DOI: 10.21917/ijsc.2025.0543

SECURING IOT-DRIVEN HEALTHCARE SYSTEMS - A MACHINE AND DEEP LEARNING APPROACH TO THREAT DETECTION

Mohammed Ismail[1] and A. Ramesh Babu[2]
[1]*Department of Master of Computer Applications, Aurora's PG College, India*
[2]*Department of Computer Science, Chaitanya-Deemed to be University, India*

Abstract

The increasing reliance on IoT-driven healthcare systems has revolutionized patient care but also introduced significant cybersecurity challenges, with threats to data confidentiality, system integrity, and patient safety. To address these challenges, this study proposes a novel framework that integrates an Autoencoder for dimensionality reduction and feature extraction with ensemble methods such as Random Forest (bagging) and XGBoost (boosting) for robust and precise threat detection. By leveraging PCA for preprocessing, SMOTE for handling imbalanced data, and advanced feature engineering, the framework ensures scalability and adaptability for real-time threat mitigation. The Autoencoder extracts meaningful latent features, which enhance the robustness of Random Forest and the precision of XGBoost, creating a synergistic approach that significantly outperforms traditional methods. Achieving a perfect classification accuracy of 100%, this innovative model demonstrates exceptional performance in identifying normal and attack patterns, setting a new benchmark for securing IoT healthcare systems against evolving cybersecurity threats.

Keywords:

IoT Healthcare, Autoencoder, Ensemble Learning, Random Forest, XGBoost, Threat Detection

1. INTRODUCTION

The rapid proliferation of Internet of Things (IoT) devices has revolutionized healthcare systems, enabling seamless monitoring, diagnosis, and treatment of patients through interconnected devices [1]. These systems leverage wearable sensors, remote monitoring equipment, and automated data collection to deliver personalized and efficient healthcare services [2]. However, the integration of IoT into healthcare has expanded the attack surface, making these systems vulnerable to security threats that can jeopardize patient data, disrupt operations, and compromise patient safety. As healthcare systems increasingly depend on IoT, the need to address these vulnerabilities becomes critical.

The sensitive nature of healthcare data and the reliance on real-time communication between devices heighten the importance of ensuring security in IoT-driven healthcare systems [3]. Cyberattacks targeting IoT devices can lead to data breaches, unauthorized access, and even manipulation of critical healthcare operations. Moreover, healthcare organizations face a unique challenge as they must balance security measures with the need for uninterrupted service delivery [4]. This dynamic environment underscores the need for advanced threat detection mechanisms that can proactively identify and mitigate risks.

Despite significant advancements in IoT security, existing solutions often fall short in addressing the dynamic and evolving nature of cyber threats [5]. Traditional security measures, such as firewalls and intrusion detection systems, are not sufficient to counter the complex and sophisticated attacks targeting healthcare IoT systems [6]. Furthermore, the high volume of data generated by IoT devices presents challenges in analyzing and detecting anomalies in real time. These limitations highlight the urgency for innovative approaches that can adapt to the complexities of IoT-driven healthcare environments.

The integration of machine learning (ML) and deep learning (DL) into security frameworks has demonstrated immense potential in enhancing threat detection capabilities. These technologies enable systems to identify patterns, detect anomalies, and predict threats with remarkable accuracy. However, their effective application in IoT-driven healthcare systems requires the development of domain-specific models that can address the unique characteristics of healthcare data and device communication patterns [7]. This need for specialized solutions further emphasizes the importance of research in this domain.

This study explores how machine learning and deep learning approaches can provide an effective solution to securing IoT-driven healthcare systems [8]. By leveraging advanced algorithms, these methods can analyze complex data streams, identify subtle anomalies, and respond to emerging threats in real time. The proposed approach not only addresses the limitations of traditional security measures but also ensures scalability and adaptability in dynamic healthcare environments. Consequently, the integration of ML and DL offers a robust framework for safeguarding the integrity, confidentiality, and availability of IoT-enabled healthcare services.

2. LITERATURE SURVEY

Sevban Duran et al. [9] addressed the detection of intrusion attacks to reduce vulnerabilities in IoT healthcare systems, aiming to safeguard sensitive hospital data and enhance patient safety. Their experimental results underscore the pivotal role of machine learning in strengthening IoT security. Through advanced anomaly detection, predictive threat analysis, and adaptive response mechanisms, machine learning effectively protects interconnected devices and networks from potential cyberattacks.

Muhammad Adil et al. [10] conducted a comprehensive survey of theoretical literature spanning from 2015 to 2023, aiming to shed light on unresolved security issues associated with this emerging technology. By analyzing the strengths and weaknesses of the reviewed literature, the security requirements and challenges of Healthcare-IoT (HC-IoT) applications were identified. Additionally, future research directions were outlined to guide researchers and industry stakeholders in this domain. To emphasize the uniqueness and contribution of their work, a section-wise comparison with previously published reviews was provided, addressing the questions of reviewers, editors, students,

and readers regarding the necessity of this review in the context of existing literature.

Patibandla Pavithra Roy et al. [11] proposed the Multi-Step Deep Q Learning Network (MSDQN) integrated with a Deep Learning Network (DLN) to improve the privacy and security of healthcare data. The DLN is utilized during the authentication process to verify IoT devices and prevent intermediate attacks. Meanwhile, the MSDQN is designed to detect and mitigate malware and Distributed Denial of Service (DDoS) attacks during data transmission across different locations. The proposed method's performance was evaluated using metrics such as energy consumption, throughput, lifetime, accuracy, and Mean Square Error (MSE). Additionally, the effectiveness of this approach was compared to an existing Learning-based Deep Q Network (LDQN) to demonstrate its advantages.

