










## Privacy-Preserving IoT Framework with Federated Learning and Lightweight NLP Integration

Kallinatha H D<sup>1</sup>, Suhas G K<sup>2</sup>, Chaithra M<sup>3\*</sup>, Shashank Dhananjaya<sup>4</sup>, Suhaas K P<sup>4</sup>, Bhat Geetalaxmi  
Jairam<sup>4</sup>, Sunitha R<sup>5</sup>

<sup>1</sup> Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumkur 572103, India

<sup>2</sup> Research Center for Computer and Information Science, Akshaya Institute of Technology, Affiliated to Visvesvaraya Technological University, Tumakuru 572106, India

<sup>3</sup> Department of Computer Science and Engineering, The National Institute of Engineering, Mysuru 570008, India

<sup>4</sup> Department of Information Science and Engineering, The National Institute of Engineering, Mysuru 570008, India

<sup>5</sup> Department of Artificial Intelligence and Machine Learning, BNM Institute of Technology, Bangalore 560070, India

Corresponding Author Email: [chaithram@nie.ac.in](mailto:chaithram@nie.ac.in)

Copyright: ©2025 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/jesa.580509>

### ABSTRACT

**Received:** 11 April 2025

**Revised:** 14 May 2025

**Accepted:** 22 May 2025

**Available online:** 31 May 2025

#### Keywords:

*Federated Learning, FedProx, Internet of Things, natural language processing, privacy, TinyBERT*

The explosive growth of the Internet of Things (IoT) has resulted in a plethora of data creation, some of which includes sensitive and private information. With the presence of smart devices everywhere, it realizes an urgent need for privacy-preserving intelligent decision-making systems in IoT environments. In this paper, presented a new framework called FLAIR-IoT (Federated Learning with Adaptive NLP Integration for Resilient IoT) that seamlessly incorporates Federated Learning (FL), lightweight NLP models, and privacy-based methods for achieving semantic understanding and context-aware intelligence at the edge. FLAIR-IoT uses small transformers TinyBERT for on-device NLP tasks like intent detection, sentiment analysis, and contextual inference. To achieve the user privacy and secure aggregation, the models on these edge nodes are jointly trained by FedProx, with user privacy preserved by differential privacy (DP) and secure aggregation. The experimental approach utilizes a heterogeneous IoT simulation, where six edge devices store non-IID datasets. With our DP and secure aggregation, we protect user privacy while our framework does on-device NLP tasks such as in the intent detection as well as context classification. Results in experiments: 97.3% intent detection accuracy, 89.7 F1 score and 2.3MB per round average communication cost, and convergence in 34 rounds huge improvement over traditional centralized and federated baselines FLAIR-IoT achieves the new state-of-the-art on privacy-preserving, intelligent IoT systems.

## 1. INTRODUCTION

The Internet of Things (IoT) is a network of connected devices that communicate and exchange data via the Internet from household appliances to industrial machinery. This has resulted in an unimaginable increase in data generation. Data generated by IoT devices, for example, rose from 0.1 zettabytes in 2013 to approximately 4.4 zettabytes by 2020. The continuous growth is due to the growing adoption of IoT in different domains such as healthcare, transportation, agriculture, and smart cities [1]. This data explosion has opportunities and challenges. It allows for better decision-making, operational effectiveness, and the creation of new services. However, it brings serious difficulties in data integration, storage management, and security. Latency issues and bandwidth limitations make traditional centralized data processing models often insufficient [2]. Hence, there is a trend towards decentralized methods like edge computing where data processing takes place nearer to the origin thereby,

minimizing latency and improving real-time processing capabilities.

Furthermore, the convergence of Big Data technologies with IoT has become significant. They provide infrastructure for processing potentially massive, heterogeneous data streams produced by IoT devices. But as the data begins to flow [3], it does open up questions over data privacy and security since later, when more such data is going to flow, sensitive data can also be compromised [4]. Natural Language Processing (NLP) [5] is a crucial component of reading unstructured data produced by humans and sensors in the IoT. This unstructured data, which describes text, speech, and sensor measurements, usually comes in a messy form unfit for a simple analytical approach. NLP methods offer a way to derive valuable information from this data, leading to better decision-making and user experiences. In smart healthcare, a large fraction of the data contained in Electronic Health Records (EHRs) is unstructured text, including clinician notes, patient histories, etc. Neural NLP techniques have been used

to get this data in greater detail and better enhance patient care and operational efficiency [6].

In a similar vein, NLP-based sentiment analysis is employed in consumer electronics to analyze streaming IoT data and adapt to customer emotions and actions in real time. This method incorporates multi-modal data from IoT, such as text, audio, and sensor readings, to obtain insights into customer satisfaction [7]. Moreover, there have been recent developments on bringing together LLMs and IoT sensor data, allowing these models to understand and reason over the reality. Such integration adds a completely new area of possible applications beyond plain text-based tasks. NLP enables organizations to structure unstructured data, perform entity recognition, topic modeling, text summarization, etc. This is a game changer for information retrieval, document categorization, and the decision-making process.

**Role of Centralized ML/DL Architectures in Smart System Building Leveraging Distributed Sources of Data** Centralized ML/DL architectures form the building block of intelligent systems by bringing together massive amounts of data from distributed sources. One drawback to this centralized approach is privacy, which is a major concern in areas like healthcare, smart homes, and industrial IoT. When sensitive data is stored in a centralized manner, that increases the risk of cyberattacks and unauthorized accesses. If a central server is compromised, then sensitive data becomes leaked to every entity. Because the data is centralized, it creates a single point of failure that makes the whole system a target for adversaries [8, 9].

Adversaries may use trained models to ascertain if a given record had been part of the training data. Such membership inference attacks compromise user confidentiality, and especially become concerning for users, when a model is exposed through an API or shared among clients [10]. With model inversion, the attacker manipulates the input features based on the model's outputs to create a model that reverse-engineers sensitive attributes, including health conditions or user identities [11, 12]. This is especially concerning when models are trained on medical or biometric data. Centralized datasets are susceptible to re-identification attacks, where anonymous data can be retraced to identify individuals when cross-validated with other information sources, even in the presence of anonymization techniques. In fact, without strong privacy-preserving mechanisms in place, anonymization is typically ineffective at protecting user confidentiality.

