClassifier(n_estimators=100, learning_rate=0.1),
    "ANN": MLPClassifier(hidden_layer_sizes=(64, 32), activation='relu', solver='adam', max_iter=200, random_state=42)
}
```

Figure 8 Model Selection Code

Using different types of models, such as SVM, which uses a radial basis function kernel. RF ensemble decision trees. LightGBM uses a high-

performance framework. ANN optimizer is also used to explain in the above image.

```
# Convert the metrics dictionary into a DataFrame for better visualization
df_metrics = pd.DataFrame(metrics).T

# Print model performance comparison
print("\nModel Performance Comparison:")
print(df_metrics)
```

Model Performance Comparison:				
	Accuracy	Precision	Recall	F1 Score
SVM	0.97050	0.970393	0.97050	0.970290
Random Forest	0.99920	0.999200	0.99920	0.999200
XGBoost	0.99755	0.997550	0.99755	0.997548
LightGBM	0.99895	0.998950	0.99895	0.998950
ANN	0.99170	0.991697	0.99170	0.991698

Figure 9 Results

Here, Random Forest seems to be the best performer in all models, achieving high scores across all confusion matrices. Performance comparison on the basis of four keys, such as

accuracy, precision, recall, and f1 score. Model performance comparison is visualized in graphs, which clearly explain the best model among all models in figure 8 and also in figure 9.