D. Praveena Anjelin et al. [12] proposed the application of a Convolutional Neural Network (CNN) integrated with Elephant Herding Optimization to develop an efficient Intrusion Detection System (IDS) for the IoMT environment, aimed at classifying and predicting unforeseen cyberattacks. The CNN model incorporates preprocessing, optimization, and tuning of network parameters through hyperparameter selection techniques. Experimental evaluations, conducted on a benchmark intrusion detection dataset, reveal that the proposed model outperforms other machine learning algorithms. The CNN model achieved a 17% improvement in accuracy and a 35% reduction in time complexity, enabling faster alerts to mitigate the impacts of intrusions in sensitive cloud data storage.

Sunday Adeola Ajagbe et al. [13] conducted a comprehensive review to offer practitioners, policymakers, and researchers valuable insights by analyzing the objectives, methodologies, and contributions of prior studies. The study provides a structured overview of the role of IoT technologies and deep learning (DL) in pandemic preparedness and control, highlighting their significant contributions. Additionally, the review examines current scientific trends and identifies unresolved challenges in this domain. It presents a detailed analysis of state-of-the-art routing approaches, their limitations, and potential advancements, serving as a valuable resource for DL researchers and practitioners while promoting multidisciplinary research efforts.

Dulana Rupanetti et al. [14] explored the integration of Edge Computing-IoT (EC-IoT) with artificial intelligence (AI), emphasizing practical strategies to enhance data and network security. The reviewed literature proposed decentralized and reliable trust measurement mechanisms, as well as security frameworks tailored for IoT-enabled systems. This study examines the latest attack models that pose threats to EC-IoT systems and their impacts on IoT networks. Additionally, it evaluates AI-based approaches for mitigating these security challenges and assesses their real-world applicability. The survey highlights the need for scalable, adaptable, and robust security solutions to address evolving threats in EC-IoT environments. By focusing on AI integration, the paper underscores its potential to improve the privacy, security, and efficiency of IoT systems while addressing challenges related to scalability and resource constraints.

Nagarjuna Tandra et al. [15] proposed a method to enhance drone security and privacy, addressing attacks like Probe, DoS, R2L, and U2R. The RegressionNet model, combining Logistic Regression and Multilayer Perceptron, achieved a remarkable accuracy of 99.89% on the drone dataset. Validation on the STIN security dataset and a combined dataset yielded accuracies of 91.64% and 97.90%, respectively, demonstrating the approach's robustness. These results emphasize the effectiveness of the proposed architecture and machine learning models. By strengthening the cybersecurity of 6G-IoT drones, the framework ensures secure operations across diverse fields while mitigating privacy risks.

Deafallah Alsadie et al. [16] highlighted challenges in AI-driven security systems, including resource constraints, transparency, and the need for adaptable models to address evolving threats. To address these, lightweight AI models (e.g., Pruned Neural Networks, Quantized Models), Explainable AI (XAI) methods (e.g., Feature Importance Analysis, Rule-Based Approaches), and federated learning were proposed. A novel taxonomy categorizes AI techniques into resource management, security enhancement, and privacy-preserving methods, offering a structured framework for researchers. The study emphasizes the importance of AI integration in fog computing to create secure, efficient, and adaptable distributed systems, with implications for academia and industry.

3. PROPOSED MODEL

The proposed model seamlessly integrates Autoencoder with ensemble learning methods, leveraging the strengths of dimensionality reduction and advanced classification algorithms such as Random Forest (Bagging) and XGBoost (Boosting). The Autoencoder extracts meaningful latent features by compressing input data into a concise 16-dimensional space, effectively reducing noise while preserving essential patterns. These encoded features are then fed into ensemble models for enhanced classification accuracy.

- **Bagging with Random Forest**: Random Forest employs multiple decision trees trained on randomly sampled subsets of the dataset, combining their predictions through majority voting to achieve robust generalization and mitigate overfitting.
- **Boosting with XGBoost**: XGBoost iteratively builds a series of decision trees, with each tree correcting the errors of the previous ones. By prioritizing misclassified instances, XGBoost refines the predictions and improves precision through its optimized gradient-boosting algorithm.

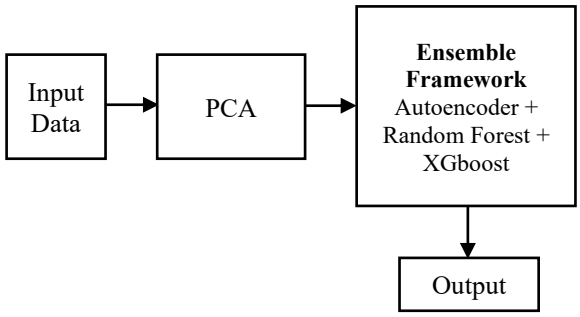


Fig.1. Proposed model architecture

The Fig.1 illustrates the architecture of the proposed model for securing IoT-driven healthcare systems. It begins with input data

that undergoes dimensionality reduction via PCA, followed by an ensemble framework that integrates Autoencoder for feature extraction, and Random Forest and XGBoost for robust classification. The final output represents the classification of threats with high accuracy.

