FedSecure: A Robust Federated Learning Framework for Adaptive Anomaly Detection and Poisoning Attack Mitigation in IoMT

Fawaz J. Alruwaili
Dept. of Computer Science & Eng.
University of North Texas
Denton, USA
fawazalruwaili@my.unt.edu

Saraju P. Mohanty
Dept. of Computer Science & Eng.
University of North Texas
Denton, USA
saraju.mohanty@unt.edu

Elias Kougianos
Dept. of Electrical Eng.
University of North Texas
Denton, USA
Elias.Kougianos@unt.edu

***Abstract*—Deep learning (DL) technologies have been increasingly employed in the Internet of Medical Things (IoMT) for complex classification problems, such as diagnosis and monitoring. Despite the significant benefits, DL has been hindered in many industries due to data privacy concerns, as it requires vast amounts of data to predict accurate results. To address this issue, Google introduced a distributed learning approach called Federated Learning (FL) where model can be trained locally by sharing gradients rather than raw data. However, FL also introduces new security threats, such as poisoning attacks, where learning process can be corrupted by malicious clients. While there are many studies have addressed FL security, they have not including holistic considerations regarding data diversity which affects the generalization of proposals to be used in real-world applications. In this paper, we propose FedSecure, an adaptive anomaly detection and poisoning attack mitigation framework for IoMT, using hybrid deep learning models. Our FedSecure was tested on distinct and diverse real-world datasets. Experimental results show that FedSecure was able to detect and mitigate poisoning attacks, thereby enhancing the security of federated learning systems in real-world applications.**

***Index Terms*—Healthcare Cyber-Physical System (H-CPS), Internet of Medical Things (IoMT), Intelligent Security, Cybersecurity, Federated Learning, Poisoning Atatcks, Anomaly Detection**

I. Introduction

The traditional healthcare system has transformed into a smart system due to the revolution of electronic devices and communication infrastructure. The medical devices and applications are integrated by the internet of medical things (IoMT) network, enabling remote and continuous patient monitoring, diagnostics, and personalized treatment plans, which improved the life and human quality. Artificial intelligence (AI) technologies are integrated into the IoMT to enhance the healthcare services and enabling a proactive management by predicting adverse events before they occur, leading for better patient outcomes [1].

Despite the significant benefits of AI technologies in IoMT, the are some challenges lead to hiner the adaption of such technologies [2]. For example, the traditional healthcare management techniques rely on a centralized framework for data analytic, which makes data privacy and security are paramount concerns, especially when dealing with medical data that has higher sensitive nature as it is not only affect the privacy, but also threaten patient life when medical decisions are based on manipulated data [3]. Additionally, AI technologies require vast amounts of data for training process to get accurate predictions, which increase the privacy and security threats [4], [5]. These challenges increased the need for a distributed and scalable AI-based framework, and preserving the privacy.

Consequently, a distributive AI paradigm called federated learning (FL) has introduced by Google in 2016 to address privacy and data sufficiency challenges by training AI model across multiple distributed devices and sharing model parameters rather than raw data. In the context of IoMT, FL can be used for building robust models with keeping patients data localized [4]. However, although FL has gained growing attention, it has also increased concerns about its security. For example, while FL training approach is different from the traditional training by preserving data localized which increasing data privacy, the nature of this approach makes the poisoning attacks easier, where there is no clear indication whether local data is legitimate or model parameters have been poisoned by malicious clients. Moreover, the non-IID (non-independent and identically distributed) nature of FL exploited by attacker to complicate poisoning attacks, making them harder to be detected [4]. Furthermore, due to frequent updatesin FL, the high cost of communication and computation remains a critical challenge, especially in large networks, which hinders scalability in real-world applications [6].

Nevertheless, while many studies have proposed valuable contributions on data poisoning attacks, but they remain unsatisfactory in the face of data nature, which reduce effectiveness in real-world applications [7]. Additionally, many of these studies primarily focus on identifying malicious clients during server-side aggregation, while the defense against such attacks at the client-side remains a challenge with limited investigation [8]. In this paper, we propose a FedSecure, a real-time adaptive anomaly detection framework for the client side to mitigate several poisoning attacks using real-world datasets obtained from the MIMIC III [9]–[11]. Our proposal

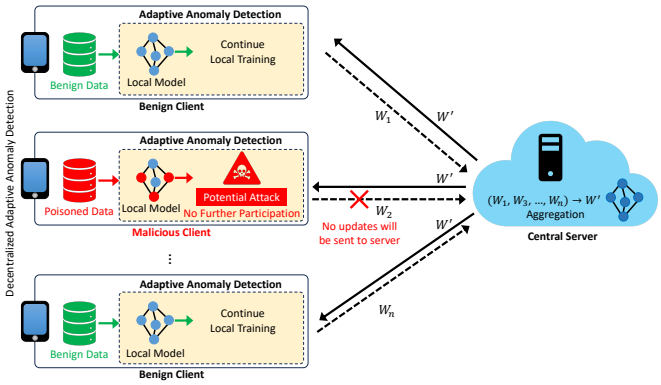


Fig. 1: Proposed FedSecure (decentralized anomaly detection)

is depicted in Fig. 1. where the poisoned data can be detected on the client side. As a result, only benign devices can be participated, ensuring that only trusted model updates are sent to the server. The proposed FedSecure not only enhance the security of FL process, but also contributes in reducing the overall load on both communication and computation overhead by filtering malicious early in FL process.

The paper is organized as follows: Section 2 introduces the related work on poisoning attacks in federated learning, while Section 3 presents the contributions of this paper. The proposed FedSecure is introduced in Section 4, and the hybird DL models used in our contribution are detailed in Section 5. The real-world dataset is described in Section 6. Section 7 presents the training process and model evaluation. Experimental results and discussion are provided in Section 8, while the conclusion and future work presented in Section 9.

II. Related Work on Poisoning Attacks

The FL security in IoMT environments has gain substantial attention, especially in addressing vulnerabilities to data poisoning as the FL nature enables attackers to make sophisticated attacks, leading to complicated detection and mitigation efforts. In recent years, there are several studies proposed different stratagies to mitigate such threats by implementing anomaly detection. These proposals have made significant advancements, particularly in IoMT where data privacy is a primary concern.

Blockchain have been incorporated in several recent studies to enhance FL security and mitigating poisoning attacks. For instance, Begum et al. [12] proposed a blockchain-driven federated learning-based intrusion detection system (BFLIDS) which secures model updates by integrating a Convolutional Neural Network (CNN) and Bi-LSTM models with a distributed ledger. Likewise, Kalapaaking et al. [13] integrated blockchain with secure multi-party computation (SMPC) for securing model updates with focusing on anomaly detection to mitigate poisoning attacks in healthcare systems. Zeng et al. [14] proposed a two-stage federated learning framework using blockchain to address non-IID data challenges in IoMT environment, enhancing client-side anomaly detection using supervised models by filtering malicious updates. Conversely, the proposed FedSecure design incorporate a distributed approach of adaptive anomaly detection, using client-specific thresholds. This design reduces the blockchain-latency, enhancing scalability via a decentralized detection process, supporting real-time IoMT applications.