It also risks contravening international data protection legislation like the General Data Protection Regulation (GDPR) and California Consumer Privacy Act (CCPA), which stress data minimization, transparency, and user control [13, 14]. On the other hand, centralized systems often struggle with granular consent management and auditability needed for regulatory compliance. These challenges have further emphasized the need for distributed learning paradigms such as Federated Learning (FL), where the model is trained without needing to transfer raw data, preventing any leakage of sensitive data.

Exponential growth in the generation of data has occurred due to the proliferation of IoT devices, which often contain sensitive personal information. Centralized machine learning architectures feel compelled to use and require collecting this data in a centralized server, which poses major privacy and security issues. As such, FL provides a decentralized scenario with multiple devices cooperating to train a global model without the need to share the raw data. Under FL, devices process data locally, sending only model updates to a central

aggregator, who accumulates and updates the global model. Instead of gathering and centralizing user information, this is the concept of keeping the personal data on local devices, increasing privacy and reducing risks of data leaks. In addition, FL addresses issues related to bandwidth limitations by avoiding the need to transmit data [15].

NLP is heavily relied on in a growing number of smart environments like, for example, smart homes and healthcare systems to be able to understand and react to commands and/or questions from users. Incorporating NLP as part of these systems allows for more natural and human-like interactions. Yet working with natural language data often includes addressing highly sensitive information, deepening privacy concerns [16]. While there have been recent breakthroughs in resolving these issues with differential privacy (DP), secure aggregation and FL variants of personalized variants however these still leave a large gap integrating semantic-aware learning capabilities to FL pipeline, especially at the edge for e.g. NLP. Though tiny models like TinyBERT and DistilBERT are a good lightweight NLP model, other deployment on FL in general, and especially with strong privacy guarantees is less explored.

**Major Contributions and Novelty of the Paper:**

- We propose a novel framework delivering context-aware privacy-preserving intelligence in IoT-enabled smart environments by integration of FL with NLP. The main contributions are as follows:
- Federated NLP Architecture Development is constructed using the FL principle among distributed IoT devices in a way that ensures that no sensitive linguistic information is transmitted to the cloud.
- Integrating privacy-preserving methods including privacy-enhancing technologies like DP and secure aggregation to safeguard model updates during the exchange, guarding against exposure during the FL process.
- Optimization for Resource-Constrained Devices Adaptation of NLP algorithms to perform efficiently on IoT devices with limited computational power, feasibility and scalability in real life applications.

The aforementioned data analytic process FL and NLP in the context of IoT environments serve as a stepping stone in the progress of user-centric systems while meeting the standards for privacy maintenance.

The structure of this paper is organized as follows: In Section 2, we present a comprehensive review of state-of-the-art research in privacy preservation in IoT Section 3 summarizing some of the most relevant works and describing the existing approaches for combating attacks in the IoT space. Section 4 presents the detailed proposed model with mathematical model and pseudocode. Section 4 provides the empirical evaluation of the proposed detection scheme and shows the results. Further, section 5 presents a comprehensive comparative discussion between our results and other relevant modern research performed. Eventually, in Section 6, we conclude the result and suggest perspectives for future work; then, the references section follows.

## 2. RELATED WORK

Recently, very serious work has been done in combining NLP capabilities with IoT environments, primarily for improving voice recognition and text processing in sensor networks. In this section summarize prominent works from the

last five years with an outlook on improvements in NLP application in IoT context. Zhang et al. [17] presented a joint intent detection and slot filling model to enhance natural language understanding in Internet of Things (IoT) voice interactions. They found that their approach performed better than high-level techniques such as capsule networks to capture semantic relationships. Utilizing natural language understanding capabilities of NLP, Mihailescu et al. [18] proposed a multi-level distributed intelligent virtual sensor frame. The system allowed users to communicate with sensor networks using natural language, making it more user-friendly and accessible. Hanifa et al. [19] analyzed voice recognition in the IoT environment and its different approaches, and their applicability in resource constrained environments. Their work gave valuable insights into optimizing voice recognition systems for deployable and efficient implementations in the IoT networks.

Zhou et al. [20] investigated the relationship between language models and IoT sensors for zero-shot activity recognition. They presented TENT, a method that aligned textual embedding's with IoT sensor signals to allow the system to identify unseen activities by appending these activities with sensory data in complementary descriptive language. Ali [21] performed a comparative study on research work on Automatic Speech Recognition (ASR) in resource constraint wireless sensor networks. The architecture has been studied such as (NSR, DSR, and ESR) classification based on bandwidth utilization, processing power, and accuracy.

FL has become a crucial approach for collaborative machine learning on decentralized devices, especially for the Internet of Things (IoT) ecosystems. This makes it a particularly suitable approach of interest in edge and fog computing contexts, which typically entail strict privacy requirements over data, bandwidth restrictions, and limited computational abilities. Rajagopal et al. [22] have worked on FL on-device model training in edge computing, keeping data local, improving privacy, and reducing latency. One example is the integration of FL and edge computing, which has been proposed to deal with the challenges of IoT, highlighting that both efficient resource management and privacy preservation are needed. Likewise, an FL based on intermediate nodes between cloud and edge devices benefits from fog computing environments. Similar levels also enhance the scalability of IoT applications and optimize resource utilization.

