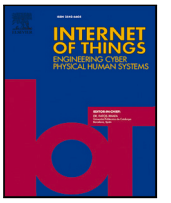
Internet of Things 25 (2024) 101038

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot



A hybrid model for energy-efficient Green Internet of Things enabled intelligent transportation systems using federated learning

Sarah Kaleem [a,*], Adnan Sohail [a], Muhammad Babar [b], Awais Ahmad [c], Muhammad Usman Tariq [d]

[a] Iqra University Karachi, Islamabad Campus, Pakistan
[b] Robotics and Internet of Things Laboratory, Prince Sultan University, Saudi Arabia
[c] College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia
[d] Department of Marketing, Operations, and Information Systems, Abu Dhabi University, Abu Dhabi, United Arab Emirates

ARTICLE INFO

Keywords:
Federated learning
Internet of Things (IoT)
Green IoT
Smart transportation
Energy-efficiency
TensorFlow federated

ABSTRACT

In the rapidly evolving Internet of Things domain, managing voluminous data streams while ensuring energy efficiency remains a cardinal challenge. Our research introduces a hybrid model designed specifically for an IoT system. This paper sheds light on our unique framework that emphasizes efficient communication and prioritizes energy conservation. The innovative model architecture is lightweight, fostering compatibility with resource-constrained devices and paving the way for personalized federated averaging during training. A crucial highlight of our methodology includes local training using lightweight optimization instead of traditional data transmission, reducing energy overheads considerably. Furthermore, we propose a novel approach for model aggregation using personalized energy-aware averaging. This process iteratively refines a global model by aggregating received updates, drastically reducing the data transfer compared to conventional methods. Lastly, we integrate an energy-aware updates management system, continually monitoring device energy metrics and making adaptive data transmission and model update decisions. The model ensures optimal participation in global updates by consistently monitoring devices' energy metrics and setting adaptive thresholds. This study leverages the richly annotated Udacity Self-Driving Car Dataset provided by Roboflow to evaluate a novel federated learning model to optimize energy consumption for IoT systems. Our experiments simulate real-world collaborative learning scenarios using the TensorFlow Federated (TFF). Our focus on energy consumption evaluation revealed that our proposed model offers significant energy savings when analyzed based on data communication round time and individual node training duration. A comparative analysis between the proposed and traditional models uncovers substantial improvements in energy efficiency. In our experiments, the proposed approach demonstrated a commendable accuracy of 93.27%. Notably, the local communication time was streamlined to 1.21 s, while the global communication was efficiently clocked at 4.76 s. When compared to traditional methods, these results not only underscore the effectiveness and efficiency of our methodology in the context of IoT systems but also highlight its superior performance.

* Corresponding author.
E-mail addresses: sarahkaleem33887@iqraisb.edu.pk (S. Kaleem), adnan.sohail@iqraisb.edu.pk (A. Sohail), mbabar@psu.edu.sa (M. Babar), abmohmmed@imamu.edu.sa (A. Ahmad), muhammad.kazi@adu.ac.ae (M.U. Tariq).

https://doi.org/10.1016/j.iot.2023.101038
Received 28 September 2023; Received in revised form 3 December 2023; Accepted 19 December 2023
Available online 21 December 2023

1. Introduction

The Internet of Things (IoT) has rapidly transformed our world by connecting billions of devices, forming an intricate web of data exchange and processing [1,2]. This exponential growth in connected devices generates immense volumes of data, which, while providing a wealth of opportunities, poses significant challenges in energy consumption and environmental sustainability [3]. While the IoT paradigm can potentially revolutionize transportation sectors, its associated energy consumption poses significant challenges. As IoT ecosystems continue to burgeon, ensuring their sustainability becomes paramount. This brings us to the intersection of Green IoT [4]. "Green IoT" was coined to encapsulate a holistic approach to environmental sustainability in the IoT domain. A foundational exploration into Green IoT, detailing the principles and paradigms that underpin this progressive field [5]. It is expanded upon the evolution from a mere energy-centric focus to a broader environmental sustainability perspective within the IoT realm, emphasizing the need to transition from energy-efficient operations to environmentally sustainable ones. Green IoT integrates principles of environmental sustainability throughout the lifecycle of IoT devices, encompassing their design, manufacturing, operation, and eventual disposal [6]. Green IoT envisions a holistic approach to sustainability, aiming for minimal carbon footprints, reduced e-waste, and eco-friendly materials. Energy optimization in IoT has been a focal area of research in recent years, primarily aimed at maximizing the efficiency of device operations and reducing overall energy consumption [7]. The overarching goal is to ensure prolonged device lifetimes, decrease operational costs, and minimize environmental impacts, thereby aligning with the broader objectives of green computing [8].

One of the most promising tools to achieve these objectives is Federated Learning (FL). FL is emerging as a game-changer in decentralized machine learning, especially for IoT devices. Traditional machine learning models often require data to be centralized in one location for processing. This can be energy-intensive and raise privacy concerns. FL flips this model by decentralizing the learning process. Instead of sending all data to a central server, devices can learn and update their models locally [9]. With FL, devices collaborate. They learn from their local data and then share their model updates (not the raw data) with other devices. This collaborative approach allows for a more comprehensive and robust model without compromising individual data privacy. One of the significant challenges with IoT devices is the energy consumption associated with transmitting large volumes of data. Since FL allows for learning at the device level, there is a drastic reduction in the amount of data that needs to be sent back and forth. This decentralized approach optimizes energy use and addresses one of the primary energy-draining activities in IoT systems [10]. With increasing concerns about data privacy, FL offers a solution. Since raw data remains on the device and only model updates are shared, there is a reduced risk of data breaches or misuse. This decentralized data storage approach ensures that individual data remains confidential [11]. IoT devices generate a vast amount of data. Processing this data centrally can be overwhelming and inefficient. FL, however, optimizes this by allowing individual devices to process their data. These devices can then derive insights beneficial for energy conservation and overall system efficiency [12]. In essence, FL is revolutionizing how IoT devices learn and process data, making the entire system more energy-efficient, private, and robust.

Furthermore, FL augments the capability to handle the vast data streams generated by IoT devices, refining the learning processes and contributing to energy savings [13]. There is a promising avenue to achieve energy-optimized IoT systems [14,15]. In the broader context of Green IoT, the FL has the potential to drive energy efficiency and promote the design of eco-friendly applications [16]. This holistic approach, combining energy optimization with a broader environmental sustainability vision, is the future trajectory of IoT deployments. This synergy of FL stands to revolutionize how data streams in IoT are managed. However, despite the rich tapestry of literature on individual domains of Energy-Optimized IoT, Green IoT, and Federated Learning, there needs to be more comprehensive studies that weave all these threads into a singular narrative. While Fernando and Kumar have touched upon the applications of Green IoT, their study needs to integrate the role of FL.

While there is significant literature on these facets independently, there must be a clear research gap in synthesizing them cohesively. Specifically, when framed within the larger narrative of Green IoT, comprehensive investigations into the collective implications of FL are scant. This gap calls for a deeper exploration into the harmonious interplay of these domains, thereby guiding the development of integrated solutions for a truly sustainable IoT ecosystem. This paper explores the harmonization of Energy Optimization and green IoT using Federated Learning, shedding light on emerging techniques, benefits, and the prospective future of this multifaceted field. This paper delves into the synergies between FL for energy optimization, aiming to provide insights into this interdisciplinary field's techniques, benefits, and future trajectories. The key contributions of our research are:

- Lightweight Model Design for IoT Nodes: Considering the inherent limitations of IoT devices, our model's architecture is intentionally crafted to be lightweight. This strategic design not only ensures that resource-constrained devices can execute advanced operations without performance degradation but also facilitates quicker training times. Moreover, its lightweight nature enhances communication efficiency and reduces energy consumption. A distinctive feature of this lightweight approach is our utilization of a lightweight optimizer.
- Dynamic Personalized Model Aggregation Technique: A unique approach to model aggregation that employs personalized energy context-aware averaging. This method ensures that the aggregation of model updates is inherently sensitive to the individual energy context of each participating device.
- Energy-Aware Updates Management: Our research integrates an advanced energy-aware updates management mechanism, which monitors real-time energy metrics of devices. It enables clients to adjust their actions based on current energy reserves. This helps the device last longer and work betters.
- Realistic Simulation of Collaborative Learning Environments: Harnessing the potential of the TensorFlow Federated (TFF) platform, our experiments adeptly emulate authentic collaborative learning scenarios. This emulation enhances our findings' credibility and solidifies our research's relevance.

- Rigorous Model Evaluation with a Comprehensive Dataset: Using the Udacity Self-Driving Car Dataset from Roboflow, our research is anchored on a solid empirical foundation. Such a thorough evaluation certifies the model's effectiveness and accentuates its potential in real-world IoT applications.