In another approach, a privacy-preserving intrusion detection system (IDS) proposed for IoMT by Torre et al. [15] employing FL with CNN to secure model updates and differential privacy techniques fro enhancing data confidentiality. The proposed model in this study performs client-side anomaly detection for different poisoning attacks. However, this approach is effective for image-based data, which limits its applicability to diverse data types of IoMT. In addition, while the CNNs are effective for identifying patterns, they are less suited for anomaly detection, especially for sequential health data that has unusual or unexpected variations. These limitations are addressed by the proposed FedSecure using Bi-LSTM autoencoder integrated with a DNN classifier which more suitable for temporal and high-dimensional nature of health data, enabling more accurately anomaly detection even with non-IID data.

Another study by Sarkar et al. [16] introduced a federated learning framework employing artificial neural networks (ANNs) to address unauthorized device intrusion in IoMT using synchronized anomaly detection. Despite the novelty of this approach, it lacks adaptive thresholds, which increases false-positive rates as the device-specific data variations is not handled. This limitation is overcomed by the proposed FedSecure, where adaptive threshold is used for each client's data distributions, and which provides more resilient detection mechanism, reducing influence of benign data fluctuations.

A centralized anomaly detection method proposed by Manzoor et al. [17] to isolate malicious clients in FL for enhancing the accuracy of global model. Clients can be evaluated by mean absolute percentage error (MAPE) using Euclidean distances and Hidden Markov Models (HMM). However, FedClamp's does not handle the non-IID data which reduces the defense against poisoning attacks, and it relies on centralized detection which leads to communication and scalability issues, especially in lager FL networks. In contrast, the proposed FedSecure addresses these limitations through adaptive decentralized anomaly detection where malicious client can be detected independently using client-specific thresholds based on each client's historical data distribution. FedSecure not only mitigates poisoning attacks early, but also reduces communication load by allowing only benign clients to be participated in FL process which enhances efficiency and adaptability to the nature of IoMT data.

In summary, many of existing studies primarily rely on blockchain integration, supervised approches, or fixed and centralized anomaly detection. However, many challenges can be faced by these approaches affecting the computational overhead, communication load, real-time detection especially with larger FL networks. Moreover, handling non-IID data remains a challenge. These limitations are addressed by the proposed FedSecure using a flexible and decentralized adaptive anomaly

detection integrating a Bi-LSTM autoencoder as a reference model and DNN classifier, where the detection process can be preformed in a distributed manner and at an early stage of FL process. In addition, FedSecure is designed to accommodate heterogeneous and non-IID nature of IoMT data, which enhances which enhances the denfense against poisoning attacks. These capabilities make the proposed FedSecure a practical and scalable solution for real-worl healthcare applications.

III. Proposed Contribution

The security of federated learning in IoMT systems has significant challenges due to vulnerabilities, especially regarding data poisoning attacks. Traditional approches which rely on centralized detection methods or that blockchain-based methods hindered by some security issues, such as high computational overhead, scalability, communication latency, handling diverse. Our proposal, FedSecure, introduces an adaptive, decentralized anomaly detection framework that directly addresses these limitations through a novel client-specific approach to anomaly detection, which enhances the robustness of FL in IoMT applications. Our main contributions are highlighted as follows:

- Adaptive and Decentralized Anomaly Detection: Unlike previous related studies that employ centralized detection or blockchain-based latency-prone frameworks, the proposed FedSecure integrates adaptive and decentralized anomaly detection using a Bi-LSTM autoencoder and a DNN classifier. By this approach, the anomaly can be detected independently on each client device using client-specific thresholds. Thus, reducing the reliance on centralized processing or blockchain consensus which reduces communication overhead and distirbuting computational overhead to enhance the scalability.
- Adaptive and Decentralized Anomaly Detection: Unlike previous related studies that employ centralized detection or blockchain-based latency-prone frameworks, the proposed FedSecure integrates adaptive and decentralized anomaly detection using a Bi-LSTM autoencoder and a DNN classifier. By this approach, the anomaly can be detected independently on each client device using client-specific thresholds. Thus, reducing the reliance on centralized processing or blockchain consensus which reduces communication overhead and distirbuting computational overhead to enhance the scalability.
- Handeling Non-IID Data: Existing methods, such as those of Torre et al. and Sarkar et al. struggle with data diversity, especially with vary significantly data distributions across different devices, which increases false-positive rates or compromised the accuracy of detection process. FedSecure overcomes these limitations by implementing a client-adaptive thresholding mechanism based on the historical statistics of anomaly scores, which reduces the false positives and enhance the detetcion accuracy which makes FedScure more flexible to the non-IID nature of IoMT data.
- Early-stage Anomaly Detection: A signaficant limitation of centralized detection approaches, such as those proposed by Manzoor et al., is that detecting malicious updates is delayed after aggregation process which adversely affect the global model. FedSecure addresses this limitation by detecting anomly in early-stage at the client-side, where the malicious clients are isolated before their updates sent to the cetral server, and only benign clients participate in the FL process. This will not only enhance the security of FL, but also reduces the unnecessary communication by excluding compromised devices early.
- Flexibility for Diverse and Complex Health Data: While many previous existing studies focus on structured or image data, FedSecure is optimized for high-dimensional, sequential nature of IoMT data, modeling complex temporal patterns within health data accurately. This is critical for anomaly detection that characterize poisoning attacks.

In summary, FedSecure contributes to a scalable and adaptable solution for enhancing FL security in IoMT environment with contributing in decentralized, client-specific detection, early-stage intervention, and handleing non-IID data. These contributions overcome the critical limitations, such as scalability, adaptability, and real-time detection.

IV. Proposed FedSecure

The proposed FedSecure introduces a robust decentralized adaptive anomaly detection framwork designed for mitigating poisoning attacks in FL environment within IoMT. The FedSecure design integrates deep learning models (Bi-LSTM autoencoder and DNN classifier) and employ them for anomaly detection to prevent the manipulated data from participating in the federated learning updates in early-stage. One of the IoMT challenges that addressed by FedSecure is handling the diverse and non-IID data, which is collected from different local medical devices.

For establishing baseline models, FedSecure trains the reference model (Bi-LSTM) on cleaned real-world data, where the DNN model is employed to classify the recontruction errors produced by the reference model (Bi-LSTM) and produce anomaly scores. After training process, these integrated models are distributed to client devices to be used in local training. FedSecure focuses on adaptive anomaly detection to mitigate poisoning attacks before any model update is sent to the central server, which reduces the communication overload by allowing only for benign local device to send model updates.