There is recent work propose frameworks which can combine FL and edge or fog computing that can be used to improve the applications of IoT. An instance would be the FLIGHT framework have designed by Zhu et al. [23], it is a lightweight FL system that can be adopted on a wide range of devices, from resource-limited edges up to cloud GPUs. FLIGHT unifies synchronous and asynchronous FL models and proposes a heuristic-based worker selection mechanism to enhance training efficiency. Another remarkable approach is the cooperative FL paradigm that fully exploits device-to-device (D2D) by Wang et al. [24] interactions and device-to-server (D2S) interactions to alleviate the network heterogeneity in edge/fog networks. Using resource pooling mechanisms, this model results in significant gains in both model training quality and network resource use. Several integrative systems have been proposed to improve privacy, security, and efficiency in healthcare using Blockchain with FL in edge, fog, and cloud systems. Frameworks such as these, allow the multiple medical IoT devices to collaborate and train

global models without sharing raw data, thus overcoming privacy concerns that arise from centralized storage of data. Most importantly, they illustrate an evolution of FL use cases across edge and fog-based deployments, opening up future research possibilities that further emphasize security, sustainability, efficiency, and scalability in IoT systems.

There are two major privacy-preserving machine learning techniques that have gained detailed discussions, DP, and secure aggregation. Differential privacy Das and Mishra [25] is a mathematical construction that protects the privacy of an individual entry in a dataset by bounding the effect of a single data point on the outcome of any analysis, ensuring that the outcome does not appreciably change if an individual data point is added or removed. Within the realm of deep learning, DP is most commonly incorporated by adding appropriately scaled noise to gradients as part of model training. This method seeks to ensure that leaked outputs of a model cannot be used to infer sensitive information. Several recent surveys by Das and Mishra [25], for example, provide broad and up-to-date reviews of progress on DP and its applications in ML, covering theoretical results and applied endeavors alike.

Secure Aggregation by Xu et al. [26] is used primarily in FL settings, where several clients work together to train a model without draining their raw data. Secure aggregation protocols enable the central server to obtain only the aggregated model updates, but not the individual contributions, preserving the privacy of clients. But the common secure aggregation methods usually suffer the problems of communication overhead and the vulnerability of some inference attacks. The TAPFed framework, for example, proposes a threshold functional encryption scheme that significantly mitigates security vulnerabilities posed by malicious aggregators while minimizing the transmission overhead.

Gaps in the Current Literature:

- **Communication Overhead:** Most secure aggregation protocols come with a high communication penalty which increases with the increase in the number of clients and/or model parameters. To address this, works like FastSecAgg combine techniques like the Fast Fourier Transform to implement multi-secret sharing to ensure reduced computation costs by Ergun et al. [27].
- **Privacy Leakage Across Multi-Round Federation:** In FL, revolution after revolution does increase the systemic risk of privacy leakage. Tip: Traditional secure aggregation is not effective against privacy leakages on multiple training rounds. Multi-RoundSecAgg framework introduces structured selection strategies for users to protect their long-term privacy while participating in multiple training rounds by Kadhe et al. [28].
- **Scalability and Efficiency:** When translating DP to deep learning models, it is common to consider compromises between privacy promises and the utility of the model. Finding the tradeoff between strong privacy guarantees and model utility is still hard. Surveys on different DP mechanisms and its effect on model accuracy and training speed by So et al. [29].
- **Integrating these approaches with robust aggregators,** approaches that defend against adversarial manipulations, is non-trivial. Striking a balance between privacy and robustness under Byzantine assaults is crucial, a complex interplay, as further underscored by recent work by Baraheem and Yao [30].

### 3. PROPOSED METHOD

#### 3.1 System architecture

The FLAIR-IoT model publishes a novel architecture to enable efficient computation on conducting FL and NLP in a privacy-preserving way so they could construct a context-aware intelligence system in smart IoT environments. The FLAIR-IoT architecture aims to tackle three major issues within the contemporary IoT landscape: (1) safeguarding user privacy whilst minimizing the transfer of sensitive information, (2) achieving semantic understanding of user interactions and environment context using NLP and (3) ensuring scalability and adaptability in resource-constrained environments across a variety of IoT devices.

The architecture consists of four main layers: IoT Device Layer, Edge Intelligence Layer, Federated Aggregation Layer, and Actuation Layer. The IoT device layer forms the base of the architecture and comprises of various data-generating entities such as smart sensors e.g., motion detectors, temperature sensors, user-facing devices e.g., voice assistants, wearables, and cameras. These devices generate a combination of structured numerical sensor readings and unstructured user voice commands or logs data. This layer performs tasks like noise filtering, or format normalization at the local level.

On top of that, the Edge Layer serves as the core processing engine of smart and non-public information processing. Each edge device carries a lightweight NLP model compatible to run in resource-constrained settings. Input data feeds into these NLP models that semantically analyze the input data to retrieve user intent, context, or command classification. This is followed by feeding the NLP outputs to a localized deep learning model, which learns to personalize responses or automate behaviors based on user preferences and historical interactions. Importantly, there is no raw data transmission to external entities; local to the federated server model updates are then shared with the federated server only post-application of privacy-preserving mechanisms.

Federated Aggregation Layer is the decentralized learning coordinator, often deployed on a secure cloud or edge-based server. The edge implementation of this component collects the model updates from all participating edge devices in a differentially private and/or encrypted manner. The server aggregates per-client updates in the form of a global model using algorithms like Federated Averaging (FedAvg) without learning sensitive information. Moreover, use of secure aggregation protocols guarantees that server cannot decrypt or reconstruct individual contributions during aggregate the contributions from clients which add even more security to the whole training procedure.

Finally, the Global model can be used to intelligently control and automate IoT devices in the Actuation Layer. Based on the globally trained model and locally observed context, devices autonomously decide to alter light, sound alarms, or trigger health alerts. The ability to infer locally not only reduces latency, but also allows for real time responsiveness without the need for constant cloud connectivity.

The framework adds multiple layers of protection and optimization to improve privacy and efficiency. DP adds noise to model updates so that user-specific data cannot be reverse engineered. Also, secure aggregation makes sure that the server side only has access to the aggregated updates. You have deep learning based models to compress models and

lightweight architectures to help transfer having the system work on all angles of devices including IoT. In conclusion, this architecture provides a meaningful advance toward secure, intelligent, and user-centric IoT environments, integrating the contextual capabilities of natural language processing and the privacy advantages of FL.