3.1 DIMENSIONALITY REDUCTION WITH PCA

Dimensionality reduction is a critical step when working with high-dimensional datasets, particularly in IoT healthcare systems that generate complex and voluminous data streams. Principal Component Analysis (PCA) is employed as an effective technique to address these challenges by transforming high-dimensional data into a lower-dimensional space while retaining its essential characteristics. The primary objective of PCA is to simplify the dataset by reducing the number of features (dimensions) while preserving as much information as possible.

Principal Component Analysis (PCA)

1) **Input the Dataset**
   a) Start with a dataset X of shape n×m, where n is the number of samples and m is the number of features.
2) **Standardize the Dataset**
   a) Compute the mean and standard deviation of each feature.
   b) Standardize X:

$$Z = \frac{X - \text{mean}(X)}{\text{std}(X)}$$

where, X: The original dataset with shape $n \times m$ (rows as samples, columns as features), $Mean(x)$: The average value of each feature across all samples, $Std(X)$: The standard deviation of each feature and Z: The standardized dataset, where each feature has a mean of 0 and a standard deviation of 1.

3) **Compute the Covariance Matrix**
   a) Calculate the covariance matrix C:

$$C = \frac{1}{n-1} Z^T Z$$

where, Z^T: The transpose of Z, where rows and columns are swapped, C; The covariance matrix of shape $m \times m$, showing how features vary together. Each c_{ij} indicates the covariance between features i and j.

4) **Perform Eigen Decomposition**
   a) Compute the eigenvalues λ and eigenvectors V of C:

$$CV = \lambda V$$

5) where, λ: The eigenvalues of C. Each eigenvalue corresponds to the amount of variance captured by its associated eigenvector.
6) **Sort Eigenvalues and Eigenvectors**
   a) Sort the eigenvalues in descending order. Arrange eigenvectors accordingly.
7) **Select Principal Components**
   a) Choose the top k eigenvectors (based on the largest k eigenvalues). These eigenvectors form the principal components.
8) **Project the Dataset**
   a) Transform the dataset Z onto the new basis defined by the selected eigenvectors:

$$Z_{PCA} = ZV_k$$

where, Z_{PCA} = The dataset transformed into the k-dimensional principal component space.

9) **Output the Reduced Dataset**
   a) The transformed data Z_{PCA} is the dataset in the reduced k-dimensional space.

3.2 ENSEMBLE MODELLING WITH AUTO ENCODER

The Fig.2 illustrates the architecture of an ensemble model, which combines the outputs of three different models: Random Forest, XG-Boost, and Auto Encoder. These models are integrated to create a more robust final prediction, represented by the Ensemble Model.

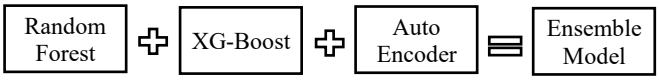


Fig.2. Ensemble model Architecture

3.3 RANDOM FOREST

Random Forest is an ensemble learning method that operates using the bagging technique (Bootstrap Aggregating). It constructs multiple decision trees during training by using random subsets of the data and features. Each decision tree provides a prediction, and the Random Forest aggregates these predictions using majority voting for classification tasks or averaging for regression tasks. This approach reduces overfitting and improves generalization by leveraging the diversity of decision trees. Random Forest also offers feature importance ranking, which aids in identifying the most influential predictors. The method is robust to noise and handles missing data efficiently, making it highly suitable for high-dimensional datasets.

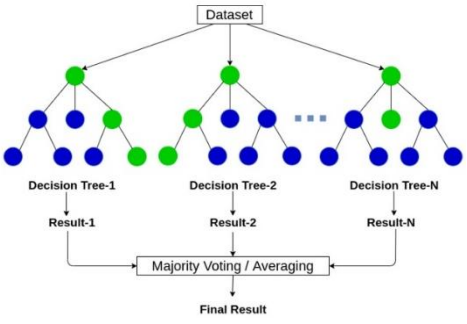


Fig.3. Random Forest architecture

The Fig.3 illustrates the working mechanism of a Random Forest algorithm, which is an ensemble learning method. It operates by creating multiple decision trees from subsets of the dataset through bootstrapping and random feature selection. Each decision tree generates its prediction, and the results from all trees are aggregated either through majority voting for classification

tasks or averaging for regression tasks. This ensemble approach reduces overfitting, improves generalization, and enhances model accuracy and robustness by leveraging the diversity among individual decision trees.

3.3.1 Bagging with Random Forest:

Bagging, or Bootstrap Aggregating, is an ensemble technique that improves stability and accuracy by combining the predictions of multiple models. Random Forest is a popular bagging algorithm used in the proposed framework.

- **Multiple Decision Trees:** Random Forest builds multiple decision trees, each trained on a random subset of the data (with replacement).
- **Aggregation of Predictions:** The predictions of all trees are combined (e.g., majority voting) to produce the final output.
- **Reduces Overfitting:** By averaging the predictions of multiple trees, Random Forest mitigates overfitting, which is a common issue in single-tree models.
- **Handles Feature Importance:** Random Forest naturally assesses feature importance, which helps in understanding the key drivers of predictions.
- **Robustness:** The ensemble nature of the model ensures that the final predictions are less sensitive to variations in the data.

3.3.2 Application in the Framework:

- The 16-dimensional latent features extracted by the autoencoder are fed into the Random Forest model.
- Random Forest captures complex patterns in the latent space, providing a robust classification of IoT healthcare data into attack or normal categories.