The structure of this paper unfolds as such: Section 2 provides an extensive review of the literature, underscoring pivotal research and key notions that shape our inquiry. Section 3 outlines our suggested framework, and it will help to understand the associated methods and foundational tenets. Section 4 unveils the outcomes of our research, coupled with a thorough discourse that situates our discoveries in the broader academic context. Conclusively, Section 5 draws together the gleaned insights and contemplates the ramifications of our findings.

2. Literature review

The ever-evolving landscape of FL has been rapidly integrating into various domains, primarily on the IoT and mobile edge networks. Several proposals have recently delved into enhancing FL's performance, ensuring energy efficiency, and overcoming intrinsic challenges. An optimized hierarchical FL framework tailored to IoT heterogeneous systems is proposed [17]. This approach, mainly designed for gradient-descent-based machine learning models, addresses the challenge of non-uniformly distributed data. By testing this framework on real-world datasets, significant improvements were observed, including a 4%–6% boost in classification accuracy and an impressive 75%–85% reduction in communication rounds. In the wake of increasing energy consumption concerns, an energy-efficient and Time-sensitive Task Assignment (ETTA) mechanism is presented to ensure energy efficiency [18]. ETTA balances energy consumption with training time while promoting client honesty and ensuring system stability specially designed for cross-silo FL. The effectiveness of ETTA in dealing with dishonest client behaviors is noteworthy. The impending shift towards sixth-generation (6G) networks was highlighted by emphasizing green and energy-efficient intelligent connectivity [19]. This paper proposed a dynamic incentive and reputation mechanism, utilizing the Stackelberg game and cooperative game theories, to optimize energy consumption and training performance in FL. Simulation results provided by the authors underscored the mechanism's potential to boost client participation and enhance overall system efficiency.

Given FL's high communication costs, especially in environments with non-identically distributed (non-IID) data, employing analog aggregation to trim communication overheads is proposed [20]. Introducing data redundancy in their approach effectively tackled non-IID data challenges. An exciting component of their research was formulating an energy-aware dynamic worker scheduling policy, which, when tested on the MNIST dataset, showcased an accuracy improvement of 9.8% and optimal performance within energy constraints. A joint approach combining wireless transmission and weight quantization looking into integrating FL into future wireless network designs [21]. By formulating an optimization problem, the authors aimed to strike a balance between computation and transmission costs. When validated through simulations, this paper's iterative algorithm for optimal bandwidth and quantization allocation further reinforced the approach's viability. A novel adaptive FL algorithm is proposed for recognizing the energy challenges faced by battery-limited mobile edge devices [22]. The algorithm ensured convergence and allowed adjustable communication compression, catering to individual device environments. The method's simulations firmly established its prowess in achieving energy savings without compromising learning performance. Energy-efficient FL in IoT networks powered by fog–cloud computing is explored [23]. Interestingly, the paper's empirical findings suggest that for large-scale IoT setups, training directly on devices seems more energy-efficient than on fog access points.

Adding another dimension to the realm of FL, the authors in [24] introduced a distributed federated learning (DBFL) framework for 5G/6G networks. DBFL proposed enhanced connectivity solutions for distant devices while reducing energy consumption by addressing the limitations of traditional IoT device-to-base station communication. A green-quantized federated learning (FL) framework that uses quantized neural networks (QNNs) is tailored for energy-constrained wireless devices. By applying quantization in training and transmission, the approach minimizes energy consumption. An optimization problem balances energy use with communication efficiency, and the study analytically deduces the system's convergence rate. The framework employs Nash bargaining to negotiate energy, accuracy, and convergence trade-offs. Remarkably, simulations reveal up to a 70% energy savings compared to traditional full-precision FL without compromising convergence [25]. The increasing attention towards environmental sustainability in AI technologies focuses on optimizing FL for energy efficiency [26]. By integrating a penalizing function with Deep Reinforcement Learning, the method proposed herein showcased a remarkable energy saving of up to 94% over benchmark methods. Yet another notable contribution introduced "AutoFL" which employed reinforcement learning to optimize model convergence time and energy efficiency in edge deployments. Results indicated AutoFL's capabilities in achieving faster model convergence and enhanced energy efficiency [27].

In [28], researchers focus on energy-efficient resource allocation in Federated Learning (FL) for wireless networks, presenting an iterative algorithm to optimize resources, resulting in up to 59.5% energy savings over traditional FL methods. Concurrently, [29] addresses communication delays in FL, especially prevalent in edge devices, by unveiling a communication-centric framework. This new approach employs probabilistic device selection and model parameter quantization, showcasing a 3.6% improvement in accuracy and a notable 87% reduction in convergence time. Meanwhile, [30] zeroes in on energy consumption in mobile edge computing-enabled cellular networks. Using an SVM-based federated learning technique, they report a 20.1% energy reduction compared to standard methods. As the advancements in FL continue, a myriad of solutions, like UAVs with edge computing and intelligent reflecting surface (IRS) technologies, as proposed in [31,32], further push the boundaries of what FL can achieve. These innovative approaches solve intrinsic challenges and pave the way for more sustainable and efficient learning in decentralized systems. The field of Federated Learning is witnessing transformative advancements. As researchers grapple with challenges of

energy efficiency, communication bottlenecks, and system stability, innovative solutions are emerging, harnessing the potential of FL to its fullest.

The extensive research in the field of FL for IoT-enabled ITS and mobile edge networks shows a lot of progress, but some critical areas still need to be fully explored. Most current innovations in FL focus on improving performance, energy efficiency, or dealing with unevenly distributed data. While these developments have expanded the scope of FL, there needs to be a comprehensive model that combines all these elements. Moreover, many studies need to combine energy efficiency with eco-friendly approaches in IoT systems thoroughly. Our study aims to fill this gap. We are introducing a new hybrid model, which focuses on smoothly combining effective communication with strong energy savings. Our work is not just another contribution to the field but a unifying solution that combines various individual advancements in existing research. This approach aims to create a more sustainable, efficient, and environmentally friendly IoT system enhanced by FL. Our contribution is not merely an incremental addition. Instead, it is a comprehensive, unifying solution that combines various fragmented advancements. Our goal is to leverage the strengths of FL to foster a more connected, smart, and green future.

3. Proposed framework

This section introduces a novel hybrid model representing a pivotal advancement in the Energy-Efficient Green IoT Systems realm. Leveraging the power of federated learning, our proposed model is a cornerstone solution to address the pressing challenges of energy optimization and sustainability within the IoT landscape. By seamlessly using federated learning techniques, our model enhances the efficiency of IoT systems and sets a precedent for pioneering approaches towards a more eco-conscious and sustainable future. This section delves into the intricacies of our hybrid model, elucidating its architecture, components, and the innovative strategies employed to revolutionize IoT energy management while preserving data privacy and security. I have included an overview of the proposed framework in Fig. 1.

As depicted in Fig. 1, the proposed hybrid model unfolds within the dynamic landscape of an IoT-enabled ITS System, where many IoT devices operate to generate data. This environment is the foundation upon which our model operates, harnessing the wealth of data produced by these IoT devices. The model embarks on an Energy-Aware Data Quality Assessment journey in the initial phases, meticulously scrutinizing the data to ensure its validity and relevance. Upon ensuring data quality, a series of essential preprocessing steps are undertaken to prepare the data for federated learning. Data Partitioning comes into play, distributing the dataset among several nodes, thus creating, simulating, and mimicking the real-time federated learning scenario. This step paves the way for forming federated data, where the data becomes ready for collaborative learning. Model Architecture Selection follows, where our model design incorporates a lightweight mechanism to ensure compatibility with resource-constrained IoT devices. With the architecture in place, the model delves into Federated Training, a pivotal phase that includes Personalized Federated Averaging and real-time Monitoring of the training process. This iterative approach enables efficient learning while safeguarding data privacy and energy efficiency.

As we progress, the importance of thorough evaluation becomes increasingly critical. Both local and global assessments of our model are essential, revealing the performance of individual nodes and the overall efficacy of the model. We also assess energy usage, offering valuable insights into the model's energy impact within the IoT system. Emphasizing the need for ongoing improvement, the model embarks on continual refinement. This phase involves fine-tuning the model, adjusting data, and exploring different federated learning strategies. These iterative enhancements are vital, ensuring that the model evolves and remains responsive to shifts in data trends and optimizes energy efficiency. Our hybrid model presents a well-rounded approach, adeptly tackling the complexities of Energy-Efficient Green IoT Systems. The strategic application of federated learning significantly reduces energy consumption, substantially contributing to developing more sustainable IoT environments. Fig. 2 provides a detailed illustration of this comprehensive framework.