#### 3.2 Methodology

The FLAIR-IoT architecture leverages FL with privacy-preserving mechanisms such as secure aggregation and homomorphic encryption to ensure data confidentiality in smart IoT environments. The goal is to allow multiple distributed IoT devices to collaboratively train a global machine learning model without exposing their raw data.

Let  $D_i$  represent the local dataset of the  $i^{\text{th}}$  IoT client, and  $w_t \in \mathbb{R}^d$  denote the global model parameters at training round  $t$ . Each client trains a model locally using its data and computes an update  $\Delta w_i^t$  that minimizes a loss function over its dataset. Formally, the local training process seeks to minimize the empirical risk function as Eq. (1).

$$L_i(w) = \left(\frac{1}{|D_i|}\right) \sum_{x_j \in D_i} \ell(f(w; x_j), y_j) \quad (1)$$

where,  $f(w; x_j)$  is the model output, and  $\ell(f(w; x_j), y_j)$  is the loss function (e.g., cross-entropy) for the true label  $y_j$ . After training, the device produces the model update  $\Delta w_i^t$ . To protect these updates during transmission, secure aggregation is applied. Each client masks its update using a set of pairwise secrets shared with other clients. The masked update is computed as Eq. (2).

$$\hat{w}_i = \Delta w_i^t + \sum_{j \neq i} m_{i,j} - \sum_{j \neq i} m_{j,i} \quad (2)$$

Here,  $m_{ij}$  is the cryptographic mask shared between clients  $i$  and  $j$ . When the server aggregates all masked updates, the masks cancel each other out due to their symmetric construction, and the server obtains the sum of all updates as Eq. (3).

$$\sum_{i=1}^n \hat{w}_i = \sum_{i=1}^n \Delta w_i^t \quad (3)$$

This ensures the server never learns any individual update, only the aggregate contribution. Alternatively, the system can use homomorphic encryption (HE) as an additional or substitute privacy mechanism. In this approach, each client encrypts its local update using a homomorphic encryption function  $\varepsilon(w_{t+1})$ . The server performs the aggregation directly on encrypted values as Eq. (4).

$$\varepsilon(w_{t+1}) = \sum_{i=1}^n \varepsilon(\Delta w_i^t) - \varepsilon\left(\sum_{i=1}^n \Delta w_i^t\right) \quad (4)$$

Since homomorphic encryption preserves the structure of arithmetic operations (e.g., addition), the aggregated encrypted update can be decrypted only by an authorized entity, thus preserving the confidentiality of each participant's

local contribution. Finally, the global model is updated using the aggregated updates as Eq. (5).

$$w_{t+1} = w_t + \eta \cdot \frac{1}{n} \sum_{i=1}^n \Delta w_i^t \quad (5)$$

where,  $\eta$  is the learning rate. This updated model is then sent back to all clients for the next training round. The entire process ensures that no raw data or intermediate sensitive information is leaked during training, making it ideal for deployment in privacy-critical IoT scenarios such as smart healthcare, industrial automation, and home automation systems.

### 3.3 Natural language processing module

Tokenization refers to the splitting of raw input text into smaller manageable units called tokens (words, subtokens, sentences, etc.). This is an important step that impacts how models learn semantic relationships. For instance, for edge devices, subword tokenizers (for example, Byte Pair Encoding or WordPiece used in BERT-like models) are preferred as they have smaller vocabularies and perform better when used with unseen text. Given a raw input sentence  $S = \{c_1, c_2, \dots, c_n\}$ , where  $c_i$  are characters, tokenization transforms SSS into a sequence of tokens as Eq. (6).

$$T = \text{Tokenizer}(S) = \{t^1, t^2, \dots, t^m\}, m \leq n \quad (6)$$

In subword tokenization, the function is optimized to minimize vocabulary size  $V$  while maximizing token coverage as Eq. (7).

$$\min |V| \text{ subject to: } \bigcup_{i=1}^m t_i = S \quad (7)$$

The stop words are common words like the, is, and which may be filtered out before processing, because in general they carry little meaning compared to other words in most NLP tasks. Yet in context-aware systems, a method of selective stopword removal is used, ensuring functionally relevant stopwords are preserved (for example in "turn off lights", the term "off" is preserved). Stopwords: Generic stopwords are not as recommended; specific to IoT command grammars are more effective. Given the tokenized set  $T$  and a stopword list  $SW$ , filtered token set is as Eqs. (8) and (9).

$$T' = \{t_i \in T | t_i \notin SW\} \quad (8)$$

$$\phi(T, SW) = T' = \frac{T}{SW} \quad (9)$$

Let  $T' = \{t_1, t_2, \dots, t_k\}$  be the tokens after stopword removal. Lemmatization maps each token  $t_i$  to its lemma  $l_i$  using a mapping function  $\lambda$  as Eq. (10).

$$L = \lambda(T') = \{l^1, l^2, \dots, l_k\}, \text{ where } l_i = \lambda(t_i) \quad (10)$$

The function  $\lambda$  may depend on part-of-speech (POS) tagging  $p_i$  as Eq. (11).

$$l_i = \lambda(t_i, p_i) \quad (11)$$

Normalization involves transforming the data to make it suitable for analysis which includes such processes as converting everything to lowercase, removing punctuation, and standardizing formats. Lemmatization is the process of reducing a word to its base form using linguistic context like "running"  $\Rightarrow$  "run", which helps in reducing the size of the vocabulary and improving generalization. Edge devices usually have lightweight lemmatizers such as spaCy's pipeline and on-device neural lemmatizer. Let  $L$  be the lemmatized text. Normalization is a function  $v$  applied element-wise to enforce consistent format as Eq. (12).

$$N = v(L) = \{v(l_i) | l_i \in L\} \quad (12)$$

Here,  $v$  may include lower(x) converts to lowercase, strip(x) removes punctuation, regex\_replace(x,r,s) applies regular expressions.