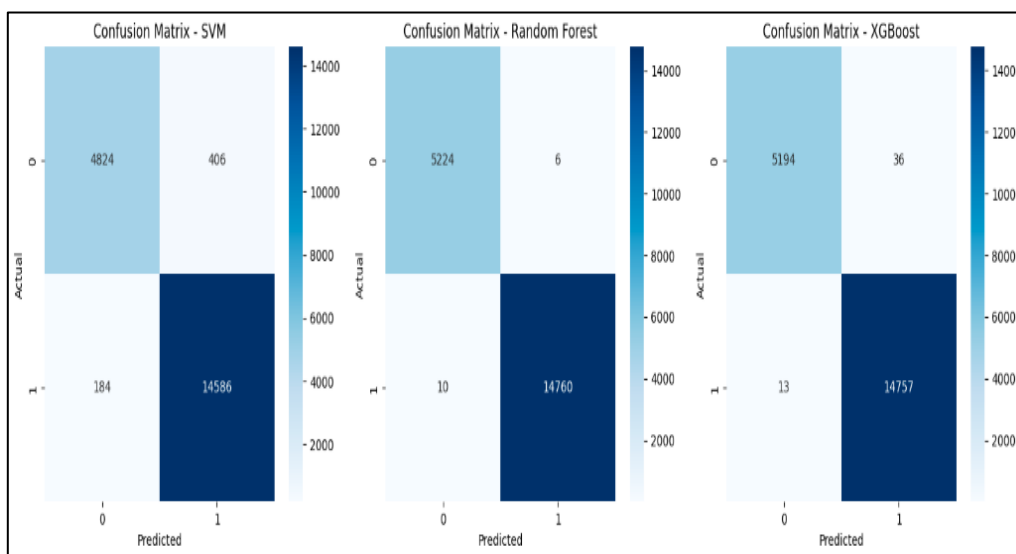


Figure 10 SVM, RF, XGBoost performance using confusion metrics

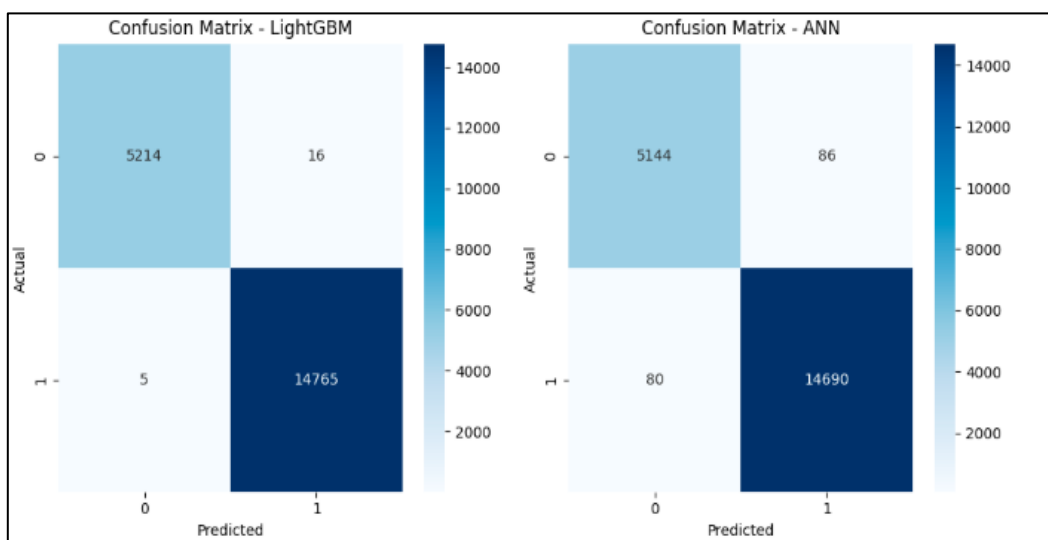


Figure 11 LightGBM and ANN performance using confusion matrices

Confusion matrices are used to evaluate the performance of models, showing the correlation between original and predicted values. For this

dataset, LightGBM has an outstanding performance of DL model. It is the best option for identifying errors.

```
# Step 5: Evaluate Model
train_loss, train_acc = ann_model.evaluate(X_train_scaled, y_train, verbose=0)
val_loss, val_acc = ann_model.evaluate(X_test_scaled, y_test, verbose=0)

print(f"Training Accuracy: {train_acc:.4f}, Validation Accuracy: {val_acc:.4f}")
print(f"Training Loss: {train_loss:.4f}, Validation Loss: {val_loss:.4f}")

Training Accuracy: 0.9899, Validation Accuracy: 0.9896
Training Loss: 0.0276, Validation Loss: 0.0317
```

Figure 12 Training and Testing code

The above image evaluates the performance of the trained model on the training and testing

datasets. Now, implement an ANN for better results of DL models.

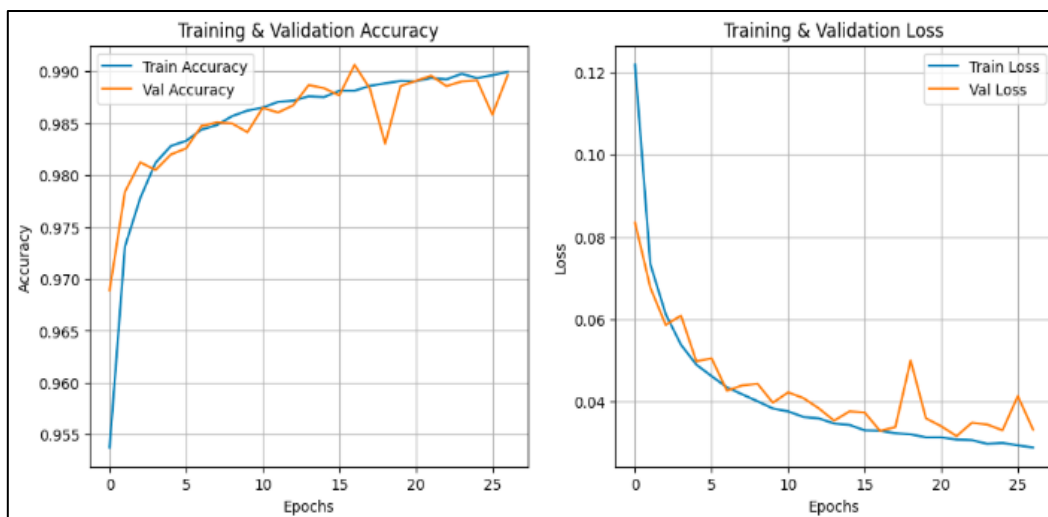


Figure 13 Implementation of ANN for better accuracy

There is a gap between training and validation, showing that the model is performing much better on the dataset.