3.1. Local training using lightweight optimization

Devices with sensors generate tremendous data. Transferring this data to a central cloud server incurs latency and energy costs. Devices train models locally instead of transmitting vast data to central servers. This reduces the energy costs associated with data transmission, which is especially beneficial for devices in remote or bandwidth-constrained environments. Each device uses its data to adjust model parameters independently without data transmission by employing local training. This can result in significant energy savings, especially considering a network's thousands or millions of IoT devices. This includes the following procedure:

- Equip IoT devices with a basic model structure suitable for the given task.
- Use local datasets to train this model. Employ lightweight optimization algorithms such as stochastic gradient descent (SGD) for constrained devices.
- Regularly save the model parameters to avoid data loss or redundancy in computation.

Algorithm 1 performs supervised learning on a local dataset L containing n data points. Initially, the model has a predefined weight w and bias b, a set learning rate α, and a specific number of training rounds or epochs E. For each epoch, the model iterates through every data point in LL and predicts the output using the current weight and bias. Based on the difference between the prediction and the actual label, it computes the gradient of the cost function J concerning the model's parameters. The weight and

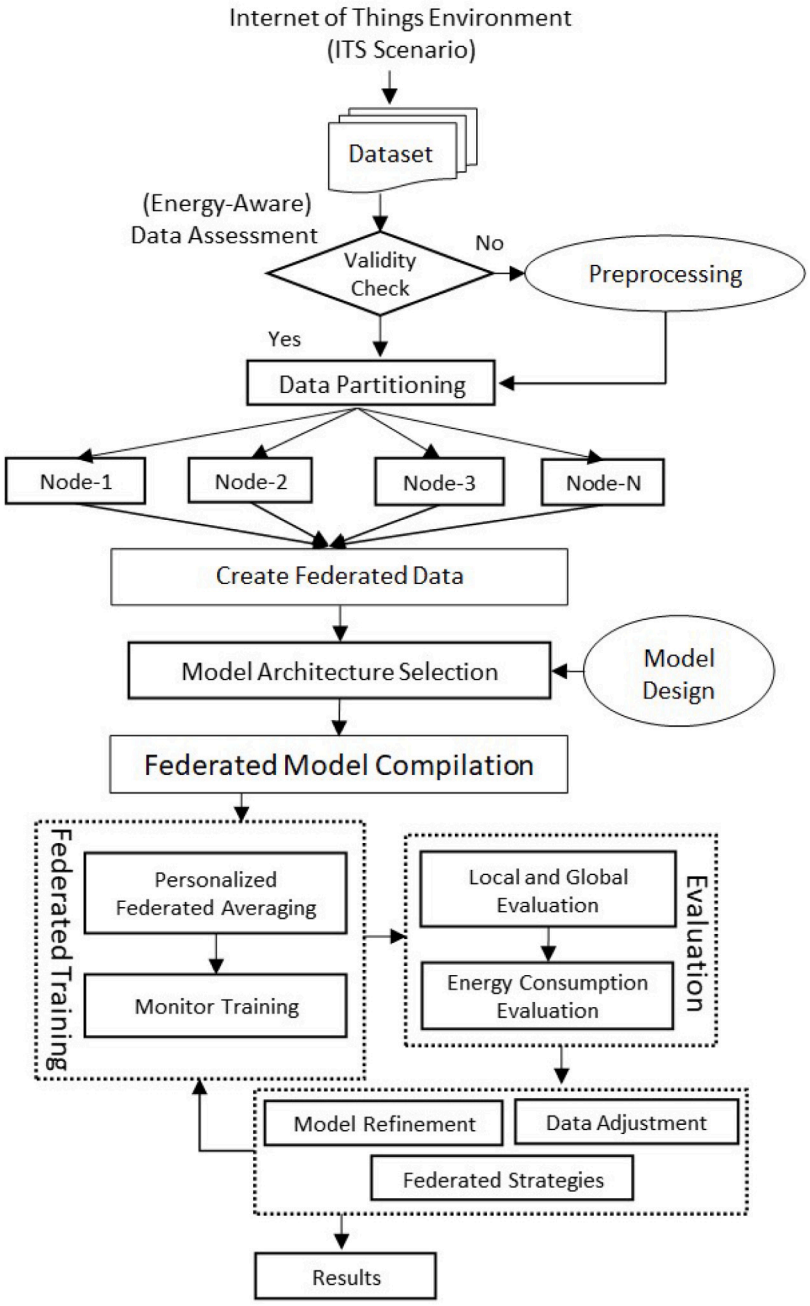


Fig. 1. Proposed framework overview.

bias are then updated using this gradient and the learning rate. This prediction, error calculation, and parameter update process is repeated for all data points across all epochs. The algorithm's core is the update rule, which adjusts the weight and bias using the cost function's gradient to minimize the prediction error. After all epochs, the final output is a trained weight and bias that ideally models the relationship in the given dataset. In addition, we utilized the optimizer. Optimizer is used to adjust the attributes of the neural network, such as weights and learning rate, to minimize the loss function. There are two major optimizers used in the proposed FL process. They are applied during the training process.

- Client Optimizer: This optimizer is used on each client during the local training phase. When individual clients train on their local data, they use this optimizer to adjust the model parameters to minimize the local loss. Common choices for client optimizers include SGD (Stochastic Gradient Descent) and its variants.

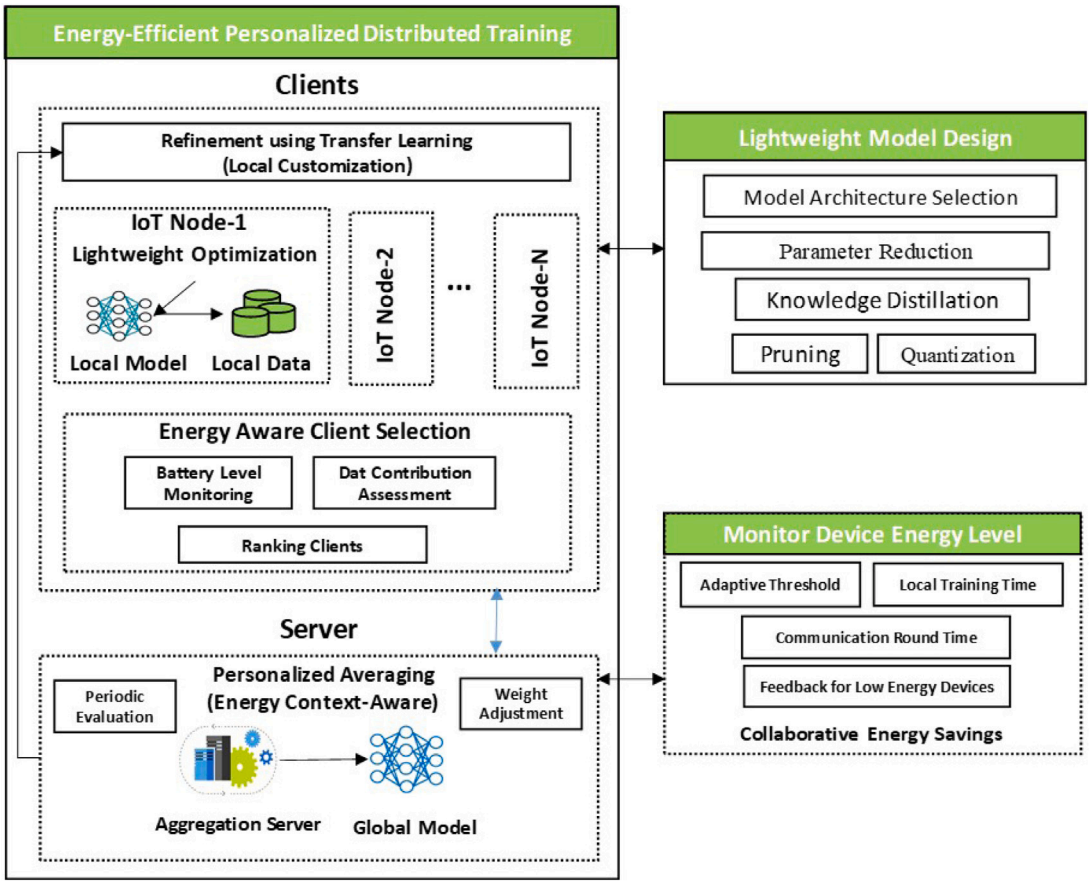


Fig. 2. Proposed framework.

Algorithm 1 Local Training using Lightweight Optimization

Require: Local dataset L with n data points (or rows) and associated labels, initial weight w, bias b, learning rate α, number of epochs E

Ensure: Trained weight w and bias b

for epoch = 1 to E **do**
  for each data point (X_i, y_i) in L **do**
    Calculate the prediction $\hat{y}_i$ using $\hat{y}_i = wX_i + b$
    Compute the gradient of the cost function $\nabla J(w, b)$
    Update weight w using $w := w - \alpha \nabla J(w, b)$
    Update bias b using $b := b - \alpha \frac{\partial J}{\partial b}$
  end for
end for
return trained weight w and bias b.