Fig. 2 illustrates the client-side workflow, where each local device data is reshaped into overlapping sequences with a specific sequence length (e.g., 60). The sequence structure is primary to capture the temporal patterns, especially in healthcare data where the value in a given time is typically depends on its previous values. This temporal relationships is preserved by when the data is reshaped into overlapping sequences, which helps the model to learn medical patterns. As depicted in Algorithm 1, after reshaping data, the pre-trained reference model (Bi-LSTM) reconstructs each input sequence

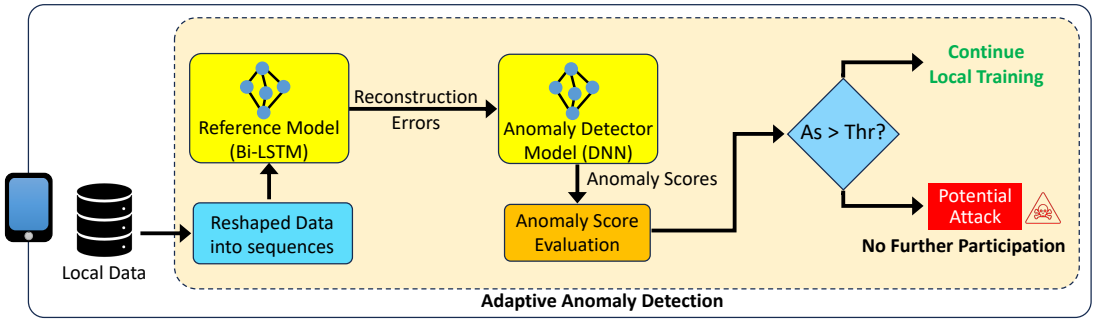


Fig. 2: Workflow of the proposed FedSecure

locally. Then, the model will produce the reconstruction errors by comparing each output to the original input. These errors measure how accurately the natural patterns are captured by the model, where the larger errors indicate large deviation from the expected patterns that the model learned. In other word, the reconstruction errors highlight the deviations from the excpected patterns by capturing the sequential dependencies in the medical data which is a critical issue in IoMT where physiological signals make it hard to distinguash between true anomalies from false positives.

The pre-trained DNN classifier further classifies the produced errors by the reference model and produces an anomaly score for each sequence. The responsibility of the DNN classifier is to refine the produced errors into anomaly score distribution tailored to each device by assigning higher probabilities to scores related to potentially poisoned behavior. The DNN classifier plays an important role for adjusting to individual device characteristics, where it produces anomaly scores that reflect the patient-specific variations. This is very important especially in federated learning which has diverse and non-IID data sources. Due to the meaningful anomaly scores produced by DNN classifier, the threshold will be effective to distinguash between benign and poisoned devices based on anomaly profile of each device. Once the anomaly scores calculated, they will be evaluated to set threshold based on their statistical distribution to detect anomalies.

In our propsal, the accuracy of detecting malicious client who manipulate data in federated learning is critical issue to mitigate potential poisoning attacks, especially in IoMT. Therefore, we designed a robust, adaptive, and dynamic device-specific threshold mechanism for detecting different types of data poisoning. Threshold mechanism adapts to local device's data distribution dynamically by analyzing and evaluting the historical statistical metrics of anomaly scores (e.g., mean, standard deviation, etc.) as shown in Fig. 3. allowing to make detection process personalized to individual variations. The statistical metric for each local device includes the mean score, standard deviation, and the difference score which is the absoulate difference between the first and the last scores in a sequence.

Once threshold is set for each local device, all local sequence scores for that device are compared against its specified threshold. If a sequence score exceeds the threshold, the device is flagged as a potential poisoning attack. In this case, the flagged local device will be classified as an abnormal device and excluded from any further model updates or communications with central server, which enhancing the security of federated learning and mitigating poisoning attacks.

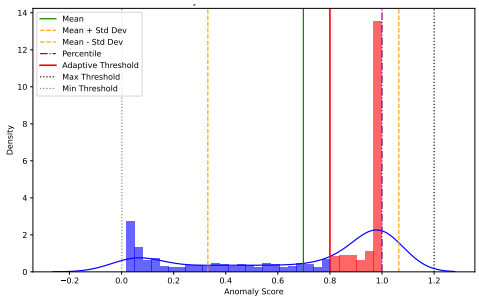


Fig. 3: Adaptive Threshold

After malicious clients are excluded through adaptive anomaly detection, only normal clients performs local updates on the local model based on their clean data. The normal devices then send their model updates to the central server ensuring that aggregation process is only performed on normal and reliable updates. Thereby, securing the process of federated learning without heavy communication or computation overhead. With assuming secure communication channels, this aggregation process preserve the integrity of the federated learning process by isolating the compromised data from affecting the global model. On the other hand, maintaining efficient communication and computational demands and ensure scalability.

V. Reference Model (Bi-LSTM) and DNN Classifier

A. Reference Model (Bi-LSTM)

In our proposed FedSecure, A Bi-LSTM (Bidirectional Long Short-Term Memory) model is used for anomaly detection with higher accuracy compared to traditional LSTM models, particularly in sequential data, such as time series data, where the data can be processed in both directions (forward and backward) [18]. The bidirectional approach can capture

Algorithm 1 Adaptive Anomaly Detection in FedSecure

Require: Client data X_i, Sequence length n, Trained Bi-LSTM model M_{BiLSTM}, Trained DNN classifier M_{DNN}, Anomaly threshold τ_i

Ensure: Anomaly flags A_i for client i

Reshape client data X_i into overlapping sequences $S_i = \{s_1, s_2, \ldots, s_k\}$ of length n
for each sequence $s_j \in S_i$ **do**
  Use Bi-LSTM model M_{BiLSTM} to reconstruct s_j, producing $\hat{s}_j$
  Compute reconstruction error $E_j = \|s_j - \hat{s}_j\|^2$
  Pass E_j through DNN model M_{DNN} to obtain anomaly score a_j
  if $a_j > \tau_i$ **then**
    Flag s_j as an anomaly: set $A_j = 1$
  else
    Mark s_j as normal: set $A_j = 0$
  end if
end for
Aggregate anomaly flags $A_i = \{A_1, A_2, \ldots, A_k\}$
if fraction of flagged sequences in A_i exceeds client threshold T_i **then**
  Exclude client i from model updates
end if

more complex patterns within sequential data more effectively, considering past and future contexts simultaneously [19]. This is essntial in the healthcare data as it has temporal patterns relationships, where the data at a given time is related to previous data. In the context of medical data, this advantage is more important in anomaly detection as we have different distributions for different patients, where the anomalies of each patient's data can be affected by events that occurred in both histories (past and future time steps) of the same patient. For example, when the system detects a sudden spike in heart rate, this spike will be better understood by the system as it considers the data points before and after this spike, which helps to distinguish between normal and abnormal patterns. This is very important in IoMT, where medical data has long dependencies, such as delayed effects of medications or cumulative effects of small vital sign fluctuations, leading to proper and timely interventions.