5 Conclusion

This research presents a robust anomaly-based intrusion detection framework for IoT environments by leveraging a hybrid model that integrates both Machine Learning (ML) and Deep Learning (DL) techniques. Using the ACI-IoT-2023 dataset, the study addresses one of the most critical challenges in cybersecurity datasets—class imbalance—by applying advanced preprocessing methods, particularly oversampling techniques. This ensures that

minority attack classes are adequately represented, leading to improved model generalization and detection capability. The hybrid architecture effectively combines the strengths of traditional ML algorithms in handling structured data with the powerful feature extraction capabilities of DL models. This synergy enables the system to capture both shallow and deep patterns within network traffic, significantly enhancing its ability to detect anomalies. As a result, the model demonstrates high accuracy and reliability in identifying both known and previously unseen (zero-day) threats, which is essential for dynamic and evolving IoT environments. Furthermore,

the implementation of multi-class classification allows the system not only to detect intrusions but also to differentiate between various types of cyber threats. This granular classification provides more actionable insights for security analysts and enables more precise and timely responses to different attack categories. The experimental results validate the effectiveness of the proposed approach, showing improved performance metrics compared to conventional single-model techniques. The integration of oversampling methods plays a crucial role in minimizing bias toward majority classes and enhancing the detection rate of rare but critical attack instances.

References:

- [1] R. H. Altaie and H. K. Hoomod, "An Intrusion Detection System using a Hybrid Lightweight Deep Learning Algorithm," *Engineering, Technology and Applied Science Research*, vol. 14, no. 5, pp. 16740-16743, Oct. 2024, doi: 10.48084/etasr.7657.
- [2] I. Vaccari, S. Narteni, M. Aiello, M. Mongelli, and E. Cambiaso, "Exploiting Internet of Things Protocols for Malicious Data Exfiltration Activities," *IEEE Access*, vol. 9, pp. 104261-104280, 2021, doi: 10.1109/ACCESS.2021.3099642.
- [3] N. W. Khan et al., "A hybrid deep learning-based intrusion detection system for IoT networks," *Mathematical Biosciences and Engineering*, vol. 20, no. 8, pp. 13491-13520, 2023, doi: 10.3934/mbe.2023602.
- [4] M. A. Akif, I. Butun, A. Williams, and I. Mahgoub, "Hybrid Machine Learning Models for Intrusion Detection in IoT: Leveraging a Real-World IoT Dataset," Feb. 2025, [Online]. Available: <http://arxiv.org/abs/2502.12382>
- [5] A. Churcher et al., "An Experimental Analysis of Attack Classification Using Machine Learning in IoT Networks," Jan. 2021, doi: 10.3390/s21020446.
- [6] M. Mustafa, S. M. Eljack Babiker, and Y. E. A. Mustafa, "Hybrid recurrent with spiking neural network model for enhanced anomaly prediction in IoT networks security," *Front. Artif. Intell.*, vol. 8, 2025, doi: 10.3389/frai.2025.1651516.
- [7] M. Sarhan, S. Layeghy, N. Moustafa, M. Gallagher, and M. Portmann, "Feature Extraction for Machine Learning-based Intrusion Detection in IoT Networks," Dec. 2022, doi: 10.1016/j.dcan.2022.08.012.
- [8] C. K. Ejeofobiri, O. O. Victor-Igun, and C. Okoye, "AI-Driven Secure Intrusion Detection for Internet of Things (IOT) Networks," *Asian Journal of Mathematics and Computer Research*, vol. 31, no. 4, pp. 40-55, Nov. 2024, doi: 10.56557/ajomcor/2024/v31i48971.