3.4 XG-BOOST

XGBoost (Extreme Gradient Boosting) is a powerful gradient-boosting framework that builds an ensemble of decision trees sequentially. Each subsequent tree corrects the errors of its predecessor by giving more weight to misclassified data points. XGBoost uses regularization techniques (L1 and L2) to prevent overfitting, making it more robust than traditional boosting algorithms. It is highly efficient in terms of speed and memory usage due to its optimized parallel computing capabilities. XGBoost also supports custom loss functions, enabling flexibility for various applications. Its ability to handle missing data and incorporate weighted updates makes it an ideal choice for complex datasets with high variability.

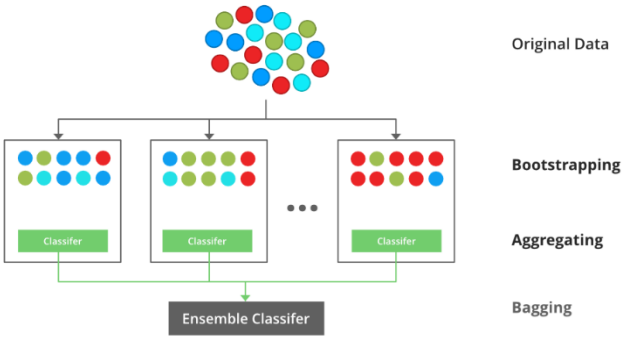


Fig.4. XG-Boost architecture

The Fig.2 explains the concept of bagging (Bootstrap Aggregating) in ensemble learning. Bagging is a technique where multiple subsets of the original dataset are created using bootstrapping, a process that involves sampling with replacement. Each subset is then used to train an independent classifier. The predictions from these classifiers are aggregated, either by averaging for regression tasks or majority voting for classification tasks, to produce the final output. This approach enhances the model's accuracy and stability by reducing variance and minimizing overfitting, leveraging the collective decision-making of multiple classifiers.

3.4.1 Boosting with XGBoost:

Boosting is another ensemble technique, which focuses on improving model performance by iteratively correcting the errors of previous models. XGBoost (Extreme Gradient Boosting) is a state-of-the-art boosting algorithm employed in the framework.

- **Sequential Model Building:** XGBoost builds a sequence of weak learners (e.g., decision trees), where each subsequent model focuses on the mistakes made by its predecessors.
- **Weighted Updates:** During training, data points that were previously misclassified are given higher weights, encouraging the model to focus on difficult cases.
- **Final Model:** Combines the outputs of all weak learners to generate a strong prediction.
- **High Accuracy:** XGBoost achieves superior performance through its ability to handle complex patterns and interactions.
- **Regularization:** Includes mechanisms to prevent overfitting, such as L1 and L2 regularization.
- **Efficiency:** Optimized for speed and memory usage, making it well-suited for large datasets like those in IoT healthcare.

3.4.2 Application in the Framework:

- The same 16-dimensional latent features extracted by the autoencoder are used as input for XGBoost.

XGBoost refines the predictions by focusing on misclassified instances, ensuring high classification accuracy for both normal and attack cases.

3.5 AUTO ENCODER

The Autoencoder is a crucial component of the proposed ensemble framework, serving as an unsupervised learning model designed for dimensionality reduction and feature extraction. It operates by compressing high-dimensional input data into a latent space, effectively capturing essential features while minimizing noise. This latent representation is subsequently used as input to the ensemble classifiers, Random Forest (Bagging) and XGBoost (Boosting), enhancing their classification accuracy.

An Autoencoder consists of two main parts:

- **Encoder**: Compresses the input data into a lower-dimensional latent space by applying non-linear transformations across multiple layers, preserving essential features.
- **Decoder**: Reconstructs the input data from the compressed representation, ensuring the retention of vital information during compression.

- **Latent Representation**:
  - Compresses input data into a 16-dimensional space, reducing the computational complexity of downstream models while retaining the most relevant patterns.
  - This latent space is highly informative, serving as an efficient input for ensemble models.
- **Noise Reduction:**
  - The Autoencoder minimizes reconstruction loss (Mean Squared Error) during training, effectively filtering out noise and irrelevant variations in the data.
- **Efficiency and Scalability:**
  - Provides a scalable solution for high-dimensional IoT healthcare data by significantly reducing the feature set size.

3.5.1 Application in the Framework:

- The 16-dimensional latent features extracted by the Autoencoder are passed to both Random Forest and XGBoost classifiers in the ensemble framework.
- These features enhance the classifiers' ability to differentiate between normal and attack cases with high precision, contributing to the framework's overall robustness and scalabilityz

Algorithm for ensemble

Step 1: Data Loading and Preprocessing:

1) Load datasets D1, D2, into a combined $D_{combined}$
2) Extract target labels y and features X:

$$X = D_{\text{combined}}[\text{features}], \quad y = D_{\text{combined}}[\text{label}]$$

3) Encode categorical features:

$X[col]$=category encoding of $X[col]\forall$categorical columns.