Update Rule

1. For each training sample (X_i, y_i):
2. Compute gradient: $\nabla J(w, b) = \frac{\partial J}{\partial w}$
3. Update weights: $w := w - \alpha \nabla J(w, b)$
4. Update bias: $b := b - \alpha \frac{\partial J}{\partial b}$

Where, JJ is the cost function, $\alpha\alpha$ is the learning rate, and $\nabla J \nabla J$ is the cost function gradient concerning model parameters.

- Server Optimizer: After clients send their model updates to the central server, the server aggregates them. The server optimizer then updates the global model based on this aggregated information. This optimizer ensures that the global model is refined using contributions from all participating clients.

Many factors drive the preference for lightweight optimization in training clients within an FL framework. Firstly, it significantly reduces the computational load on participating devices, which is particularly important given that these devices often have limited computational capabilities. This approach ensures energy efficiency, which is crucial for battery-powered devices, and enhances the overall scalability of the FL process. This can accommodate thousands or millions of devices with varied capacities. Lightweight optimization also contributes to faster convergence of the learning model, an essential aspect in a federated setting to minimize training time and resource consumption. This efficiency enables broader participation from a diverse range of devices, leading to more robust and generalized models due to the increased diversity in training data. Moreover, it optimizes network efficiency by reducing the size of data and model updates, thus conserving bandwidth. Such methods are adaptable to dynamic environments where client availability and network conditions are variable. Additionally, lightweight optimization can be pivotal in preserving privacy, as it can be tailored to limit the amount of data required from each client.

3.2. Model aggregation using personalized energy context-aware averaging

In our research, the concept of Energy Context-Aware Averaging is central to the framework. This method is vital after the local training phase is completed on individual clients. Unlike traditional approaches that require substantial data transmission, our method only transmits the model updates, specifically the weights and biases, to a central server. This approach leads to a significant reduction in the volume of data transmitted. Each client packages its model information into a format and sends it safely to the central server after finishing their local training (or after a specific interval). Upon receiving these parameters, the central server conducts a thorough verification to ensure their integrity and completeness. This step is critical for maintaining the accuracy and reliability of the model. To aggregate model parameters from different devices and create a globally coherent model, we utilize Algorithm 2. This algorithm efficiently combines parameters from different devices to construct a globally coherent model. Post-aggregation, these parameters are normalized to produce the global weight and bias values. These aggregated and normalized parameters are then distributed back to all participating devices.

Each device updates its local parameters accordingly and gets on another round of training upon receiving the aggregated model parameters. This time, utilizing the updated global model. This cyclical process harnesses the power of distributed learning across multiple devices, simultaneously ensuring the privacy and integrity of the data involved. One of the notable advantages of this approach is that the aggregated model benefits from the diversity of datasets contributed by various devices. This diversity often leads to enhanced performance and robustness of the model. In our IoT-enabled ITS environment, devices send their model parameters to the central server at specific intervals. This systematic approach not only streamlines the model updating and aggregation process but also contributes to the overall efficiency and effectiveness of the distributed learning system. The clients send their model parameters to a central server at specified intervals as follows:

1. Upon completing a local training round or at specified intervals, each IoT device serializes its model parameters (like weights w and biases b).
2. The serialized model parameters are transmitted to the central server using a secure and efficient communication protocol.
3. The central server acknowledges receipt to ensure the integrity and completeness of data transmission.

Algorithm 2 Energy Context-Aware Personalized Weighted Averaging

Require: Model parameters w_i, b_i from n devices, number of data samples d_i from each device.
Ensure: Aggregated model parameters $w_{\text{global}}, b_{\text{global}}$

Initialization
Set $w_{\text{sum}} = 0$ and $b_{\text{sum}} = 0$
Set $d_{\text{total}} = \sum_{i=1}^{n} d_i$
Receive Model Parameters
for each device i from 1 to n **do**
  Receive w_i and b_i from device i
  Update w_{sum} using $w_{\text{sum}} = w_{\text{sum}} + d_i \times w_i$
  Update b_{sum} using $b_{\text{sum}} = b_{\text{sum}} + d_i \times b_i$
end for
Aggregation
Compute $w_{\text{global}} = \frac{w_{\text{sum}}}{d_{\text{total}}}$
Compute $b_{\text{global}} = \frac{b_{\text{sum}}}{d_{\text{total}}}$
Dispatch Aggregated Model
for each device i from 1 to n **do**
  Send w_{global} and b_{global} to device i
end for
return $w_{\text{global}}, b_{\text{global}}$

Algorithm 2 aggregates model parameters from nn different devices to produce a globally coherent model. Initially, summed weights and biases are set to zero, and the total number of data samples across all devices is computed. For each device, the

algorithm retrieves its specific model parameters (weight wiwi and bias bibi) and scales them by the number of data samples from that device. The scaled parameters are then added to the cumulative sums. Once all device parameters are accumulated, they are normalized by the total number of data samples to produce global weight wglobal and bias bglobal. These aggregated parameters are then dispatched back to all the devices, providing each device with a consistent and collective model representation derived from the entire network. The weighted averaging is also described using Eqs. (1) and (2).

$$w_{\text{global}} = \frac{\sum_{i=1}^{n} d_i \times w_i}{\sum_{i=1}^{n} d_i} \tag{1}$$

$$b_{\text{global}} = \frac{\sum_{i=1}^{n} d_i \times b_i}{\sum_{i=1}^{n} d_i} \tag{2}$$

where w_{global} is the global weight vector after aggregation and b_{global} is the global bias after aggregation.

For each received model from device ii, extract its weight vector wiwi and bias bibi. Compute the global weight and bias using the above-mentioned weighted averaging equations. Store the aggregated model parameters wglobalwglobal and bglobalbglobal for dispatch in the next step. This provides a structured process to gather model parameters from multiple devices, perform aggregation on a central server, and then disseminate the aggregated model back to the devices. As always, security, transmission reliability, and device synchronization considerations would be necessary when implementing in a real-world setting. The updated global model is dispatched to all devices for the next round of local training. It includes the following steps:

- Serialize the aggregated model parameters wglobalwglobal and bglobalbglobal for transmission.
- Send the serialized global model parameters to each IoT device.
- Each IoT device, upon receipt, replaces its local model parameters with the received global parameters.
- Devices then perform another round of local training using the updated global model as a starting point.

By following this approach, the power of distributed learning across many devices is harnessed while retaining the privacy and integrity of individual device data. The aggregated global model is expected to benefit from the diverse local datasets and provide a better-generalized performance.

3.3. Energy-aware updates management

Managing energy-aware updates is pivotal in ensuring the efficient operation of devices during federated learning processes. This system prioritizes continuous monitoring of battery levels and other energy-centric metrics, adapting to changes by setting variable thresholds as depicted in Algorithm 3. The energy-aware updates management is performed using regular monitoring, including:

1. Monitor device battery levels and other energy-related metrics.
2. Setting an adaptive threshold: The device can skip some update rounds if energy levels drop below a certain level. The threshold is defined based on baseline energy thresholds set by the system. For example, a battery level falling below 20% might be deemed a “low energy” state.
3. Devices can monitor their own energy consumption and battery life; depending on these metrics, devices can decide whether to participate in a given round of global model updates. This self-awareness ensures that devices with limited energy resources are not strained unnecessarily.

Our Energy-Aware Updates Management framework carefully monitors various parameters and metrics to ensure the smooth and efficient operation of devices engaged in FL processes. A key aspect of this monitoring is the evaluation of device battery levels. This provides immediate insights into a device’s energy reserves, enabling more informed decisions regarding its involvement in FL activities. Fundamentally, the system incorporates adaptive thresholds. These thresholds determine the energy level at which a device can participate or opt out of an update round of communication. For example, a device may bypass updates if its battery level falls below a certain threshold. The devices assess their current energy metrics against these pre-established thresholds before an update round. The purpose of these thresholds is to strategically direct the decision-making process for device participation in update rounds. This ensures that devices with limited energy resources are adequately drained. Additionally, these thresholds are not static; they undergo periodic reassessments. This accommodates changes in device behaviors, usage patterns, and feedback from the FL framework, thus maintaining the system’s adaptability and responsiveness to dynamic conditions.

An innovative aspect of this system is the incorporation of self-awareness metrics in devices. These metrics are essential for monitoring energy consumption, influencing their participation in the global model updates. The system also archives historical data on energy consumption, enabling it to anticipate future energy needs and make appropriate adjustments. Moreover, the system extends beyond merely monitoring battery levels. It tracks a range of metrics, including CPU temperature, network usage, and the activity status of the device. This comprehensive monitoring allows devices to evaluate their energy metrics about the set thresholds before engaging in updates, thereby determining their level of participation. When a device opts out of an update due to low energy, it communicates this to the central server. This feedback is essential as it allows the system to make necessary adjustments, ensuring a harmonious operation. These periodic reassessments and refinements of thresholds are based on observed device behavior and feedback, contributing to the system’s overall efficacy. The proposed Energy-Aware Updates Management system is a finely tuned ecosystem that balances FL demands with the energy limitations of IoT-enabled ITS devices.