- [9] I. Izhar, A. Abdullah, M. Zunnurain Hussain, M. Zulkifl Hasan, and C. Author, "Spectrum of Engineering Sciences ISSN (e) 3007-3138 (p) 3007-312X ENHANCING IOT/IIOT INTRUSION DETECTION: A COMPARATIVE STUDY OF HYBRID CNN-LSTM AND ADVANCED DNN ML MODEL ON EDGE-IIOTSET," 2025, doi: 10.5281/zenodo.17491327.
- [10] Maqbool, M. S., Hanif, I., Iqbal, S., Basit, A., & Shabbir, A. (2023). Optimized feature extraction and cross-lingual text reuse detection using ensemble machine learning models. *Journal of Computing & Biomedical Informatics*, 5(01), 26-40.
- [11] Abid, K., Aslam, N., Fuzail, M., Maqbool, M. S., & Sajid, K. (2023). An efficient deep learning approach for prediction of student performance using a neural network. *VFAST Transactions on Software Engineering*, 11(4), 67-79.
- [12] Kanwal, F., Abid, M. K., Maqbool, M. S., Aslam, N., & Fuzail, M. (2023). Optimized classification of cardiovascular disease using machine learning paradigms. *VFAST Transactions on Software Engineering*, 11(2), 140-148.

- [13] Aslam, N., Meeran, M. T., Aslam, M., Maqbool, M. S., & Saeed, B. (2025). Understanding Urban Expansion Through Multi-Temporal Satellite Data Analysis. *Kashf Journal of Multidisciplinary Research*, 2(09), 252-273.
- [14] Hasnain, M. A., Ali, S., Malik, H., Irfan, M., & Maqbool, M. S. (2023). Deep learning-based classification of dental disease using x-rays. *Journal of Computing & Biomedical Informatics*, 5(01), 82-95.
- [15] Basit, A., Hanif, I., Maqbool, M. S., Qayyum, W., Hasnain, M. A., & Nazeer, R. (2023). Cross-lingual information retrieval in a hybrid query model for optimality. *Journal of Computing & Biomedical Informatics*, 5(01), 130-141.
- [16] Hasnain, M. A., Ali, Z., Maqbool, M. S., & Aziz, M. (2024). X-ray image analysis for dental disease: A deep learning approach using efficientnets. *VFAST Transactions on Software Engineering*, 12(3), 147-165.
- [17] Rafiqee, M. M., Qaiser, Z. H., Fuzail, M., Aslam, N., & Maqbool, M. S. (2023). Implementation of efficient deep fake detection technique on videos dataset using deep learning method. *Journal of Computing & Biomedical Informatics*, 5(01), 345-357.
- [18] Maqbool, M. S., Fatima, N., Nazeer, R., Aslam, N., Abbas, F., Sumra, U., & Nadeem, M. (2025). A HYBRID DATASET-BASED ENSEMBLE STRATEGY FOR EFFICIENT BREAST CANCER DETECTION. *Kashf Journal of Multidisciplinary Research*, 2(12), 39-57.
- [19] Muhammad Noman, Muhammad Sajid Maqbool, Dr. Naeem Aslam, Muqadas Nadeem, Hira Saleem, & Hanzla. (2026). SLEEP DISORDER SCORING AUTOMATED USING ADVANCED DATA SCIENCE AND MACHINE LEARNING TECHNIQUES. *Policy Research Journal*, 4(3), 853-868. Retrieved from <https://policyrj.com/1/article/view/1713>
- [20] Zainab Naveed, Rubaina Nazeer, Muhammad Sajid Maqbool, Dr. Naeem Aslam, Hira Saleem, & Muqadas Nadeem. (2026). AN END-TO-END ORTHOPEDIC DISEASE IMAGE CLASSIFICATION SYSTEM USING CONVOLUTIONAL NEURAL NETWORKS. *Policy Research Journal*, 4(3), 837-852. Retrieved from <https://policyrj.com/1/article/view/1712>
- [21] Maqbool, M. S., Fatima, N., Nazeer, R., Aslam, N., Abbas, F., Sumra, U., & Nadeem, M. (2025). A HYBRID DATASET-BASED ENSEMBLE STRATEGY FOR EFFICIENT BREAST CANCER DETECTION. *Kashf Journal of Multidisciplinary Research*, 2(12), 39-57.