Furthermore, due to the nature of federated learning where the model can be trained collaboratively from different sorurces, the ability of Bi-LSTM model to maintain higher accuracy across diverse potentially poisoned data is critical. The relationship between temporal pattern is preserved due to the model capacity of analyzing sequences in both directions, which helps to detect anomalous and mitigate poisoning attacks that degrade the global model

Technically, before the data is input to the Bi-LSTM model, the raw time series data is reshaped into overlapping sequences as illustrated in Fig. 5 with a specific length representing time windows. The sequences of time steps are represented as $X = \{x_1, x_2, \ldots, x_T\}$, where x_t denotes the input at time

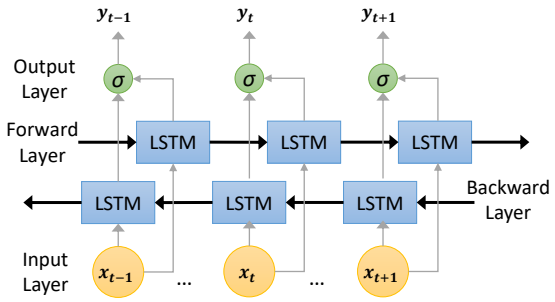


Fig. 4: An unfolded architecture of Bi-LSTM.

Fig. 5: Overlapping Sequences for Bi-LSTM

t, while T represents the sequence length. These sequences are processed in both forward and backward directions using two LSTM layers, as depicted in Fig. 4. Each sequence is processed from the past to future using the forward layer, while the backward layer processes the sequence in a reverse direction, from future to past.

The forward layer produces a hidden state $h_t^{\rightarrow}$ at each time step t. This hidden state is calculated based on the current input x_t and the previous hidden state $h_{t-1}^{\rightarrow}$:

$$h_t^{\rightarrow} = \text{LSTM}_{\text{forward}}(x_t, h_{t-1}^{\rightarrow}). \tag{1}$$

At the same time, the backward layer processes the sequence in reverse directions, producing a hidden state $h_t^{\leftarrow}$ for future data points relative to each x_t:

$$h_t^{\leftarrow} = \text{LSTM}_{\text{backward}}(x_t, h_{t+1}^{\leftarrow}). \tag{2}$$

Since healthcare data patterns can be impacted by events in both directions, they can be recognized effectively using such a bidirectional structure. A combined hidden state h_t at each time step t will be formed by concatenating both the forward and backward layers:

$$h_t = \begin{pmatrix} h_t^{\rightarrow} \\ h_t^{\leftarrow} \end{pmatrix}. \tag{3}$$

The combined hidden state h_t is passed to the output layer with a rich temporal representation of each time step from both directions to be used in further processing. In the output layer, the combined hidden state h_t will be transformed to produce the output sequence $Y = \{y_1, y_2, \ldots, y_T\}$, where each y_t represents the output at time step t:

$$y_t = f(h_t), \tag{4}$$

where f refers to a transformation, such as a dense layer, that projects h_t to the output space. For anomaly detection, this output y_t may represent the reconstructed form of x_t, which will be compared to the original input to compute reconstruction errors. For each time step t, the reconstruction error is calculated as the difference between the original input x_t and the reconstructed output y_t:

$$\text{Error}_t = \|x_t - y_t\|^2. \tag{5}$$

The reconstruction error Error_t is used to measure how the Bi-LSTM model captures the expected patterns in healthcare data accurately. Higher reconstruction errors indicate potential anomalies.

B. DNN Classifier

The DNN classifier is employed in our proposal as essntial component used for adaptive anomaly detection, where it analyzes the reconstruction errors produced by Bi-LSTM model and classifies them into anomaly scores as depicted in Fig. 6. This helps to identify potential poisoning attacks or abnormal behavior in the local devices data and excluding the malicious clients from any further participations in the federated learning process.

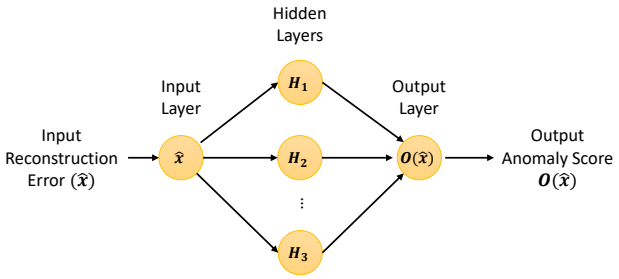


Fig. 6: DNN Classifier

Since healthcare data has complex patterns and each local device has different distribution patterns, the DNN classifier has the ability to capture such client-specific patterns within reconstruction errors using multiple hidden layers, where normal and anomalous variations can be distinguished. This is because the DNN is trained on reconstruction errors of individual client data, which adapts the model to specific properties of the local device, helping handle non-IID data in federated learning, and reducing false positives and negatives in anomaly detection.

Each reconstruction error $\hat{x}$ of a sequence is processed by the DNN classifier to produce anomaly scores $O(\hat{x})$ as shown in Fig. 6. Each error determines the deviation between the original input sequence x and its reconstructed error $\tilde{x}$, which is computed using the Mean Squared Error (MSE):

$$\hat{x} = \frac{1}{m}\sum_{i=1}^{m}(x_i - \tilde{x}_i)^2,$$

where m indicates the number of features in each sequence. The error value is input through the input layer, then is passed through hidden layers, where a non-linear ReLU activation function is applied by each layer H_j to capture and refine the input features. In each layer, the ReLU activation is defined as:

$$h_j = \max(0, W_j h_{j-1} + b_j),$$

where W_j and b_j indicate the weights and biases for layer j, respectively. Through the output layer, a single anomaly score is produced for each sequence, represented by $O(\hat{x})$, which indicates the anomalous nature of each sequence. The Sigmoid activation function is used to compute the anomaly score as follows:

$$O(\hat{x}) = \sigma(W_o h_n + b_o) = \frac{1}{1 + e^{-(W_o h_n + b_o)}},$$

where W_o and b_o indicate the weights and bias of the output layer, and h_n represents the output from the last hidden layer. The output $O(\hat{x})$ is interpreted as a binary classification: $O(\hat{x}) \approx 0$ for normal sequences, and $O(\hat{x}) \approx 1$ for anomalous sequences.

VI. Dataset and Data Processing

A. Dataset Description

In our proposal FedSecure, we utilized the MIMIC-III dataset which is widely used critical care units database. For privacy, the database contains de-identified health-related data for thousands of patients. It provides different vital signs and lab measurements recorded at regular interval times, which is essential for implementing anomaly detection in Internet of Medical Things (IoMT) context. The primary goal of using such dataset is to simulate the realistic conditions of comprehensive healthcare data which reflects the variability in real-world IoMT systems. We extracted continous time series data from five patients containing 50,401 instances to manage the extensive dataset and to capture diverse patient characteristics while maintaining computational time of our experiment.