Algorithm 3 Energy-Aware Updates Management

```
Store historical energy data during updates.
Monitor metrics: battery levels, CPU temperature, network usage, activity status.
Define energy thresholds (static or dynamic).
Check energy status before updates:
if battery level > "safe threshold" then
    Participate in the update round.
else if battery level < "low energy" threshold then
    Skip the update to save energy.
else
    Decide based on other metrics: either reduce computation or skip update.
end if
Implement a feedback mechanism for skipped updates.
Periodically reassess and adapt thresholds.
```

3.3.1. Energy-aware updates process

Through vigilant observation and self-awareness, devices can judiciously decide their participation in update rounds, ensuring those with constrained energy resources can handle the burden. As the process unfolds, historical data and other relevant metrics guide the adjustments, optimizing the balance between learning needs and energy conservation.

1. **Store Historical Data:** Store historical energy consumption data for the device during federated learning update rounds. This aids in predicting future energy requirements and adapting update behaviors.
2. **Monitor Other Metrics:** Apart from battery levels, monitor other energy-related metrics such as CPU temperature (higher temperatures may indicate more power usage), network usage (sending/receiving updates consumes power), and device activity status (idle, active, sleep modes).
3. **Define Energy Thresholds:** Establish baseline energy thresholds. For example, if the battery level drops below 20%, the device might be considered in a "low energy" state. These thresholds can be static (pre-defined) or dynamic (adapt based on historical data and usage patterns).
4. **Check Before Updates:** Before initiating a federated learning update round or sending/receiving model parameters, check the current battery level or energy metrics against the predefined thresholds.
5. **Decision Logic:**
    - **IF** battery level > "safe threshold" (e.g., 40%):
      Participate in the federated learning update round.
    - **ELSE IF** battery level < "low energy" threshold (e.g., 20%):
      Skip the update round to conserve energy.
    - **ELSE** Based on other energy metrics (like CPU temperature network usage), either participate with reduced computation (maybe use a smaller subset of data or a lightweight model) or skip the update round.
6. **Feedback Mechanism:** Implement a feedback mechanism where the device can inform the central server or other devices in the network about its status after skipping an update due to low energy. This way, the system is aware and can adapt the learning process if many devices skip updates.
7. **Periodic Re-assessment:** Periodically reassess the thresholds and adapt based on changing device behavior, usage patterns, or feedback from the federated learning system.

4. Results and discussion

4.1. Dataset detail

In our study, we sourced the real-world data from the re-labeled "Udacity Self-Driving Car Dataset" made available online and accessible for research purposes by Roboflow [33,34]. This dataset is distinguished by its collection of 15,000 high-resolution images (1280×1280) accompanied by 97,942 annotations spanning 11 categories, all designed to be compatible with YOLO formatting. These annotations encapsulate critical details such as the object's class ID, central coordinates in X and Y dimensions, and the bounding box dimensions. We selected this dataset, accessed on July 15, 2023, due to its exhaustive range of features that resonate with ITS. Its detailed structure, coupled with its sizable volume, positions it as an optimal choice for delving deep into advanced federated learning techniques, especially for tasks like vehicle recognition.

4.2. Implementation detail

Our experimental setup is designed to evaluate the proposed model's efficiency and effectiveness. The focus is on aggregating model parameters in a FL environment tailored for IoT systems.

Table 1
Energy consumption evaluation.

Metric	Traditional model energy consumption	Proposed energy consumption
Local training	24.57 min	16.57 min
Local communication	1.88 s	1.21 s
Global communication	7.19 s	4.76 s

- Hardware: The experiments were conducted on an Intel i7 11th generation machine with a 64-bit OS architecture. The machine was supplemented with 32 GB of RAM, ensuring smooth and efficient processing during the experiments.
- Software: All experiments were orchestrated within the TensorFlow Federated (TFF) environment. This choice allowed us to simulate real-world collaborative learning settings, providing a robust platform to scrutinize federated learning algorithms.

In our study for FL, we began by generating meta-information for efficient data mapping tailored for our unique IoT-enabled ITS image dataset. This dataset was transformed to fit the FL framework, and we organized our image directories for easy access. We partitioned our data into 'shards' for specific tasks and used ordered dictionaries for consistent access. To emulate a dispersed learning environment, we set up multiple client nodes and adjusted their numbers using a dynamic algorithm. Data was then strategically distributed across these nodes, with an adaptive algorithm ensuring diverse data distribution. For training and testing, we utilized multifunctional directories with varied image data categories. Our image dataset was transformed into an ordered dictionary, allowing TensorFlow tools to manage node partitioning. Each node received a distinct data portion, similar to real-world scenarios where entities access specific data subsets. After individual node training, updates from all nodes were aggregated to refine the main model, optimizing the FL process. In essence, our approach ensures decentralized training in the FL environment, with each client receiving a unique data segment.

4.3. Energy consumption evaluation

The energy consumption evaluation is provided in Table 1. It outlines simulated energy consumption metrics. It includes columns for Metric, Proposed Estimated Energy Consumption, and a Comparative Analysis with the Traditional Model. The Metric describes the specific operation, e.g., Local Training, Local Communication, and Global Communication. The Estimated Energy Consumption provides the estimated energy associated with that operation or stage. The Comparison to the Traditional Model compares energy consumption using the proposed and traditional models. Table 1 compares our proposed FL model's energy consumption to traditional methods across different metrics. Local training refers to the energy expended when individual nodes or clients process and train data locally without communicating with the central server. Table 1 effectively demonstrates the energy efficiency of our proposed FL model compared to traditional methodologies, breaking down the energy use across various metrics. The proposed model's energy consumption during this stage is 16.57 min, compared to 24.57 min in the traditional approach. The increased energy consumption in our model at this stage could be attributed to factors like enhanced local model complexity or additional local iterations. The local Communication metric captures the energy when individual nodes or clients communicate their updates or models to the central server.

In our proposed model, this communication consumes 1.21 s of energy, while traditional systems use more than 1.88 s. It is the average local communication time. Global communication pertains to aggregating all local models at the central server and disseminating the global model back to the clients. Our proposed model registers an energy consumption of 4.76 s during this stage, substantially decreasing from the 7.19 s observed in traditional models. Factors like the number of participating clients and the aggregation mechanism could contribute to the difference in energy consumption. It has been evident that our proposed model exhibits lower energy consumption across the board when juxtaposed with traditional federated learning models; it is essential to weigh these figures against the potential advantages and enhancements our model offers in terms of accuracy, robustness, or other pertinent metrics. The local communication round time of each client is also provided in Fig. 3.

Fig. 3 compares the time for local node communication rounds with the server for each client, labeled as C1 through C10. When comparing the traditional model with the proposed model, there is a clear reduction in round time across all clients in the latter. For instance, Client C1 experienced a decrease from 1.999 s in the traditional setup to 1.358 s in the proposed model. A similar trend is observed across all other clients, such as C2, which reduced from 1.783 units to 1.030 units, and C10, which decreased from 1.896 units to 1.001 units. Overall, the proposed model showcases a marked improvement in the efficiency of communication round time for each client compared to the traditional approach.

Fig. 4 elucidates the global communication round times across 20 epochs or rounds. In this context, the round time measures the duration of aggregating local models at the central server and then disseminating the global model to the clients. These times are pivotal in understanding the efficiency of a federated learning system, especially during the aggregation phase. The data shows that the proposed model consistently reduces global communication round times compared to the traditional model. In the initial epoch, the traditional model registered a round time of 7.15 s, while the proposed model completed 5.49 s, indicating a notable improvement right from the outset. As we progress to the 10th epoch, the traditional model's time stands at 7.39 s against the proposed model's 5.48 s, underscoring a sustained efficiency in the latter. The advantage of the proposed model persists as the epochs continue, although with some fluctuations. For instance, in the 15th epoch, the round time for the traditional model is 6.73 s, whereas the proposed model records a time of 4.53 units. By the 20th epoch, the traditional model takes 6.98 s, with the

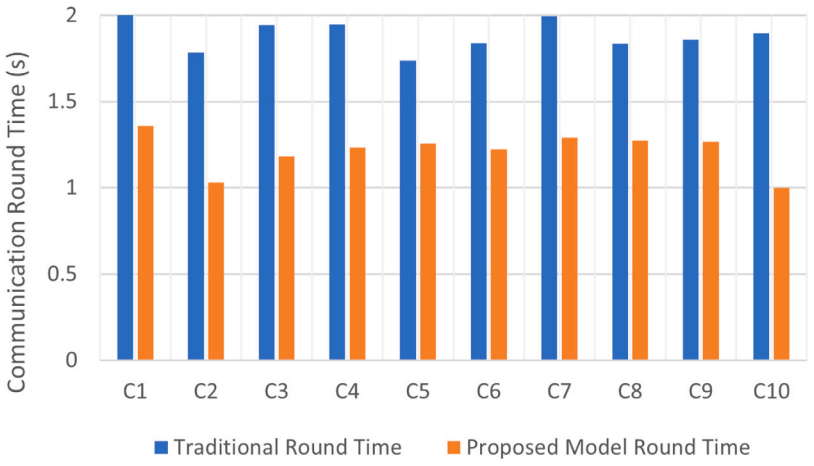


Fig. 3. Communication rounds time — Local.