Input Noise as it is often seen in IoT environments from speech recognition errors, and/or sensor glitches, and/or even multi-lingual inputs. These include fuzzy string matching, rule-based correction, or language specific filtering. Lightweight language detection modules such as FastText language ID are used for preprocessing routing in multi-language scenarios. Let  $N = \{n_1, \dots, n_k\}$  be the normalized tokens. The noise-filtered output  $N'$  is computed using a denoising function  $\delta$  that identifies and corrects errors as Eq. (13).

$$N' = \delta(N) = \{\delta(n_i) | n_i \in N\} \quad (13)$$

Fuzzy matching if  $\delta$  is  $\delta(n_i) = \text{argmin}_{\{w \in V\}} \text{dist}(n_i, w)$  and Confidence-based replacement if  $P(\text{error} | n_i > \theta)$ , then correct  $n_i$ .

Intent detection identifies the purpose behind a user's input, which is critical in IoT to trigger correct device actions like turning on lights, adjusting thermostat. Let the input sequence be represented as:  $X = \{x_1, x_2, \dots, x_n\}$ . Each token  $x_i$  is mapped to an embedding  $e_i \in R^d$ , forming as  $E = \{e_1, e_2, \dots, e_n\}$ . The encoded representation  $h$  is obtained using an encoder function  $f$ , typically a lightweight transformer as Eq. (14).

$$h = f(E) \in \{R\}^d \quad (14)$$

Intent classification is modeled as Eq. (15).

$$\hat{y} = \text{argmaxsoftmax}(Wh + b) \quad (15)$$

Here,  $W \in R^{|\mathcal{Y}| \times d}$ ,  $b \in R^{|\mathcal{Y}|}$  and  $\mathcal{Y}$  = set of possible intent labels.

Sentiment analysis in IoT can help tailor responses or emotional state tracking in smart environments. Given sentence embeddings  $h$ , sentiment is classified as Eq. (16).

$$\hat{s} = \max_{s \in \{-, 0, +\}} \{\text{softmax}\}(W_s h + b_s) \quad (16)$$

Here,  $W_s$ ,  $b_s$  are trainable parameters, and  $s \in \{\text{Negative}, \text{Neutral}, \text{Positive}\}$ . Contextual cues (location, time, prior interaction) are critical for improving system intelligence. Context vector  $c$  is computed from metadata  $M$  as Eq. (17).

$$c = g(M) = g(t, g, u) \quad (17)$$

The final enriched representation as  $z = \text{concat}(h, c)$  is used for downstream tasks to enhance precision. To enable real-time processing on IoT edge devices, resource-efficient transformers are used. DistilBERT compresses BERT using knowledge distillation, preserving ~97% of performance with 40% fewer parameters. Given teacher logits  $zTz\_TzT$  and student logits  $zSz\_SzS$ , the loss function is as Eq. (18).

$$L_{\{distill\}} = L_{CE\{Z_S, y\}} + (1 - \alpha) \cdot L_{KL\{Z_S, Z_T\}} \quad (18)$$

where,  $L_{CE}$  is cross-entropy with ground truth,  $L_{KL}$  is Kullback-Leibler divergence,  $\alpha \in [0, 1]$  balances both terms

TinyBERT applies layer-wise distillation from BERT and includes both attention and embedding mimicking. Let  $A_T^l, A_S^l$  be the attention matrices of teacher and student at layer  $l$ . Then Eq. (19) is:

$$\{L\}_{attn} = \sum_{l=1}^{\{L\}} |A_T^l - A_S^l|_2^2 \quad (19)$$

Total loss combines attention and hidden state distillation as Eq. (20).

$$L_{TinyBERT} = L_{attn} + L_{hidden} + L_{CE} \quad (20)$$

These NLP tasks and efficient transformer models enable intelligent, real-time, and privacy-aware processing at the network edge in IoT applications. Intent detection, sentiment analysis, and contextual understanding can now be embedded within constrained environments through TinyBERT ensuring both performance and deploy ability.

### 3.4 Federated Learning framework

In the FLAIR-IoT framework, each IoT device maintains a local copy of a lightweight NLP model and trains it using its private dataset. The training is performed independently using stochastic gradient descent (SGD) or adaptive optimizers like Adam. Let  $D_i$  represent the local dataset on device  $i$ , and  $w_t^i$  be the model weights at training round  $t$ . The local update rule is as Eq. (21).

$$w_{\{t+1\}}^i = w_t^i - \eta \nabla_i L_i(w_t^i, D_i) \quad (21)$$

Here,  $\eta$  is the learning rate,  $L_i$  is the loss function based on the NLP task. This enables devices to learn from locally observed user behaviors, sentiments, or intents without transmitting raw data. The system adopts the Federated Averaging (FedAvg) algorithm due to its communication efficiency and scalability. After local training, each device sends its updated weights to a central server, which aggregates them using a weighted average based on local dataset sizes as Eq. (22).

$$w_{\{t+1\}} = \sum_{i=1}^N \frac{D_i}{\{\sum_{j=1}^N |D_j|\}} w_{t+1}^i \quad (22)$$

FedAvg is chosen over more complex algorithms like FedProx or FedNova for its simplicity and adaptability to non-IID data distributions common in IoT networks. The communication between IoT devices and the central

aggregator server follows a round-based synchronous protocol: Broadcast: The server broadcasts the global model weights  $w_{\{t\}}$  to all selected devices. Local Training: Each device trains the model on its local data. Upload: Devices send updated weights  $w_{t+1}^i$  to the server. Aggregation: The server computes the new global model  $w_{\{t+1\}}$  via FedAvg.