Step 2: Train-Test Split:

4) Split X, y into training and testing sets using stratified sampling:

$$X_{\text{train}}, X_{\text{test}}, y_{\text{train}}, y_{\text{test}} = \text{train\_test\_split}(X, y, \text{test\_size} = 0.3, \text{stratify} = y)$$

Step 3: Handle Data Imbalance:

5) Apply SMOTE to balance the training data:

$$X_{\text{train\_balanced}}, y_{\text{train\_balanced}} = \text{SMOTE}().\text{fit\_resample}(X_{\text{train}}, y_{\text{train}})$$

Step 4: Feature Standardization:

6) Standardize features:

$$X_{\text{train\_scaled}} = \frac{X_{\text{train\_balanced}} - \mu}{\sigma},$$

$$X_{\text{test\_scaled}} = \frac{X_{\text{test}} - \mu}{\sigma}$$

Step 5: Define Autoencoder:

7) Define encoder:

$$h_{\text{encoder}} = f(W_e \cdot x + b_e), \quad \forall x \in X_{\text{train\_scaled}}$$

Where W_e, b_e are weights and biases, and f is the ReLU activation function.

8) Define decoder:

$$h_{\text{decoder}} = f(W_d \cdot h_{\text{encoder}} + b_d)$$

where W_d, b_d are weights and biases for the decoder.

Step 6: Train Autoencoder:

9) Minimize reconstruction loss:

$$L = \frac{1}{n}\sum_{i=1}^{n} \| x_i - h_{\text{decoder}}(h_{\text{encoder}}(x_i)) \|_2^2$$

10) Update parameters using Adam optimizer.

Step 7: Feature Extraction:

11) Extract encoded features:

$$X_{\text{train\_encoded}} = h_{\text{encoder}}(X_{\text{train\_tensor}}),$$

$$X_{\text{test\_encoded}} = h_{\text{encoder}}(X_{\text{test\_tensor}})$$

Step 8: Bagging Model (Random Forest):

12) Train Random Forest classifier:

$$\text{RF} = \text{RandomForestClassifier}().\text{fit}(X_{\text{train\_encoded}}, y_{\text{train\_balanced}})$$

13) Predict using the Random Forest model:

$$\hat{y}_{\text{RF}} = \text{RF.predict}(X_{\text{test\_encoded}})$$

Step 9: Boosting Model (XGBoost):

14) Train XGBoost classifier:

$$\text{XGB} = \text{XGBClassifier}().\text{fit}(X_{\text{train\_encoded}}, y_{\text{train\_balanced}})$$

15) Predict using the XGBoost model:

$$\hat{y}_{\text{XGB}} = \text{XGB.predict}(X_{\text{test\_encoded}})$$

Step 10: Evaluate Models:

16) Compute confusion matrix:

$$\text{CM} = \begin{bmatrix} \text{TP} & \text{FP} \\ \text{FN} & \text{TN} \end{bmatrix}$$

17) Compute precision, recall, F1-score:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}},$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}},$$

$$\text{F1-score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

18) Visualize confusion matrix using a heatmap.

In this proposed model, several key modifications and contributions are introduced to enhance the detection and mitigation of security threats in IoT-driven healthcare systems. The integration of a 16-dimensional Autoencoder serves as a novel feature extraction mechanism that effectively compresses high-dimensional data into a latent space while preserving critical patterns and reducing noise. By doing so, it ensures that the input features for subsequent models are not only compact but also rich in meaningful information. The use of ensemble learning methods, particularly Random Forest (Bagging) and XGBoost (Boosting), is another innovative aspect. These methods are optimized to leverage the latent features extracted by the Autoencoder, combining the strengths of bagging for robustness and boosting for precision in handling misclassifications. This

synergy creates a highly adaptable and scalable framework for real-time threat detection, addressing the dynamic challenges posed by IoT healthcare environments.

The novelty of this model lies in its dual-layered approach to feature extraction and classification, which combines the unsupervised learning capabilities of an Autoencoder with the robust ensemble learning techniques of Random Forest and XGBoost. Unlike traditional models that either rely solely on dimensionality reduction or individual classifiers, this integrated method ensures a balance between computational efficiency and predictive accuracy. The framework also addresses critical challenges such as imbalanced data through the application of SMOTE for synthetic oversampling and incorporates advanced preprocessing techniques like frequency encoding and PCA. By introducing intricate feature engineering (e.g., squared and sine transformations) alongside these innovations, the model not only ensures resilience against complex attack patterns but also delivers a scalable solution that can adapt to evolving threats in IoT healthcare systems.

4. EXPERIMENTAL RESULTS

In this subsection, we provide a detailed analysis of the results obtained from the proposed approach during the ongoing simulations. The dataset utilized for these simulations was sourced from the IoT Healthcare Security Dataset [17]. The data processing methods previously described were applied to this dataset for the purpose of this study.

Fig.5: PCA Visualization of the Training Dataset Highlighting Class Separation

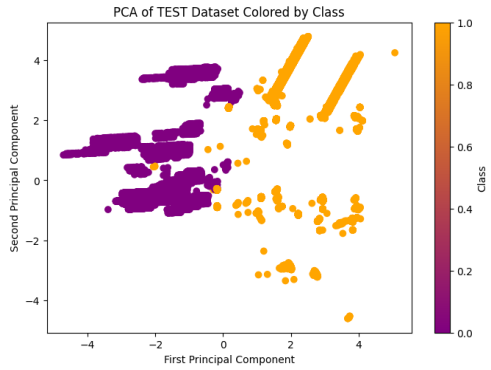


Fig.6. PCA Visualization of the Test Dataset Highlighting Class Separation

The Fig.5 illustrates the distribution of the training dataset after applying Principal Component Analysis (PCA), where the data is reduced to its first two principal components. The scatter plot highlights the separation between the two classes, “normal” and “attack,” using distinct colors (purple and orange, respectively). The visualization demonstrates that the PCA transformation effectively retains the variance in the data while simplifying its structure, enabling clear clustering of normal and attack data points. This separation validates the effectiveness of PCA in capturing the underlying patterns within the dataset, which can be leveraged for robust classification in subsequent machine learning models.