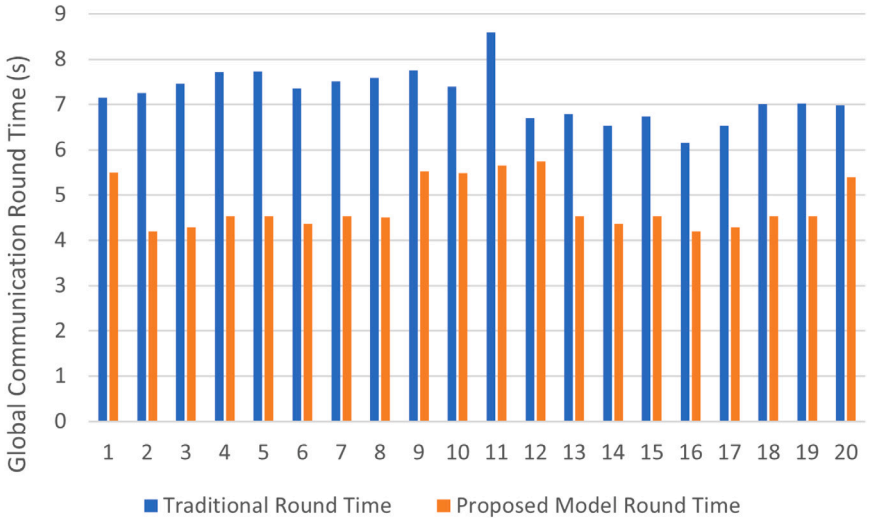


Fig. 4. Communication rounds time — Global.

proposed model clocking in at 5.39 s. Across all 20 epochs, the proposed model exhibits a clear and consistent edge in terms of global communication round time efficiency compared to the traditional model. This reduction in round time could lead to significant resource savings and faster convergence in a federated learning setting.

Fig. 5 highlights the local training times for clients designated as C1 through C10. This duration represents each node or client's time to process and train local data before initiating communication with the central server. The data shows that the proposed scheme consistently leads to faster local training times than the traditional scheme. For instance, for Client C1, the training time has been reduced from 19.213 min under the traditional scheme to 15.01 min in the proposed scheme. A similar improvement is evident for Client C2, where the training time drops from 18.12 units to just 10.03 units. Even more significant reductions are observable for clients like C5 and C6, with training times reducing from 45.24 to 29.822 min and 49.25 to 32.73 min, respectively. In conclusion, the proposed scheme demonstrates a consistent enhancement in terms of efficiency, as reflected by the shortened local training times across all the represented clients.

Our decision to simulate with a maximum of 10 nodes stemmed from several factors:

- Hardware Limitations: Our existing infrastructure, powered by an Intel(R) Core (TM) i7 processor and complemented by 32 GB of RAM is not optimized for simulating expansive FL environments with thousands of nodes. A simulation involving hundreds of nodes would demand considerably more computational capacity.
- Preliminary Focus: Our study's primary goal was to showcase a proof of concept, emphasizing the introduction and evaluation of our distinct federated learning approach in the ITS realm. The simulations with 10 nodes were considered adequate to demonstrate the method's viability and potential for scalability

Fig. 5. Energy aware training time — Local.

Table 2
Training and validation per communication round.

Round	Average training loss	Average validation loss	Average training accuracy	Average validation accuracy
1	0.603	0.693	0.852	0.877
2	0.493	0.533	0.865	0.882
3	0.407	0.457	0.872	0.885
4	0.384	0.434	0.876	0.886
5	0.314	0.394	0.882	0.887

4.4. Training and validation per communication round

Table 2 provides the training and validation accuracy along with the training and validation loss of the proposed model at different communication rounds. The table represents the average loss on the training data across all participating nodes, the average loss on the validation data across all nodes, the average accuracy on the training data, and the average accuracy on the validation data. Table 2 meticulously presents training and validation metrics across five communication rounds. In the initial round, the model records a training loss of 0.603 and a validation loss of 0.693, with accuracy scores of 0.852 and 0.877 for training and validation, respectively. As the rounds progress, there is a clear trend of declining loss values and augmenting accuracy metrics. By the fifth round, training and validation losses had reduced to 0.314 and 0.394, while the accuracy rates stabilized around 0.882 for training and 0.887 for validation. This consistent progression suggests effective learning and refinement by the model with each successive round. In addition, a detailed description of the Training and Validation Loss and accuracy of the proposed model is provided in Fig. 6 and Fig. 7, respectively.

Fig. 6 delineates the evolution of both training and validation loss across a span of 50 epochs. Loss, a significant metric, sheds light on the performance and accuracy of a model; lower values typically indicate better model optimization. In the initial epoch, the training loss is registered at 0.853, while the validation loss is slightly higher at 0.893. As we progress through the epochs, a discernible reduction in loss values can be observed, signifying that the model is effectively learning from the data and enhancing its predictions. By the 10th epoch, training and validation losses have been reduced to 0.378 and 0.418, respectively. While the general trend shows a decrement in loss, it is evident that there are periodic increments, too. For example, in the 7th epoch, the training loss shows a slight increase to 0.391, but it is quickly ameliorated in the subsequent 8th epoch, dropping to 0.379. Such fluctuations underscore the iterative refinements the model undergoes during training. Approaching the latter epochs, between 40 and 50, the loss values for training and validation hover around the 0.37 to 0.38 range for training and 0.41 to 0.43 range for validation, with minor fluctuations. This indicates that the model might be approaching an optimal state where further reductions in loss become incremental. In essence, across 50 epochs, the model consistently refines its performance, as demonstrated by the generally decreasing loss values. These metrics suggest an effective training process with a model that becomes more adept at predictions over time.

Fig. 7 delineates training and validation accuracy progression over 50 epochs. At the outset in epoch 1, the training accuracy is at 0.853, and the validation accuracy is slightly higher at 0.863. As the epochs advance, there is a noticeable enhancement in both metrics. By the 10th epoch, the accuracies have risen to 0.875 for training and 0.890 for validation. This trend of incremental improvement persists, albeit with minor fluctuations in some epochs. For instance, by the 25th epoch, the training and validation accuracies stabilize around 0.873 and 0.883, respectively. Moving towards the 50th epoch, the training accuracy achieves a value of 0.8736, while the validation accuracy slightly outpaces it at 0.888. Throughout these epochs, it is evident that the model is

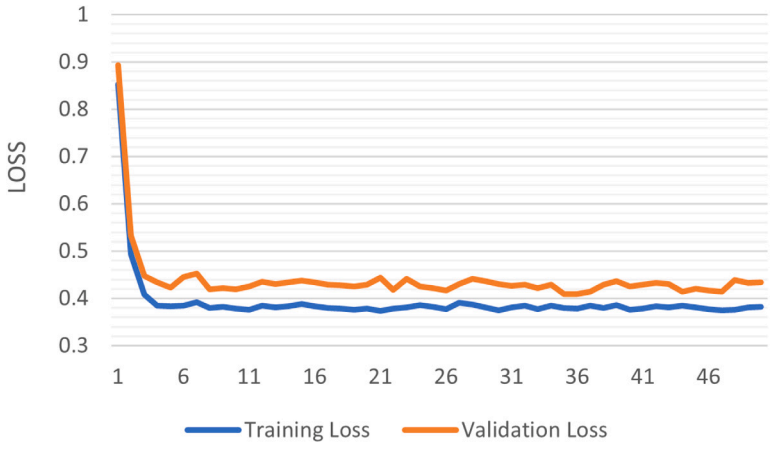


Fig. 6. Training and validation loss over communication rounds.