B. Data Preprocessing

Since the healthcare data is non-IID where each patient may have different distribution from other ptients due to various factors such as, age, current body energy, and medical conditions. As a result, to maintain the individual characteristics of health data in IoMT environment and to support the requirements of federated learning where each individual patient operates independently, each patient's data is preprocessed separately. The features extracted and utilized in our experiments include heart rate and respiratory rate, both are recorded at 1 minute interval. New temporal features were extracted from the timestamp, such as hour, day, and

weekday. Each local device in our implementation of federated learning is related to unique patient in the dataset, where all data preprocessing steps are performed based on the IoMT requirements to simulate a real-world patient monitoring.

VII. Training Process and Model Evaluation

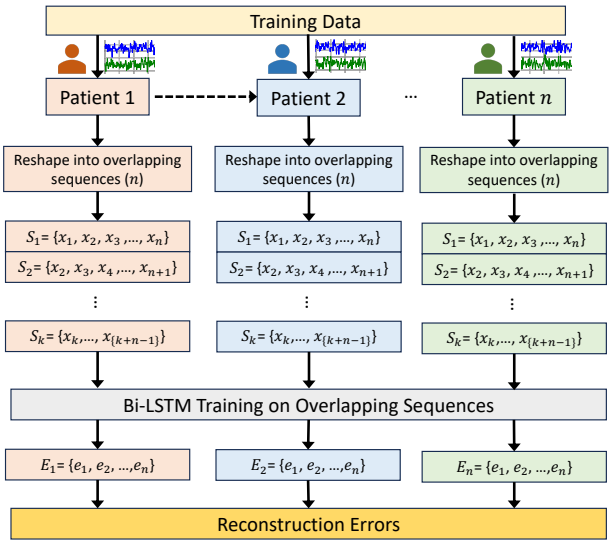


Fig. 7: Training Process of Bi-LSTM on Overlapping Sequences.

A. Training Process

Once data is preprocessed, each patient data is prepared for Bi-LSTM training by reshaping the data into overlapping sequences with specific length n as illustrated in Fig. 7. where each sequence includes a set of feature vectors, $S_i = \{x_1, x_2, \ldots, x_n\}$, where each x_j is related to a specific data point within the sequence. Using such overlapping sequences, the model can capture and recognize transitions in patterns over time. Once data is reshaped, each patient's sequences are input into the Bi-LSTM model for training. Then, these sequences are processed by the model in a bidirectional manner to analyze both past and future contexts. During the training process, a *reconstruction error* is calculated by the model for each sequence as $E_i = \{e_1, e_2, \ldots, e_n\}$ for each patient i, used as a measure of the model's ability in reconstructing normal patterns. Finally, the reconstruction errors for each patient are analyzed to establish anomaly detection thresholds. Performing individualized training and error analysis enables the model to capture patient-specific patterns and enhances personalized anomaly detection in federated learning.

In training DNN classifier, to avoid potential bias toward normal patterns and enables the DNN to reliably detect anomalous sequences, we ensured that training reconstruction errors have sufficient sequence anomalies. Based on the distirbution of training reconstruction errors, we applied patient-specific patterns threshold to reflect the characteristics of each patient rather than using a global threshold for all patients. Using such threshold, the false positives or negatives will be reduced, which might be increased if the variations of each patient not considered. Threshold is set by integrating the statistical (mean and standard deviation) and percentile-based thresholds. By this combination, the sensitivity of detecting sequence anomalies and avoiding false positives from slight variations in normal data is balanced. In this approach, threshold can be adapted to the normal fluctuations for of each patient and avoiding normal variations to be incorrectly classified as anomalies. The threshold defines the boundary based on the expected distribution of reconstruction errors. The errors that exceed threhsold are considered as anomalies. This provides a flexible adjustements without changing the true anomaly rate. When the threshold is set carefully, the unexpected deviations will be isolated.

B. Model Evaluation

1) Bi-LSTM Model Evaluation: The performance of Bi-LSTM and DNN classifier models is examined within several several metrics and visualizations. Fig. 8. indicates the density of reconstruction errors for training dataset which shows a high concentration of low error values. This confirm that Bi-LSTM model was trained on mostly clean data. In addition, the low errors indicates that the model was able to capture normal patterns, which is very important to distinguish between normal and abnormal behavior.

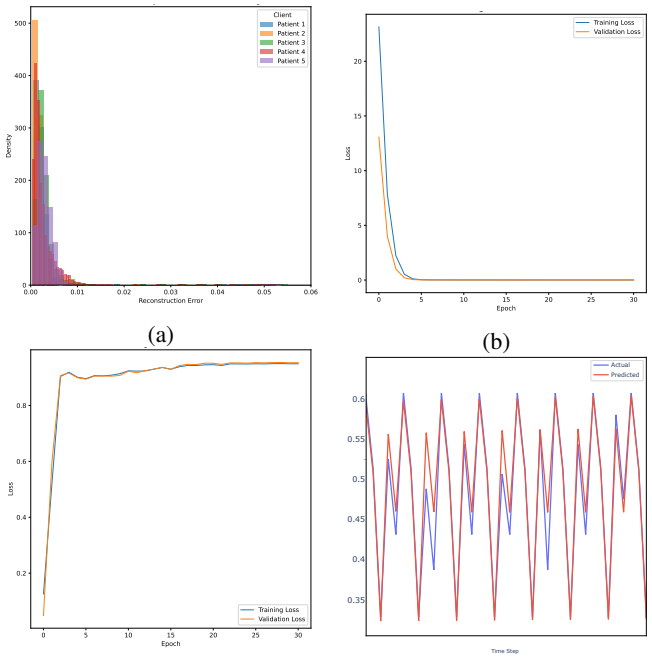


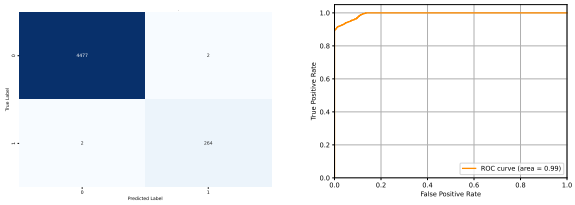
Fig. 8: Bi-LSTM Evaluation

The progression of training and validation losses are depicted in Fig. 8 (b). The initial sharp drop indicates the fast learning, where the Bi-LSTM model reconstruct normal sequences quickly with low errors. At training progresses, both losses converge towards each other and stabilize at low values over 31 epochs. This indicates that the model is

well-generalized with no overfitting. The Bi-LSTM accuracy is tracked as shown in, Fig. 8 (c). The training accuracy stabilizing at 95.0%, while the validation accuracy reaching 95.4%. This indicates the consistent learning of the model, where the close alignment between training and validation accuracy indicates that the model is generalized well to new data, which is important factor for anomaly detection in a federated IoMT environment.