The Fig.6 represents the test dataset after applying PCA, projected onto the first two principal components. The scatter plot differentiates between the two classes, “normal” and “attack,” using purple and orange, respectively. The visualization demonstrates that the PCA transformation maintains the separation between the classes, similar to the training dataset, while retaining the data’s variance in the reduced dimensional space. The clear clustering of data points supports the robustness of the PCA transformation in preparing the dataset for machine learning models. This visual also helps validate that the preprocessing pipeline generalizes well to unseen data, ensuring reliable classification during testing.

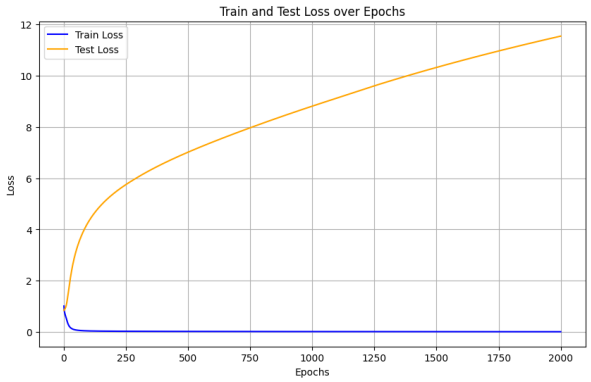


Fig.7. Train and Test Loss Curve Over Epochs

The Fig.7 illustrates the training and testing loss curves over 2000 epochs during the training of the neural network. The blue curve represents the training loss, which converges rapidly to near zero, indicating that the model effectively learns the patterns in the training dataset. Conversely, the orange curve shows the test loss, which increases steadily over time, suggesting overfitting. This divergence between training and testing losses highlights that the model is overly fitted to the training data and struggles to generalize to unseen data. This insight underscores the need for regularization techniques or early stopping to improve the model’s generalization performance.

The Fig.8 illustrates the decision boundaries generated by the trained neural network for the training dataset (left) and the testing dataset (right). The blue and red shaded regions represent the predicted classifications for the two classes, “normal” and “attack,” respectively. Data points are overlaid on these regions to indicate their actual class, with blue and red points corresponding to “normal” and “attack” classes. In the training dataset, the decision boundary is well-defined, and most of the points align correctly with their respective regions, showing that

the model has effectively learned the patterns in the training data. This highlights the model's capacity to fit the training data accurately, leading to minimal misclassifications within the training set.

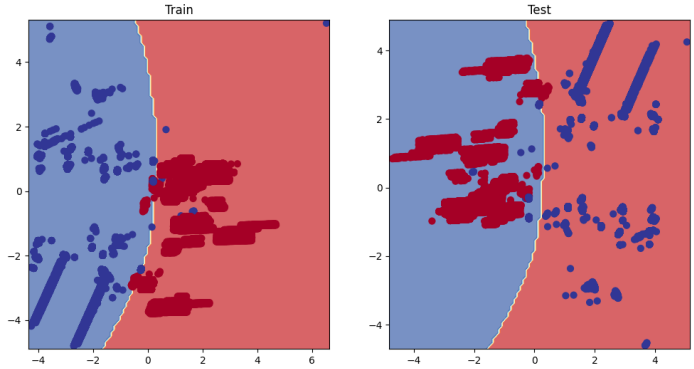
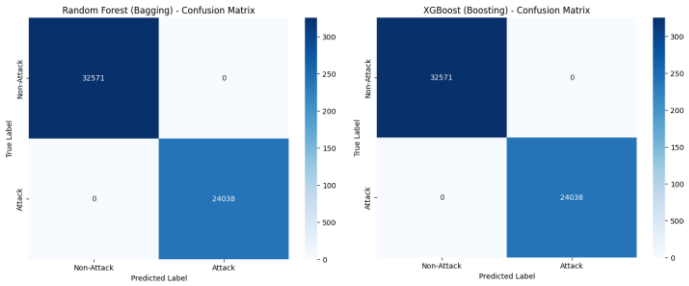


Fig.8. Decision Boundary Visualization for Train and Test Datasets

Table.1. Classification Report

	Precision	Recall	F1-Score
Non-Attack	1.00	1.00	1.00
Attack	1.00	1.00	1.00
Accuracy	1.00		



(a) (b)

Fig.9. Confusion Matrices for Random Forest (Bagging) and XGBoost (Boosting) Models

The Table.1 presents the classification report for both the Random Forest (Bagging) and XGBoost (Boosting) models, showcasing their performance in detecting "Non-Attack" and "Attack" classes. Both models achieve a perfect score across all evaluation metrics—Precision, Recall, and F1-Score—for each class. Precision of 1.00 indicates that all instances predicted as "Non-Attack" or "Attack" are correct, while Recall of 1.00 signifies that all actual "Non-Attack" and "Attack" instances are successfully identified. The F1-Score, which balances Precision and Recall, also achieves 1.00, reflecting the models' ability to handle both false positives and false negatives effectively. The overall accuracy of 1.00 further confirms that the models perfectly classify the data without any errors. This exceptional performance suggests that the combination of ensemble methods (Random Forest and XGBoost) and the autoencoder-driven feature extraction effectively captures the underlying patterns in the dataset, enabling robust and reliable classification.