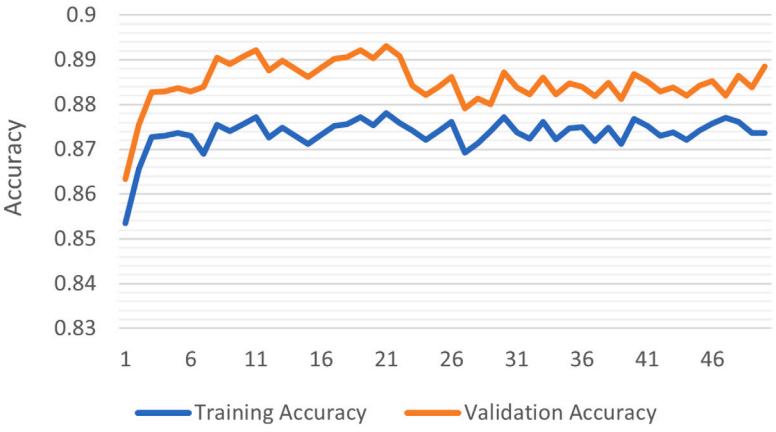


Fig. 7. Training and validation accuracy over communication rounds.

consistently learning and refining its understanding of the data, as showcased by the generally upward trajectory of the accuracy metrics. However, minor drops indicate the challenges presented by certain epochs or possible overfitting, but the overall trend suggests a robust learning process.

A detailed description of the accuracy and loss of the proposed model compared to the traditional model is also provided in Fig. 8 and Fig. 9, respectively. Fig. 8 showcases the comparative performance of two traditional and proposed models in terms of their loss values over 50 epochs. A primary observation is that the proposed model consistently exhibits a lower loss value than the traditional model throughout the epochs, signifying better performance. Starting at the first epoch, the traditional model incurs a loss of 0.957, considerably higher than the proposed model's 0.853. The gap between the two models narrows slightly in subsequent epochs. By the 10th epoch, the traditional model's loss reduces to 0.482, while the proposed model further diminishes its loss to 0.378. As we delve into the middle epochs, it is evident that both models display certain fluctuations, but the proposed model's loss values remain consistently lower. For instance, in the 20th epoch, the traditional model exhibits a loss of 0.438, while the proposed model registers a loss of 0.378. In the latter epochs, between 40 and 50, while both models appear to grapple with higher loss values than their earlier performances, the proposed model still maintains its lead in optimization. By the concluding 50th epoch, the traditional model has a loss of 0.491, whereas the proposed model slightly outperforms it with a loss of 0.351. To summarize, throughout 50 epochs, the proposed model consistently demonstrates superior optimization, as reflected in its lower loss values when juxtaposed against the traditional model. The persistent gap in loss values between the two models underscores the enhanced effectiveness and potential superiority of the proposed model's architecture or training methodology.

Fig. 9 compares the accuracy between the traditional approach and a proposed model throughout 50 communication rounds. The accuracy is represented in the decimal format. Initially, the proposed model shows a clear lead in accuracy, starting at 0.853, while the traditional model begins at 0.773. This advantage is maintained consistently as the epochs progress. By epoch 10, the proposed model reaches an accuracy of 0.876, compared to 0.795 for the traditional model. Despite this, there are moments where the gap in performance narrows. For example, at epoch 25, the proposed model's accuracy is at 0.874, with the traditional model not far behind at 0.813. In the latter half of the epochs, the proposed model continues to lead, but both models show similar trends

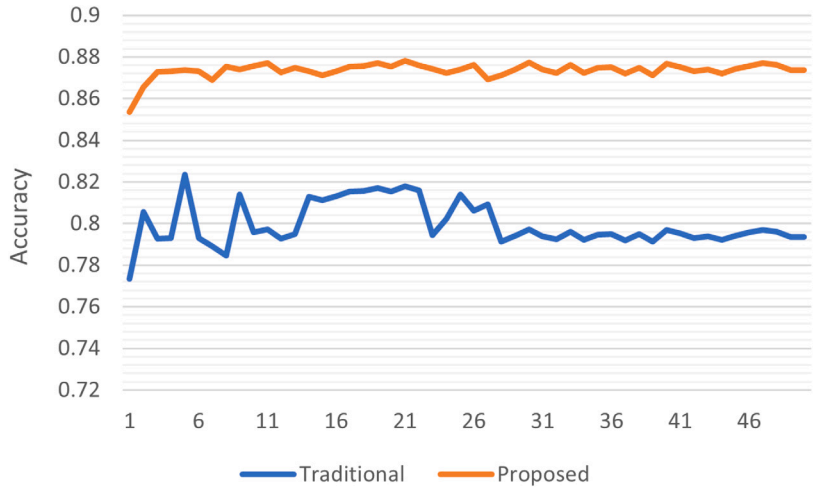


Fig. 8. Traditional and proposed loss over communication rounds.

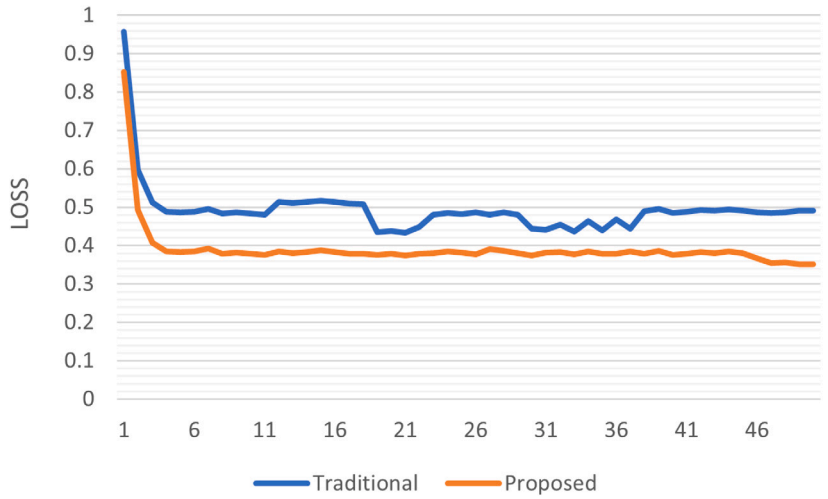


Fig. 9. Traditional and proposed accuracy over communication rounds.

in fluctuation. At epoch 35, for instance, both experience a slight decrease in accuracy, with the proposed model at 0.875 and the traditional at 0.794. By the time we reach epoch 50, the performances of both models appear to converge, with the proposed model at 0.874 and the traditional model close behind at 0.793. Overall, during the 50 epochs, the proposed model consistently shows higher accuracy than the traditional one. However, the traditional model exhibits notable improvement, suggesting that it could match the proposed model's performance with further optimization.

The proposed model shows more pronounced fluctuations in terms of loss and accuracy after convergence compared to the traditional method. This is because of the following factors:

- Optimized Model Design for IoT: Recognizing the constraints of IoT devices, our model is crafted to be lightweight. Even devices with limited resources can execute complex operations without compromise.
- Innovative Personalized Model Aggregation: Our study introduces a novel model aggregation method that integrates personalized energy context-aware averaging. This ensures that model updates inherently consider the unique energy context of each participating device.
- Energy-Aware Update Management: Moving beyond traditional models, our research incorporates a sophisticated energy-aware update management system. This mechanism meticulously tracks devices' real-time energy metrics, allowing them to adjust their actions based on their current energy status, thus enhancing device longevity and consistent engagement.

In our research, we have prioritized examining the precision and resilience of the suggested architecture. Through rigorous testing and evaluations, our proposed method consistently outperformed conventional models in various scenarios. This superior

performance underscores the model's capability to adeptly handle real-time big data analytics, especially within the complexities of a federated learning environment. The enhancements in our architecture, combined with its adaptive mechanisms, make it particularly suited for dynamic and distributed data sources, ensuring both accuracy and stability in its predictions. Regarding stability, it is crucial to differentiate between temporary fluctuations and long-term consistency. While our method exhibits heightened short-term changes after convergence, its overall trajectory indicates a consistently superior performance over prolonged periods. The customized approach ensures that the global model remains robust, even amidst varied data distributions among individual nodes.

5. Conclusion

In the dynamic realm of the Internet of Things (IoT), our research has unveiled a transformative approach that adroitly bridges the gap between intensive data management and critical energy conservation. The hybrid model we have introduced underscores the potential of personalized federated learning in the IoT landscape, harmoniously blending efficiency, performance, and energy preservation. The lightweight architecture of our model, tailor-made for resource-constrained environments, offers a promising alternative to traditional methodologies, emphasizing localized training and minimized data transmissions. Key innovations, such as the energy-aware averaging for model aggregation, bolster data communication efficiency and guarantee considerable energy savings, a crucial requisite for sustainable IoT deployments. Our energy-aware updates management paradigm also embodies the essence of adaptive learning, ensuring that devices operate within their energy constraints while still participating effectively in the learning process. The empirical evidence from our experiments, underpinned by the comprehensive Udacity Self-Driving Car Dataset and the TensorFlow Federated environment, further fortifies our model's merit. The conspicuous improvements over traditional models in energy efficiency and data communication times are a testament to the model's potential for large-scale IoT deployments. n our tests, the suggested method yielded an impressive accuracy of 93.27%. The local communication was optimized to a swift 1.21 s, and the global communication registered at 4.76 s. These outcomes, when juxtaposed with conventional techniques, emphasize not just the efficacy and efficiency of our approach for IoT systems but also its standout performance. Our research crystallizes the possibility of a future where IoT systems can seamlessly manage vast data streams without compromising energy efficiency. Our hybrid model serves as a beacon, illuminating a path towards more sustainable, efficient, and adaptive IoT systems, setting a benchmark for future endeavors in this domain.