Finally, the ability of the reference model for reconstructing temporal sequences is shown in Fig. 8 (d), which illustrates a sample of the actual and predicted values for the test data. The close alighment between the actual and reconstructed sequences, especially the signaficant trends and fluctuations shows theat the model effectively can retain the temporal dependencies within sequences which is essntial feature for identifying the deviations from learned normal patterns. Consequently, the success of the reference model in reconstructing expected sequences patterns indicates that anomalies represented by larger reconstruction errors can be detected reliably. This enhances the proposed FedSecure's ability to mitigate poisoning attacks effectively in a federated learning environment.

2) DNN CLassifier Evaluation: In evaluating the DNN classifier, the model shown high accuracy and reliability in distinguishing between normal and anomaly sequences as depicted in confusion matrix, ROC curve, and cross-validation metrics. The Confusion Matrix shown in Fig. 9 (a) shows the ability of the model classification and performance, where there are 4477 true negatives, 264 true positives, with low error rates of misclassification. The ROC curve, shown in Fig. 9 (b), with an AUC of 0.99, demonstrates the ability of the DNN classifier for distinguash between normal and anomaly sequences using different thresholds specified based on the distribution of reconstruction errors for each patient.



(a) (b)

Fig. 9: DNN Classifier Evaluation

Fold	Train Acc.	Train Loss	Val Acc.	Val Loss
1	0.9953	0.0118	0.9998	0.0065
2	0.9972	0.0070	0.9996	0.0013
3	0.9969	0.0088	0.9993	0.0034
4	0.9982	0.0061	0.9993	0.0038
5	0.9971	0.0067	0.9996	0.0018

TABLE I: Cross-validation results for the DNN Classifier.

To ensure the robustness and generalization performance of the DNN classifer, it was evaluated using a five-fold cross-validation. The model achieved consistently high train and validation accuracy across all folds as shown in Table I. The train accuracy ranged from 99.53% to 99.71%, while the validation accuracy ranged from 99.98% to 99.96%. This indicates that the model effectively generalizes to unseen data. The low losses of train and validation in each fold indicates that the pattern of reconstruction errors were effectively captured by the model without overfitting.

Overall, these results indicate that the DNN classifier can make accurate predictions even with threshold change, which reflects the effectiveness of the proposed FedSecure framework in detecting anomalies and malicious behavior within the IoMT environment in federated learning. This approach mitigates poisoning attacks by reliably excluding poisoned data early-stage in the federated learning process, achieving high reliability with minimal communication overhead.

VIII. Experimental Results and Discussion

In this section, we present and analyze our experimental results and highlight the impact of various poisoning attacks on local and global models. The results are compared against the baseline and our proposed FedSecure approach. These experiments were implemented using Python programming language and TensorFlow for deep learning model, and were conducted on a MacBook Pro.

As shown in Table II, we conducted six distinct experimental runs for evaluating the impact of various poisoning attacks and the performance of our proposed FedSecure. For consistent comparison, each run has 30 iterations with a maximum of n=5 local devices except run 6 where the isolation mechanism in our FedSecure was activated and the poisoned device was successfully detected and isolated, reflecting the effectiveness of our proposal in mitigating poisoning attacks within the federated learning environment for non-IID data in IoMT.

TABLE II: Summary of Experimental Runs

Run	Description
Run 1	Baseline (no poisoning) to establish standard model performance.
Run 2	Data Poisoning Attack 1, introducing variability in data patterns.
Run 3	Data Poisoning Attack 2, applying pattern scaling to alter data distributions.
Run 4	Backdoor Attack 1, using gradual pattern injection to mimic slight malicious behavior.
Run 5	Backdoor Attack 2, introducing repeated patterns to create a malicious signal.
Run 6	FedSecure, demonstrating our proposed defense mechanism.

A. Local Model

For local models, the comparision between Dev2 (Poisoned Device) and Dev4 (un-poisoned) provides how they impacted by different poisoning scenarios. For Dev2, the baseline in Run 1 shows a gradual reduction in loss and regular improvement in

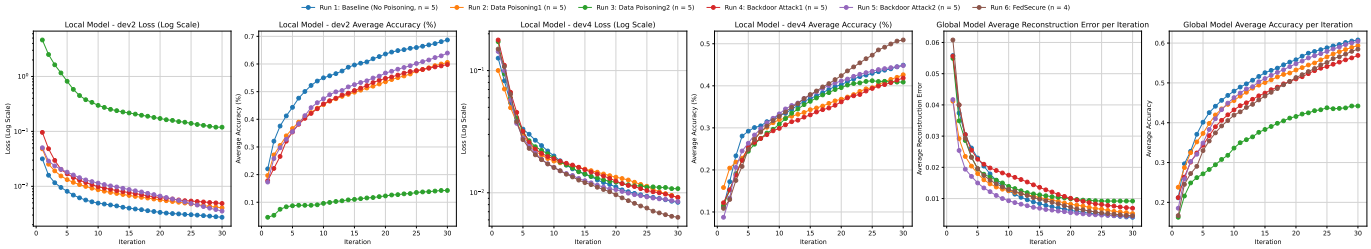


Fig. 10: Loss and Accuracy per Local Device and Run.

accuracy over iterations. In poisoned runs (Runs 2–5), shows slower loss and accuracy reduction, especialy in Run3 and Run 4. This indicates that poisoning attacks prevent Dev2 from gaining similar level of performance, where Run 3 shows most impact. In contrast, FedSecure in Run 6 shows stable loss and accurate trends which are close to the baseline. This means that FedSecure successfully mitigated poisoning effects.

For Dev2, in Run 1 (Baseline with no poisoning), the local model shows expected behavior as it is trained on clean data over 30 iterations, with reasonable continuous improvements in both loss and accuracy which indicates that the local model learning effectively. In Run 2 (Data Poisoning1), similar to Run1 where the loss and accuracy improved over iterations, but with a noticeable increase in loss and slower accuracy increasing copared to the baseline. This indicates that poisoned data degraded the model ability to reach the baseline. The performance of local model in Run 3 (Data Poisoning2) hs more sever affect. Comparing to the baseline, the loss is extremely high where accuracy remains lower than the majority of training process. This indicates that the Data Poisoning2 has more affect the learning process of the model compared to the Data Poisoning1, where the model struggles to make expected predictions even with 30 iterations. The performance of local model is impacted in Run 4 (Backdoor Attack1) but less than in Data Poisoning2. Despite this, the model loss decreses over time and the accuracy improved gradually indecating that the model recovers the initial disruption caused by the backdoor attack. Although the model still able to improve predictions over iterations similar to Run 2 with data poisoning, the accuracy still lower than the baseline, which indicates that backdoor attack has continuous affect on the model. In Run 5 where Backdoor Attack2 is applied, the poisoning is slightly more sever. The high loss and low accuracy in the initial iterations are improved over time. Similar to Run 4 where the model recovers the disruption, but with longer time fro accuracy to be in a reasonable value. However, the accuracy still lower than the baseline performance, which indicates that Backdoor Attack2 is more adverse than Backdoor Attack1.