- [22] Maqbool, M. S., Fatima, N., Nazeer, R., Aslam, N., Abbas, F., Sumra, U., & Nadeem, M. (2025). A HYBRID DATASET-BASED ENSEMBLE STRATEGY FOR EFFICIENT BREAST CANCER DETECTION. *Kashf Journal of Multidisciplinary Research*, 2(12), 39-57.
- [23] Mahnoor Zaman, Nosheen Fatima, Muhammad Sajid Maqool, Dr. Naeem Aslam, Rubaina Nazeer, & Hira Saleem. (2026). INGREDINET: INTELLIGENT CNN FOR FOOD INGREDIENT RECOGNITION AND CLASSIFICATION. *Policy Research Journal*, 4(3), 789-805.
- [24] Maqbool, M. S., Fatima, N., Nazeer, R., Aslam, N., Abbas, F., Sumra, U., & Nadeem, M. (2025). A HYBRID DATASET-BASED ENSEMBLE STRATEGY FOR EFFICIENT BREAST CANCER DETECTION. *Kashf Journal of Multidisciplinary Research*, 2(12), 39-57.
- [25] Aslam, N., Meeran, M. T., Aslam, M., Maqbool, M. S., & Saeed, B. (2025). Understanding Urban Expansion Through Multi-Temporal Satellite Data Analysis. *Kashf Journal of Multidisciplinary Research*, 2(09), 252-273.

- [26] M. A., Ali, Z., Maqbool, M. S., & Aziz, M. (2024). X-ray image analysis for dental disease: A deep learning approach using efficientnets. *VFAST Transactions on Software Engineering*, 12(3), 147-165.
- [27] Abid, K., Aslam, N., Fuzail, M., Maqbool, M. S., & Sajid, K. (2023). An efficient deep learning approach for prediction of student performance using neural network. *VFAST Transactions on Software Engineering*, 11(4), 67-79.
- [28] A. AlHayan and J. Al-Muhtadi, "A Hybrid STL-Deep Learning Framework for Behavioral-Based Intrusion Detection in IoT Environments," *Applied Sciences (Switzerland)*, vol. 15, no. 12, Jun. 2025, doi: 10.3390/app15126421.
- [29] S. Sadhwani, M. A. H. Khan, R. Muthalagu, P. M. Pawar, and K. Suresh, "A hybrid BiLSTM-CNN approach for intrusion detection for IoT applications," *Sci. Rep.*, vol. 16, no. 1, Dec. 2026, doi: 10.1038/s41598-025-29079-y.
- [30] B. Suri, V. Reddy, and S. Srinivasan, "INTERNET OF THINGS (IOT) NETWORKS: AI-POWERED SECURE INTRUSION DETECTION," *Int. J. Appl. Math. (Sofia)*, vol. 38, no. 9, p. 2025.
- [31] M. A. Talukder, M. Khalid, and N. Sultana, "A hybrid machine learning model for intrusion detection in wireless sensor networks leveraging data balancing and dimensionality reduction," *Sci. Rep.*, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-87028-1.
- [32] M. Ge, N. F. Syed, X. Fu, Z. Baig, and A. Robles-Kelly, "Toward a Deep Learning-Driven Intrusion Detection Approach for Internet of Things," Jul. 2020, [Online]. Available: <http://arxiv.org/abs/2007.09342>