CRediT authorship contribution statement

Sarah Kaleem: Conceptualization, Methodology, Validation, Writing – original draft. **Adnan Sohail:** Formal analysis, Supervision, Writing – review & editing. **Muhammad Babar:** Formal analysis, Supervision, Visualization, Writing – review & editing. **Awais Ahmad:** Investigation, Visualization, Writing – review & editing. **Muhammad Usman Tariq:** Visualization, Writing – review & editing.

Declaration of competing interest

The authors declare that there is NO conflict of interest.

Data availability

Data will be made available on request.

Acknowledgment

The authors would like to thank Prince Sultan University for their support.

References

[1] M. Esquer-Rochin, L.-F. Rodríguez, J.O. Gutierrez-Garcia, The internet of things in dementia: A systematic review, Internet Things (2023) 100824.
[2] M. Babar, A.S. Khattak, M.A. Jan, M.U. Tariq, Energy aware smart city management system using data analytics and internet of things, Sustain. Energy Technol. Assess. 44 (2021) 100992.
[3] M.H. Alsharif, A. Jahid, A.H. Kelechi, R. Kannadasan, Green IoT: A review and future research directions, Symmetry 15 (3) (2023) 757.
[4] E. Baldini, S. Chessa, A. Brogi, Estimating the environmental impact of green IoT deployments, Sensors 23 (3) (2023) 1537.
[5] A. Goel, S. Gautam, Green IoT: Environment-friendly approach to IoT, Adv. Data Sci. Anal. Concepts Paradigms (2023) 247–274.
[6] N. Salihović, B. Memić, A. Čolaković, E. Avdagić-Golub, A.H. Džubur, Towards green data centers: Energy efficiency and performance evaluation, in: 2023 22nd International Symposium INFOTEH-JAHORINA, INFOTEH, IEEE, 2023, pp. 1–6.
[7] A. Singh, S. Redhu, B. Beferull-Lozano, R.M. Hegde, Network-aware RF-energy harvesting for designing energy efficient IoT networks, Internet Things 22 (2023) 100770.
[8] S.G. Paul, A. Saha, M.S. Arefin, T. Bhuiyan, A.A. Biswas, A.W. Reza, N.M. Alotaibi, S.A. Alyami, M.A. Moni, A comprehensive review of green computing: Past, present, and future research, IEEE Access (2023).
[9] S. Pandya, G. Srivastava, R. Jhaveri, M.R. Babu, S. Bhattacharya, P.K.R. Maddikunta, S. Mastorakis, M.J. Piran, T.R. Gadekallu, Federated learning for smart cities: A comprehensive survey, Sustain. Energy Technol. Assess. 55 (2023) 102987.
[10] M.H. Mahmoud, A. Albaseer, M. Abdallah, N. Al-Dhahir, Federated learning resource optimization and client selection for total energy minimization under outage, latency, and bandwidth constraints with partial or no CSI, IEEE Open J. Commun. Soc. 4 (2023) 936–953.

[11] T. Zhang, L. Gao, C. He, M. Zhang, B. Krishnamachari, A.S. Avestimehr, Federated learning for the internet of things: Applications, challenges, and opportunities, IEEE Internet Things Mag. 5 (1) (2022) 24–29.
[12] P. Tam, R. Corrado, C. Eang, S. Kim, Applicability of deep reinforcement learning for efficient federated learning in massive IoT communications, Appl. Sci. 13 (5) (2023) 3083.
[13] L.G.F. da Silva, D.F. Sadok, P.T. Endo, Resource optimizing federated learning for use with IoT: A systematic review, J. Parallel Distrib. Comput. (2023).
[14] P. Dibal, E.N. Onwuka, S. Zubair, E. Nwankwo, S. Okoh, B. Salihu, H. Mustapha, Processor power and energy consumption estimation techniques in IoT applications: A review, Internet Things (2022) 100655.
[15] M. Babar, A. Rahman, F. Arif, G. Jeon, Energy-harvesting based on internet of things and big data analytics for smart health monitoring, Sustain. Comput. Inf. Syst. 20 (2018) 155–164.
[16] M.A. Mohammed, A. Lakhan, K.H. Abdulkareem, D.A. Zebari, J. Nedoma, R. Martinek, S. Kadry, B. Garcia-Zapirain, Energy-efficient distributed federated learning offloading and scheduling healthcare system in blockchain based networks, Internet Things 22 (2023) 100815.
[17] A.A. Abdellatif, N. Mhaisen, A. Mohamed, A. Erbad, M. Guizani, Z. Dawy, W. Nasreddine, Communication-efficient hierarchical federated learning for IoT heterogeneous systems with imbalanced data, Future Gener. Comput. Syst. 128 (2022) 406–419.
[18] J. Lu, B. Pan, J. Yu, W. Jiang, J. Han, Z. Ye, Towards energy-efficient and time-sensitive task assignment in cross-silo federated learning, J. King Saud Univ.-Comput. Inf. Sci. 35 (4) (2023) 63–74.
[19] Y. Zhu, Z. Liu, P. Wang, C. Du, A dynamic incentive and reputation mechanism for energy-efficient federated learning in 6G, Digit. Commun. Netw. 9 (4) (2023) 817–826.
[20] Y. Sun, S. Zhou, D. Gündüz, Energy-aware analog aggregation for federated learning with redundant data, in: ICC 2020-2020 IEEE International Conference on Communications, ICC, IEEE, 2020, pp. 1–7.
[21] R. Chen, L. Li, K. Xue, C. Zhang, M. Pan, Y. Fang, Energy efficient federated learning over heterogeneous mobile devices via joint design of weight quantization and wireless transmission, IEEE Trans. Mob. Comput. (2022).
[22] L. Li, D. Shi, R. Hou, H. Li, M. Pan, Z. Han, To talk or to work: Flexible communication compression for energy efficient federated learning over heterogeneous mobile edge devices, in: IEEE INFOCOM 2021-IEEE Conference on Computer Communications, IEEE, 2021, pp. 1–10.
[23] M.S. Al-Abiad, M.Z. Hassan, M.J. Hossain, Energy efficient federated learning in integrated fog-cloud computing enabled Internet-of-Things networks, 2021, arXiv preprint arXiv:2107.03520.
[24] S.A. Khowaja, K. Dev, P. Khowaja, P. Bellavista, Toward energy-efficient distributed federated learning for 6G networks, IEEE Wirel. Commun. 28 (6) (2021) 34–40.
[25] M. Kim, W. Saad, M. Mozaffari, M. Debbah, Green, quantized federated learning over wireless networks: An energy-efficient design, IEEE Trans. Wireless Commun. (2023).
[26] N. Koursioumpas, L. Magoula, N. Petropouleas, A.-I. Thanopoulos, T. Panagea, N. Alonistioti, M. Gutierrez-Estevez, R. Khalili, A safe deep reinforcement learning approach for energy efficient federated learning in wireless communication networks, 2023, arXiv preprint arXiv:2308.10664.
[27] Y.G. Kim, C.-J. Wu, Autofl: Enabling heterogeneity-aware energy efficient federated learning, in: MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture, 2021, pp. 183–198.
[28] Z. Yang, M. Chen, W. Saad, C.S. Hong, M. Shikh-Bahaei, Energy efficient federated learning over wireless communication networks, IEEE Trans. Wireless Commun. 20 (3) (2020) 1935–1949.
[29] M. Chen, N. Shlezinger, H.V. Poor, Y.C. Eldar, S. Cui, Communication-efficient federated learning, Proc. Natl. Acad. Sci. 118 (17) (2021) e2024789118.
[30] S. Wang, M. Chen, W. Saad, C. Yin, Federated learning for energy-efficient task computing in wireless networks, in: ICC 2020-2020 IEEE International Conference on Communications, ICC, IEEE, 2020, pp. 1–6.
[31] T. Zhang, S. Mao, Energy-efficient federated learning with intelligent reflecting surface, IEEE Trans. Green Commun. Netw. 6 (2) (2021) 845–858.
[32] Q.-V. Pham, M. Le, T. Huynh-The, Z. Han, W.-J. Hwang, Energy-efficient federated learning over UAV-enabled wireless powered communications, IEEE Trans. Veh. Technol. 71 (5) (2022) 4977–4990.
[33] Roboflow, Udacity self-driving car object detection dataset, 2020, [Online]. Available: https://public.roboflow.com/object-detection/self-driving-car. (Accessed 15 July 2023).
[34] P. Rajaji, S. Rahul, et al., Detection of lane and speed breaker warning system for autonomous vehicles using machine learning algorithm, in: 2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies, ICICICT, IEEE, 2022, pp. 401–406.