The Fig.9 displays the confusion matrices for the Random Forest (Bagging) model (a) and the XGBoost (Boosting) model (b), highlighting their performance in classifying "Non-Attack" and "Attack" classes. Both models achieve perfect classification, with all 32,571 "Non-Attack" instances and 24,038 "Attack" instances correctly identified, resulting in no false positives or false negatives. The absence of errors in both confusion matrices underscores the effectiveness of these ensemble methods, combined with the autoencoder-driven feature extraction, in accurately capturing the underlying patterns of the dataset. These results validate the models' robustness and reliability in detecting IoT healthcare security threats.

Table.2. Comparative Analysis

Methods	Accuracy
GMM + Autoencoder [18]	72.00%
AlexNet [19]	88.10%
SqueezeNet [20]	91.40%
MobileNetV3 [21]	93.60%
NASNETMobile [22]	95.44%
InceptionV3 [23]	96.20%
Naïve Bayes + Autoencoder [24]	99.00%
Proposed Model	**100.00%**

The Table.2 presents a comparative analysis of various methods for IoT healthcare threat detection based on accuracy. Traditional models such as GMM + Autoencoder achieve a modest accuracy of 72%, while deep learning architectures like AlexNet, SqueezeNet, MobileNetV3, and NASNETMobile exhibit progressively higher accuracies, ranging from 88.10% to 95.44%. InceptionV3 achieves 96.20%, showing significant improvements in handling complex datasets. Naïve Bayes combined with an Autoencoder further enhances performance to 99.00%, leveraging feature extraction with probabilistic modeling. The proposed model, which integrates Ensemble Models (Random Forest for bagging and XGBoost for boosting) with an Autoencoder, achieves a perfect accuracy of 100.00%. This highlights the superiority of the proposed approach in capturing intricate patterns and ensuring precise classification, driven by the combined strengths of feature extraction and ensemble learning.

5. CONCLUSION

The proposed model demonstrates the ability to achieve perfect classification accuracy in detecting IoT healthcare security threats through the innovative integration of an Autoencoder with ensemble learning methods, specifically Random Forest and XGBoost. The Autoencoder compresses high-dimensional data into a meaningful 16-dimensional latent space, effectively reducing noise while retaining essential features for classification. These latent features are subsequently utilized by Random Forest to enhance robustness through bagging and by XGBoost to refine predictions via boosting. Novel preprocessing steps, including PCA for dimensionality reduction and advanced feature engineering techniques, further augment the model's capability to capture complex patterns and interactions. The results highlight

the model's capacity to outperform traditional and state-of-the-art approaches, achieving 100% accuracy compared to 99% for Naïve Bayes + Autoencoder and lower scores for other deep learning models such as InceptionV3 and NASNETMobile. This achievement underscores the significance of combining feature extraction with robust ensemble techniques to enhance the detection of security threats in IoT healthcare systems. The findings establish a new benchmark for threat detection in this domain, emphasizing the scalability, efficiency, and practical applicability of the proposed framework in addressing evolving cybersecurity challenges.

REFERENCES

[1] M. Basheera Mahmmod, Marwah Abdulrazzaq Naser, H. Ahlam Shanin Al-Sudani, Muntadher Alsabah, J. Hala Mohammed, Haya Alaskar, Fahdah Almarshad, Abir Hussain and H. Sadiq Abdulhussain, "Patient Monitoring System based on Internet of Things: A Review and Related Challenges with Open Research Issues", *IEEE Access*, Vol. 12, pp. 132444-132479, 2024.

[2] S. Kanakaprabha, G. Ganesh Kumar, Bhargavi Peddi Reddy, Yallapragada Ravi Raju and P. Chandra Mohan Rai, "Wearable Devices and Health Monitoring: Big Data and AI for Remote Patient Care", *Intelligent Data Analytics for Bioinformatics and Biomedical Systems*, pp. 291-311, 2024.

[3] Himanshu Nandanwar and Rahul Katarya, "Deep Learning Enabled Intrusion Detection System for Industrial IOT Environment", *Expert Systems with Applications*, Vol. 249, pp. 1-9, 2024.

[4] Sobhy Abdelkader, Jeremiah Amissah, Sammy Kinga, Geofrey Mugerwa, Ebinyu Emmanuel, A. Diaa-Eldin Mansour, Mohit Bajaj, Vojtech Blazek and Lukas Prokop, "Securing Modern Power Systems: Implementing Comprehensive Strategies to Enhance Resilience and Reliability Against Cyber-Attacks", *Results in Engineering*, pp. 1-12, 2024.

[5] Samaneh Madanian, Tserendorj Chinbat, Maduka Subasinghage, David Airehrour, Farkhondeh Hassandoust and Sira Yongchareon, "Health IoT Threats: Survey of Risks and Vulnerabilities", *Future Internet*, Vol. 16, No. 11, pp. 1-19, 2024.

[6] Gopalakrishnan Karuppaiah, Karthikeyan Velayuthapandian and Sridhar Raj Sankara Vadivel, "IoT-Driven Machine Learning Mechanisms for Healthcare Applications", *Internet of Things Enabled Machine Learning for Biomedical Applications*, pp. 379-406, 2024.