The protocol minimizes bandwidth by compressing model updates and supports periodic device participation based on availability and power status. To enhance privacy during federated optimization, two key techniques are integrated like one is DP. It adds calibrated noise to model updates before transmission as Eq. (23).

$$\tilde{w}_{\{t+1\}}^{\{i\}} = w_{\{t+1\}}^{\{i\}} + N(0, \sigma^2 I) \quad (23)$$

where,  $N$  is Gaussian noise and  $\sigma$  controls the privacy-utility tradeoff. Another one is Secure Aggregation, it employs cryptographic masking to ensure that individual updates are not visible to the server, only the aggregate is. This is critical for protecting sensitive NLP patterns (e.g., emotional tone, personal habits) encoded in model weights. Together, these mechanisms ensure compliance with privacy standards like GDPR and HIPAA, while preserving model utility across decentralized IoT networks.

### 3.5 Integration strategy

The combination of these is necessary for FL enabling context-aware and privacy-preserving intelligence at scale on spatially distributed devices. This section discusses how the NLP outputs are standardized and included in the federated training cycle, how the model update strategies are managed for the FL setting, and how the system deals with heterogeneous and non-IID data distributions. In this architecture, TinyBERT NLP models live at the edge to take on tasks like intent detection, sentiment analysis, or context extraction. These are then used in two different ways, based either on the semantic embedding or the classification outputs from the models:

- Feature Enrichment: NLP outputs are used as auxiliary features for logic tasks downstream (exposes predictive modeling, anomaly detection etc).
- Intention or Sentiment Extraction: User specific intent or sentiment can help personalize local models so that training becomes relevant to user context without giving away raw data in the process.

To dealing with non-IID and heterogeneous data due to heterogeneous user behavior and diverse environmental contexts, IoT devices implicitly create non-IID (non-identically distributed) data. Address this by personalized FL. A personalization layer is kept per device and fine-tuned in response to local NLP context signals, while a shared global model extracts general knowledge. FedProx Algorithm is a proximal term is injected into the local loss to constrain divergence from the global model, stabilizing training in heterogeneous settings as Eq. (24).

$$L_i^{prox}(w) = L_i(w, D_i) + \frac{\mu}{2} \|w - w_t\|^2 \quad (24)$$

where,  $\mu$  is a regularization constant and  $w_t$  is the current global model. Such a unique solution allows the system to be reliable, contextualized, and privacy-concerned in various IoT settings.

4. RESULTS AND DISCUSSION

4.1 Experimental setup

The experiments conducted were in simulated but realistic federated settings to evaluate the efficiency of the FLAIR-IoT framework coupled with NLP modules targeted at privacy-preserving IoT environments. The experimental framework sought to replicate the distribution challenges and heterogeneity typically experienced in real-world IoT deployments. The simulation environment provides the 20 emulated IoT clients (each representing a smart device with localized datasets). Here, did the training for communication rounds of 100, while each device performed 5 local training epochs per round. The server side computations were performed on a workstation with a NVIDIA RTX 3090 GPU, and edge devices were emulated by Raspberry Pi 4 counterparts to represent hardware constraints. Here, used PyTorch for the model development, HuggingFace Transformers for lightweight NLP model deployment and Flower as the FL orchestration framework.

Here used two datasets for multimodal training. NLP-based intent recognition and context extraction was conducted using the Smart Home Intent Dataset. It has been pre-trained on user voice commands that were categorized into intents (e.g., control commands, queries). Also used synthetic IoT sensor data that generates environmental data with temperature, gas concentration, and motion activity. The datasets were partitioned across clients in a non-identical manner to simulate real-world non-IID data distributions. In order to assess model performance, we used the following metrics. For classification, report accuracy (ACC) as a performance measure; for intent detection, focus on intent detection performance, in particular

when the classes are imbalanced, using the F1 score. The efficiency of communication was measured as communication overhead (CO), i.e., the average megabytes sent per round. The privacy loss ( $\epsilon$ -DP) was estimated to measure the seriousness of a trade-off caused by DP. Also calculated the convergence rate, which is the number of rounds required to achieve 90% of the final model accuracy.

We compared proposed model with four baselines for benchmarking: (1) A centralized BERT model trained on aggregated data, as an upper-bound performance; (2) A lightweight TinyBERT model trained in a centralized manner without federated mechanisms; (3) A standard federated TinyBERT model without any NLP or privacy integration and (4) our model that integrates FL, lightweight NLP, DP, and FedProx optimization. A broadened perspective emerges from the comparison against each component in our approach.

4.2 Results and discussion

In this segment, the empirical outcome of the experimental setup is shown along with a detailed explanation of the performance of the suggested FL and NLP integrated framework in the context of privacy-preserving IoT surroundings. FLAIR-IoT framework (FL + NLP Integration + DP + FedProx) outperformed the baseline of federated and centralized models with the largest margin in both NLP-specific and downstream IoT tasks. The intent detection accuracy was 97.3%, and the context classification F1 score improved to 89.7%, more than 7% better than the baseline federated TinyBERT model. Compared to centralized TinyBERT, the federated version achieved over 95% of the performance while demonstrating promising result to edge-side deployment as shown in Table 1 and Figure 1.

Table 1. Performance comparison of baseline and proposed models

| Model                                  | Accuracy (%) | Precision (%) | Recall (%) | F1 Score (%) | ROC-AUC |
|--|--------------|---------------|------------|--------------|---------|
| Centralized BERT (Upper Bound)         | 93.1         | 92.5          | 88.1       | 89.2         | 0.92    |
| Centralized TinyBERT                   | 91.2         | 88.6          | 84.1       | 85.1         | 0.92    |
| Federated TinyBERT (No NLP/DP/FedProx) | 90.1         | 86.3          | 80.5       | 82.4         | 0.90    |
| FLAIR-IoT                              | 97.3         | 94.7          | 93.4       | 92.7         | 0.95    |