For Dev4, in Run1 where no poisoning is applied, the local model shows gradual improvement 30 iterations. The loss and accuracy showing in the figure incdicate that the local model is successfully learns and capturing important patterns. In Run 2 (Data Poisoning1), a slight reduction in model performance is observed, which is expected as the model was injected with poisoned data. Although the model loss decreased gradually, the accuracy lower than the baseline. Run 3 (Data Poisoning2) shows slower loss and accuracy reduction which indicates that this type of poisoning has more persistent negative effect on the model and harder to overcome than the first type. Run 4 (Backdoor Attack1) is similar to previous runs, but with notable fluctuations. The inintial loss is higher than Data Poisoning2 which indicates that the model learning ability is disrupted significantly by backdoor attack in the first iterations. Although the model recovers this in later iterations as observed in previous poisoning atatcks, the accuracy remains lower than the baseline. In Run 5, the local modal experienced with another type of backdoor attack, which cause different pattern. The initial accuracy much lower in this run, which indicates that this type of attack has more impact on the model ability to converge. The local model loss starts at a relatively high value, but it decreases over iterations. However, over the 30 iterations, the accuracy still lower than the baseline. The result indicates that backdoor attack impacted the model performance more than data poisoning scenarios. Finally, in Run 6, where the FedSecure is applied, we observe an interesting trend. At the initial iterations, local model loss and accuracy are similar of what observed in previous runs, but hte model exhibits a stronger improvement over later iterations exceeding all previous runs including the baseline. This indicates that FedSecure improved the model performance effectively, and successfully identifying and mitigating the effects of malicious data.

B. Global Model

The global model loss and accuracy depicted in Fig. illustrates the model robustness against poisoning attacks in different scenarios (Runs 1-5) and the FedSecure effectiveness in (Run 6). In the absence of poisoning (baseline- Run 1), the global model shows a typical behavior with stable, consistent improvement in both loss and accuracy over 30 iterations. The smooth reduction in loss and steady increase in accuracy reflect the generalization and ability of the model for making accurate predictions after each round. Runs 2 and 3 introduce data poisoning which causes a slower convergence and larger fluctuations as shown in Fig. where the initial iterations show higher losses and lower accuracies compared to the baseline. This indicates that poisoning degrades the learning rate and accuracy of the global model. Especially in the Run 3, where the global model struggling to reach the baseline performance levels and reaching a steady state earlier than Run 1. This

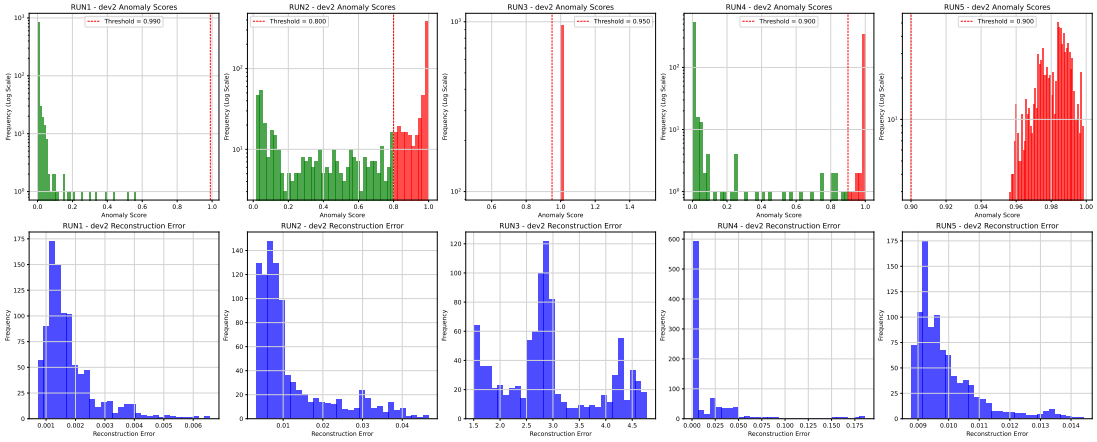


Fig. 11: Anomaly Scores and errors distribution of Malicious device (Dev2).

is might be due to the injected data anomalies. In (Runs 4 and 5), backdoor attacks are introduced, which cause more fluctuations with irregular patterns both in loss and accuracy similar to the data poisoning runs in. The initial loss in Run 5 slightly lower than Run4, but it increased over time. This reflect the negative impact of Backdoor Attack2 where it has a more persistent effect compared to Backdoor Attack1. The accuracy of both Run4 and Run5 improves over iterations, but still lower than the baseline performance. This pattern with the higher loss and a slower accuracy improvement indicates that backdoor attacks affect the global model, which could be due to the targeted nature of these attacks. In Run 6 (FedSecure), the global model shows a recovery similar to the baseline, which indicates the effectiveness of the FedSecure approach. The model loss shown lower than runs with poisoning ir backdoor attacks, where the accuracy increases steadily. However, in some cases, poisoned runs may show higher short-term accuracy by incorporating all data without filtering the malicious data and less data used in aggregation. Although this approach compromises long-term accuracy and increases vulnerability to malicious data, FedSecure isolates malicious contributions in early-stage and mitigates the poisoning attacks ot remain the global model resilient against poisoning attacks in the long term. However, ahough the accuracy is still slightly lower than the baseline. The performance of global model in Run 6 is notably better than in Run 3 (Data Poisoning2) and Run 5 (Backdoor Attack2) which indicates that the FedSecure's adaptive anomaly detection successfully reduced the impact of adversarial attacks.

C. Anomaly Scores Distribution

For anomaly scores distribution, in Run 1(No Poisoning - Baseline), the anomaly scores of dev2 are concentrated around very low values with slight deviations, where all the scores fall below the set threshold of 0.990, indicating that the data was identified as normal by the model. In Run 2 - Data Poisoning1, the threshold for anomaly detection is set at 0.800. The anomaly scores has a more spread-out distribution compared to Run 1. Although majority of the data falls in the normal range, this run shows increased variability in anomaly scores, where they scattered across multiple bins due to the poisoned data which contributes to outliers which being flagged as anomalies byt he FedSecure. This aligns with the observed loss and accuracy in Run 2, where the accuracy improved over iterations. In Run 3, the anomaly scores distribution is significantly higher compared to the previous run, where the most data fall in high range. This indicates that the poisoning has affected the data, which makes the most data to be flagged as anomalies. The accuracy of the model 44.20% at the end of iterations confirms that this type of strong poisoning has reduced the model performance, where it loses the ability to generalize to normal data with this posioning. In Run 4, the Backdoor Attack1, balaned the distribution of anomaly scores. The FedSecure detects some anomalies, at higher bins, but with less scores than in Run 3. This reflect the accuracy fo the global model where it is imperoved steadily in this run, which indicates that this type of attack caused moderate disruption. The result of anomaly scores shoes that backdoor data manipulations leading to partial detection of malicious. Although the model can resist against backdoor poisoning, but still can not overcome the attack. Run 5 - Backdoor Attack2, the a stronger backdoor effect can be noted, where the anomaly scores distribution fall above the specified adaptive threshold. This indicates that this type of attack has more persistent effect on the data. However, the accuracy of the global model in this run shows that the model has a slightly better recovery than in Run 4 which indicates that this type poisoning can be mitigated, but it might affect the model performance in long-term.