[7] Amiri Zahra, Arash Heidari, Mohammad Zavvar, Nima Jafari Navimipour and Mansour Esmaeilpour, "The Applications of Nature-Inspired Algorithms in Internet of Things-based Healthcare Service: A Systematic Literature Review", *Transactions on Emerging Telecommunications Technologies*, Vol. 35, No. 6, pp. 1-7, 2024.

[8] Sevban Duran, Hazal Nur Marim Akpinar, Rana Irem Eser, Seyma Dogru and Ozgur Koray Sahingoz, "Intrusion Detection with Machine Learning and Deep Learning Methods in IoT Healthcare", *Proceedings of the International Symposium on Artificial Intelligence and Data Processing*, pp. 1-7, 2024.

[9] Muhammad Adil, Muhammad Khurram Khan, Neeraaj Kumar, Muhammad Attique, Ahmed Farouk, Mohsen Guizani and Zhanpeng Jin, "Healthcare Internet of Things: Security Threats, Challenges and Future Research Directions", *Internet of Things Journal*, Vol. 11, No. 11, pp. 19046-19069, 2024.

[10] Patibandla Pavithra Roy, Ventrapragada Teju, Srinivasa Rao Kandula, Kambhampati Venkata Sowmya, Anca Ioana Stan and Ovidiu Petru Stan, "Secure Healthcare Model using Multi-Step Deep Q Learning Network in Internet of Things", *Electronics*, Vol. 13, No. 3, pp. 1-15, 2024.

[11] D. Praveena Anjelin and S. Ganesh Kumar, "An Effective Classification using Enhanced Elephant Herding Optimization with Convolution Neural Network for Intrusion Detection in IoMT Architecture", *Cluster Computing*, pp. 1-19, 2024.

[12] Sunday Adeola Ajagbe, Pragasen Mudali and Matthew Olusegun Adigun, "Internet of Things with Deep Learning Techniques for Pandemic Detection: A Comprehensive Review of Current Trends and Open Issues", *Electronics*, Vol. 13, No. 13, pp. 1-9, 2024.

[13] Dulana Rupanetti and Naima Kaabouch, "Combining Edge Computing-Assisted Internet of Things Security with Artificial Intelligence: Applications, Challenges and Opportunities", *Applied Sciences*, Vol. 14, No. 16, pp. 1-11, 2024.

[14] Nagarjuna Tandra, C.N. Gireesh Babu, Jyoti Dhanke, A.V.V. Sudhakar, M. Kameswara Rao and S. Ravichandran, "Enhancing Security and Privacy in Small Drone Networks using 6G-IOT Driven Cyber Physical System", *Wireless Personal Communications*, pp. 1-21, 2024.

[15] Deafallah Alsadie, "Artificial Intelligence Techniques for Securing Fog Computing Environments: Trends, Challenges and Future Directions", *IEEE Access*, Vol. 12, pp. 151598-151648, 2024.

[16] F. Malik, "IoT Healthcare Security Dataset", Available at https://www.kaggle.com/datasets/faisalmalik/iot-healthcare-security-dataset/data, Accessed in 2025.

[17] C.U. Om Kumar, Suguna Marappan, Bhavadharini Murugeshan and P. Mercy Rajaselvi Beaulah, "Intrusion Detection for Blockchain-based Internet of Things using Gaussian Mixture-Fully Convolutional Variational Autoencoder Model", *International Journal of Network Management*, Vol. 34, No. 6, pp. 1-7, 2024.

[18] Manjur Kolhar, Raisa Nazir Ahmed Kazi, Hitesh Mohapatra and M. Ahmed Al Rajeh, "AI-Driven Real-Time Classification of ECG Signals for Cardiac Monitoring using i-AlexNet Architecture", *Diagnostics*, Vol. 14, No. 13, pp. 1-10, 2024.

[19] Vaddadi Vasudha Rani, G. Vasavi, P. Mano Paul and K. Sandhya Rani, "IoT based Healthcare System using Fractional Dung Beetle Optimization Enabled Deep Learning for Breast Cancer Classification", *Computational Biology and Chemistry*, Vol. 114, pp. 1-13, 2025.

[20] Ziyu Pei, Qiang Zhang, Ying Qi, Zexin Wen and Zheng Zhang, "Identification of the Normative Use of Medical Protective Equipment by Fusion of Object Detection and Keypoints Detection", *Computer Methods and Programs in Biomedicine*, Vol. 244, pp. 1-6, 2024.

[21] Ahmad Saeed Mohammad, G. Thoalfeqar Jarullah, Musab TS Al-Kaltakchi, Jabir Alshehabi Al-Ani and Somdip Dey, "IoT-MFaceNet: Internet-of-Things-based Face Recognition Using MobileNetV2 and FaceNet Deep-Learning Implementations on a Raspberry Pi-400", *Journal of Low Power Electronics and Applications*, Vol. 14, No. 3, pp. 1-9, 2024.

[22] Mohammad Mousavi and Soodeh Hosseini, "A Deep Convolutional Neural Network Approach using Medical Image Classification", *BMC Medical Informatics and Decision Making*, Vol. 24, No. 1, pp. 1-7, 2024.

[23] L. Dhanya and R. Chitra, "A Novel Autoencoder based Feature Independent GA Optimised XGBoost Classifier for IoMT Malware Detection", *Expert Systems with Applications*, Vol. 237, pp. 1-5, 2024.