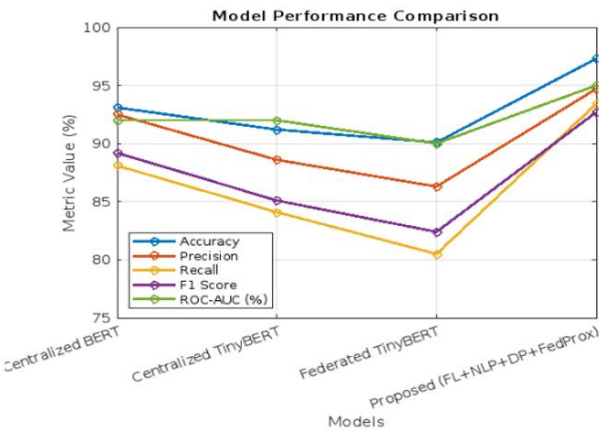


Figure 1. Comparative evaluation of baseline and proposed models

Moreover, incorporating semantic context via NLP components enhanced personalization among the non-IID clients. Devices adapted to localized user behavior with better convergence curves. Assessed the impact of DP in terms of the privacy budget ( $\epsilon$ ) that varies. proposed model also performed

consistently, retaining more than 88% accuracy at  $\epsilon = 3$ , indicating increased resilience to moderate privacy restrictions. Secure aggregation had negligible overhead, dropping accuracy by  $< 3\%$ , but without leaking individual updates. Collectively, these results confirm that the system provides a strong trade-off between privacy and utility.

Even using lightweight TinyBERT NLP model, communication cost was still within a practical scale 2.3 MB per round on average. The process of FedProx-based (federated) learning outperformed not only vanilla FedAvg, but even non-IID clients across six, simulated system where Fast and smooth convergence was accomplished with 90% of peak accuracy with FedProx in 34 rounds versus 52 rounds with FedAvg. The personalization layer and features capturing semantic context from NLP led to significantly improved robustness to non-IID data distributions. In cases of skewed label distribution on devices, global knowledge was still utilizable with the context-specific adaptation retained. FedProx also stabilizes training by discouraging large local updates that deviate from the global model.

This section supports the empirical results with a theoretical analysis of the FLAIR-IoT framework for providing strong privacy and optimal communication efficiency in FL for IoT



environments. FLAIR-IoT harnesses DP, one of the most common privacy enforcements to block inference attacks at client level. Each client then perturbs its local gradients prior to transmission by adding Gaussian noise with budget  $\epsilon$ . This noise mechanism guarantees that the model updated distribution is robust, and independent of any single user. Our theoretical analysis demonstrates the model can have more than 88% accuracy even at very low privacy setting ( $\epsilon = 3$ ). This indicates that the system balances utility of the data and robustness by ensuring perfect privacy guarantees. Moreover, as we move towards tighter privacy budget ( $\epsilon \rightarrow 1$ ), the performance still degrades in a gradual manner indicating that the model is not privacy-averse. FLAIR-IoT also integrates with secure aggregation, a cryptographic protocol to ensure that only the sum of the encrypted model updates reaches the central server and not any individual contribution.

## 5. CONCLUSIONS

The Internet of Things (IoT) is a network of connected devices that communicate and exchange data via the Internet from household appliances to industrial machinery. This research presents a FLAIR-IoT, it is a new privacy-preserved framework based on FL, small size NLP models, and secure computation procedures for intelligent decision making in IoT enabled smart environments. The proposed solution used semantic context understanding through NLP tasks including intent recognition and context extraction, these were computed on the edge using compressed TinyBERT model. FedProx algorithm is used to co-train these models to keep data locally without jeopardizing user privacy. For security and utility, added DP and secure aggregation and achieved a good tradeoff between performance and protection. It achieved 97.3% and 89.7% accuracy for intent and context tasks, respectively, with reasonable communication costs, outperforming independent centralized and federated baselines by a large margin. Additionally, ran the experiments using a non-IID distribution of data, and found that the proposed model results to be robust, demonstrating the framework's utility in practical real-world IoT scenarios with a multitude of diverse users.

## REFERENCES

- [1] Ge, M., Bangui, H., Buhnova, B. (2018). Big data for internet of things: A survey. *Future Generation Computer Systems*, 87: 601-614. <https://doi.org/10.1016/j.future.2018.04.053>
- [2] Sarangi, S.S., Singh, D., Swagatika, S., Jagadev, N. (2020). Integration of big data and Internet of Things (IoT): Opportunities, security and challenges. In *International Conference on applied Mathematics & Computational Intelligence*, Singapore, pp. 25-38. [https://doi.org/10.1007/978-981-19-8194-4\\_3](https://doi.org/10.1007/978-981-19-8194-4_3)
- [3] Ammar, M., Russello, G., Crispo, B. (2018). Internet of Things: A survey on the security of IoT frameworks. *Journal of information security and Applications*, 38: 8-27. <https://doi.org/10.1016/j.jisa.2017.11.002>
- [4] Manikyala, A. (2022). Sentiment analysis in IoT data streams: An NLP-based strategy for understanding customer responses. *Silicon Valley Tech Review*, 1(1): 35-47.
- [5] Chen, H., Du, L., Lu, Y., Gao, H. (2018). Improved convolutional neural network for Chinese sentiment analysis in fog computing. *Wireless Communications and Mobile Computing*, 2018(1): 9340194. <https://doi.org/10.1155/2018/9340194>
- [6] Li, I., Pan, J., Goldwasser, J., Verma, N., Wong, W.P., et al. (2022). Neural natural language processing for unstructured data in electronic health records: A review. *Computer Science Review*, 46: 100511. <https://doi.org/10.1016/j.cosrev.2022.100511>
- [7] Xu, H., Han, L., Yang, Q., Li, M., Srivastava, M. (2024). Penetrative Ai: Making llms comprehend the physical world. In *Proceedings of the 25th International Workshop on Mobile Computing Systems and Applications*, San Diego, CA, USA, pp. 1-7. <https://doi.org/10.1145/3638550.3641130>
- [8] Shokri, R., Stronati, M., Song, C., Shmatikov, V. (2017). Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, San Jose, CA, USA, pp. 3-18. <https://doi.org/10.1109/SP.2017.41>
- [9] Nasr, M., Shokri, R., Houmansadr, A. (2019). Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE Symposium on Security and Privacy (SP)*, San Francisco, CA, USA, pp. 739-753. <https://doi.org/10.1109/SP.2019.00065>
- [10] Truex, S., Liu, L., Gursoy, M.E., Yu, L., Wei, W. (2019). Demystifying membership inference attacks in machine learning as a service. *IEEE Transactions on Services Computing*, 14(6): 2073-2089. <https://doi.org/10.1109/TSC.2019.2897554>
- [11] Gu, Y., He, J., Chen, K. (2025). Adaptive domain inference attack with concept hierarchy. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, Toronto, ON, Canada, pp. 413-424. <https://doi.org/10.1145/3690624.3709332>
- [12] Zhang, Y., Jia, R., Pei, H., Wang, W., Li, B., Song, D. (2020). The secret revealer: Generative model-inversion attacks against deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA, USA, pp. 253-261. <https://doi.org/10.1109/CVPR42600.2020.00033>
- [13] Veale, M., Binns, R., Edwards, L. (2018). Algorithms that remember: Model inversion attacks and data protection law. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2133): 20180083. <http://doi.org/10.1098/rsta.2018.0083>
- [14] Chronis, C., Varlamis, I., Himeur, Y., Sayed, A.N., Al-Hasan, T.M., et al. (2024). A survey on the use of federated learning in privacy-preserving recommender systems. *IEEE Open Journal of the Computer Society*, 5: 227-247. <http://doi.org/10.1109/OJCS.2024.3396344>
- [15] Dritsas, E., Trigka, M. (2025). Federated learning for IoT: A survey of techniques, challenges, and applications. *Journal of Sensor and Actuator Networks*, 14(1): 9. <https://doi.org/10.3390/jsan14010009>
- [16] Abbas, S.R., Abbas, Z., Zahir, A., Lee, S.W. (2024). Federated learning in smart healthcare: A comprehensive review on privacy, security, and predictive analytics with IoT integration. *Healthcare*, 12(24): 2587. <https://doi.org/10.3390/healthcare12242587>