D. Reconstruction Errors Distribution

The reconstruction errors in Run 1 - No Poisoning (Baseline) has low values with few higher reconstruction errors due to natural variability. This reflects that model sucessfully captured the normal pattern and able to reconstruct the normal data in the absence of poisoning. The error distributions in the Run 2 - Data Poisoning1 aligns with the increased anomaly scores and matches the global model performance in Run 2.

This indicates that the poisoning has affected reconstruction. Run 3 - Data Poisoning2, the incresing in reconstruction error distribution can be noted. This consistent with the high anomaly scores, which indicates that the model struggling to reconstruct data correctly due to the aggressive poisoning. In addition, the accuracy of the model in this run shows this consclusion, where the model could not adapt well to this type of poisoning, leading to higher errors. Also, the reconstruction errors in Run 4 and Run 5 - Backdoor Attacks are increased, but with less drastic compared to Run 3. As the errors are gradually increased, the model still able to reconstruct much of data but disrupted by the backdoor patterns. The improvement of accuracy in Run 4 and Run 5 reflects this result.

IX. Conclusion and Fute Work

The FedSecure framework demonstrates how adaptive distributed anomaly detection contributes to mitigate poisoning attacks in federated IoMT environments against complex attacks using a reference model (Bi-LSTM) autoencoder and DNN classifier. The experimental results on real-world data (MIMIC III) shows that FedSecure approach can identify poisoning patterns and mantain the model integrity and accuracy. The FedSecure increases the communication efficiency and scalability of federated learning, where the malicious clients are isolated in early stage of federated learning process. Therefore, only benign updates are sent to the central server. Although FedSecure robustness against multiple poisoning attacks, some other types of attack will be included in our future work, such as label-flipping and model inversion attacks. In addition, our future work will include exploring lightweight models to enhance edge-computing optimizations.

References

[1] D. C. Nguyen, Q.-V. Pham, P. N. Pathirana, M. Ding, A. Seneviratne, Z. Lin, O. Dobre, and W.-J. Hwang, "Federated learning for smart healthcare: A survey," *ACM Computing Surveys (Csur)*, vol. 55, no. 3, pp. 1–37, 2022.

[2] Z. Liu, Z. Liu, and X. Yang, "Poisoning attack based on data feature selection in federated learning," in *2023 13th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. IEEE, 2023, pp. 106–110.

[3] M. Ali, F. Naeem, M. Tariq, and G. Kaddoum, "Federated learning for privacy preservation in smart healthcare systems: A comprehensive survey," *IEEE journal of biomedical and health informatics*, vol. 27, no. 2, pp. 778–789, 2022.

[4] A. Brecko, E. Kajati, J. Koziorek, and I. Zolotova, "Federated learning for edge computing: A survey," *Applied Sciences*, vol. 12, no. 18, p. 9124, 2022.

[5] N. Rodríguez-Barroso, D. Jiménez-López, M. V. Luzón, F. Herrera, and E. Martínez-Cámara, "Survey on federated learning threats: Concepts, taxonomy on attacks and defences, experimental study and challenges," *Information Fusion*, vol. 90, pp. 148–173, 2023.

[6] Z. Li, V. Sharma, and S. P. Mohanty, "Preserving data privacy via federated learning: Challenges and solutions," *IEEE Consumer Electronics Magazine*, vol. 9, no. 3, pp. 8–16, 2020.

[7] S. Siddiqi, F. Qureshi, S. Lindstaedt, and R. Kern, "Detecting outliers in non-iid data: A systematic literature review," *IEEE Access*, 2023.

[8] A. Mishra, S. Saha, S. Mishra, and P. Bagade, "A federated learning approach for smart healthcare systems," *CSI transactions on ICT*, vol. 11, no. 1, pp. 39–44, 2023.

[9] A. Johnson, T. Pollard, and R. Mark, "Mimic-iii clinical database," 2023.

[10] A. E. Johnson, T. J. Pollard, L. Shen, L.-w. H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, and R. G. Mark, "Mimic-iii, a freely accessible critical care database," *Scientific Data*, vol. 3, no. 1, pp. 1–9, 2016.

[11] A. Goldberger, L. Amaral, L. Glass, J. Hausdorff, P. Ivanov, R. Mark, J. Mietus, G. Moody, C. Peng, H. Stanley *et al.*, "Physionet: Components of a new research resource for complex physiologic signals. circ. 101 (23): e215-e220. circulation electronic pages. 2000. june 13."

[12] K. Begum, M. A. I. Mozumder, M.-I. Joo, and H.-C. Kim, "Bflids: Blockchain-driven federated learning for intrusion detection in iomt networks," *Sensors*, vol. 24, no. 14, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/14/4591

[13] A. P. Kalapaaking, I. Khalil, and X. Yi, "Blockchain-based federated learning with smpc model verification against poisoning attack for healthcare systems," *IEEE Transactions on Emerging Topics in Computing*, vol. 12, no. 1, pp. 269–280, 2024.

[14] Z. Lian, Q. Zeng, W. Wang, T. R. Gadekallu, and C. Su, "Blockchain-based two-stage federated learning with non-iid data in iomt system," *IEEE Transactions on Computational Social Systems*, vol. 10, no. 4, pp. 1701–1710, 2023.

[15] D. Torre, A. Chennamaneni, J. Jo, G. Vyas, and B. Sabrsula, "Towards enhancing privacy-preservation of a federated learning cnn intrusion detection system in iot: Method and empirical study," *ACM Trans. Softw. Eng. Methodol.*, Sep. 2024, just Accepted. [Online]. Available: https://doi.org/10.1145/3695998

[16] C. Zhong, A. Sarkar, S. Manna, M. Z. Khan, A. Noorwali, A. Das, and K. Chakraborty, "Federated learning-guided intrusion detection and neural key exchange for safeguarding patient data on the internet of medical things," *International Journal of Machine Learning and Cybernetics*, pp. 1–31, 2024.

[17] H. U. Manzoor, M. S. Khan, A. R. Khan, F. Ayaz, D. Flynn, M. A. Imran, and A. Zoha, "Fedclamp: An algorithm for identification of anomalous client in federated learning," in *2022 29th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE, 2022, pp. 1–4.

[18] S. Yang, "Research on network behavior anomaly analysis based on bidirectional lstm," in *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, 2019, pp. 798–802.

[19] Y. Fadili, Y. El Yamani, J. Kilani, N. El Kamoun, Y. Baddi, and F. Bensalah, "An enhancing timeseries anomaly detection using lstm and bi-lstm architectures," in *2024 11th International Conference on Wireless Networks and Mobile Communications (WINCOM)*, 2024, pp. 1–6.