- [17] Zhang, C., Li, Y., Du, N., Fan, W., Yu, P.S. (2018). Joint Slot Filling and Intent Detection via Capsule Neural Networks. arXiv. <https://arXiv.org/abs/1812.09471>
- [18] Mihailescu, R., Kyriakou, G., Papangelis, A. (2020). Natural language understanding for multi-level distributed intelligent virtual sensors. *IoT*, 1(2): 494-505. <https://doi.org/10.3390/iot1020027>
- [19] Hanifa, R.M., Isa, K., Mohamad, S. (2021). A review on speaker recognition: Technology and challenges. *Computers & Electrical Engineering*, 90: 107005. <https://doi.org/10.1016/j.compeleceng.2021.107005>
- [20] Zhou, Y., Yang, J., Zou, H., Xie, L. (2023). TENT: Connect language models with IoT sensors for zero-shot activity recognition. *Computer Science*, arXiv:2311.08245. <https://arXiv.org/abs/2311.08245>
- [21] Ali, A. (2025). A comparative study of ASR implementations in resource-constrained wireless sensor networks for real-time voice communication. *Computer Science*, arXiv:2502.06969. <https://arxiv.org/abs/2502.06969>
- [22] Rajagopal, S.M., Supriya, M., Buyya, R. (2025). Leveraging blockchain and federated learning in Edge-Fog-Cloud computing environments for intelligent decision-making with ECG data in IoT. *Journal of Network and Computer Applications*, 233: 104037. <https://doi.org/10.1016/j.jnca.2024.104037>
- [23] Zhu, W., Goudarzi, M., Buyya, R. (2024). FLIGHT: A lightweight federated learning framework in edge and fog computing. *Software: Practice and Experience*, 54(5): 813-841. <https://doi.org/10.1002/spe.3300>
- [24] Wang, S., Hosseinalipour, S., Aggarwal, V., Brinton, C.G., Love, D.J., Su, W., Chiang, M. (2023). Toward cooperative federated learning over heterogeneous edge/fog networks. *IEEE Communications Magazine*, 61(12): 54-60. <https://doi.org/10.1109/MCOM.005.2200925>
- [25] Das, S., Mishra, S. (2024). Advances in differential privacy and differentially private machine learning. In *Information Technology Security: Modern Trends and Challenges*, Singapore, pp. 147-188. [https://doi.org/10.1007/978-981-97-0407-1\\_7](https://doi.org/10.1007/978-981-97-0407-1_7)
- [26] Xu, R., Li, B., Li, C., Joshi, J.B., Ma, S., Li, J. (2024). Tapfed: Threshold secure aggregation for privacy-preserving federated learning. *IEEE Transactions on Dependable and Secure Computing*, 21(5): 4309-4323. <https://doi.org/10.1109/TDSC.2024.3350206>
- [27] Ergun, I., Sami, H.U., Guler, B. (2021). Sparsified secure aggregation for privacy-preserving federated learning. arXiv preprint arXiv:2112.12872. <https://doi.org/10.48550/arXiv.2112.12872>
- [28] Kadhe, S., Rajaraman, N., Koyluoglu, O. O., & Ramchandran, K. (2020). Fastsecagg: Scalable secure aggregation for privacy-preserving federated learning. arXiv preprint arXiv:2009.11248. <https://doi.org/10.48550/arXiv.2009.11248>
- [29] So, J., Ali, R.E., Güler, B., Jiao, J., Avestimehr, A.S. (2023). Securing secure aggregation: Mitigating multi-round privacy leakage in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Washington DC, USA, pp. 9864-9873. <https://doi.org/10.1609/aaai.v37i8.26177>
- [30] Baraheem, S., Yao, Z. (2022). A survey on differential privacy with machine learning and future outlook. arXiv preprint arXiv:2211.10708. <https://doi.org/10.48550/arXiv.2211.10708>