

IDMN-SP: Improved Deep Maxout Network with Adaptive Structured Pruning for Resource-Efficient Federated Learning in IoT Big Data Analytics

CH. Ellaji ^{a*}, R.S. Ponmagal ^b, V. Saritha ^c

^{a*} *Research Scholar, Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Chennai, Tamil Nadu, India. E-mail: ce6149@srmist.edu.in, Orcid: <https://orcid.org/0009-0004-8696-1466>*

^b *Department of Computing Technologies, Faculty of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chennai, Tamil Nadu, India. E-mail: ponmagas@srmist.edu.in, Orcid: <https://orcid.org/0000-0002-7228-1310>*

^c *Department of Computer Science and Engineering, School of Engineering and Technology, Sri Padmavati Mahila Visvavidyalayam, Tirupati, Andhra Pradesh, India. E-mail: cseheadspmvv@gmail.com, Orcid: <https://orcid.org/0000-0002-9658-3663>*

Abstract

The exponential growth of Internet of Things (IoT) ecosystems generates massive heterogeneous big data, requiring efficient, privacy-preserving analytics that operate under strict edge resource constraints. Current federated learning (FL) frameworks often struggle with high computational costs, poor bandwidth, and ineffective pruning or ReLU activation bottlenecks that trigger severe accuracy losses and dead neurons. To address these challenges, this paper introduces the Improved Deep Maxout Network with Adaptive Structured Pruning (IDMN-SP), a lightweight federated learning architecture designed for resource-efficient IoT big data analytics. IDMN-SP features two core innovations: replacing traditional ReLU with a Bendable Linear Unit (BLU) function to eliminate dead neurons and enhance feature representation, and an adaptive gradient-based structured pruning mechanism employing normalized L2-norm thresholding to selectively remove redundant convolutional filters without discarding key features, paired with a targeted fine-tuning recovery phase. Evaluated on the CIFAR-10 benchmark dataset under non-IID conditions ($\beta = 0.5$), the proposed framework achieves an outstanding classification accuracy of 84.73%, a Precision of 84.41%, a Recall of 84.59%, and an F1-Score of 84.50%. Moreover, IDMN-SP drastically improves resource efficiency by reducing model complexity to 24.19 million FLOPs, shrinking model size to 1.94 MB, and lowering communication bandwidth to 1.94 MB per round, thereby outperforming baseline methods like FedDA by 4.56 percentage points in accuracy. These results demonstrate that IDMN-SP successfully bridges the gap between high predictive accuracy and strict resource conservation, making it an optimal solution for edge-based IoT environments.

Keywords: Federated Learning (FL), Internet of Things, Adaptive Structured Pruning, Deep Maxout Network, IoT Big Data Analytics.

Introduction

The recent trend of exponentially growing Internet of Things (IoT) ecosystems has radically altered the data generation processes across the various industry application domains such as smart healthcare, smart transportation, industrial automation, and precision agriculture, with billions of interconnected devices ever-generating vast amounts of heterogeneous big data which demand efficient and scalable analytics frameworks which have to operate under strict resource limits at the network edge (Wu et al., 2023; Li et al., 2024). These demands have been exacerbated by the growing complexity of deep learning models needed to generate meaningful IoT big data analytics, with the current neural network architectures consuming large amounts of computational resources, memory resources, and energy demand that are many times higher than most resource-constrained IoT edge devices have (Jiang et al., 2022; Yu et al., 2025). The inherent latency, high bandwidth usage, and sensitive data privacy implications of conventional centralized cloud-based analytics models make them both fundamentally inappropriate for high-sensitivity real-world Internet of Things applications where data locality and policy-governed data security are crucial issues (Mughal et al., 2024; Ye et al., 2020). These complexity issues have raised the pressing demand for lightweight, privacy-preserving, and resource-efficient analytics frameworks that can run directly on the network edge without affecting model accuracy or data safety (Chen et al., 2023).

Federated learning (FL) has become an interesting distributed training concept, which specifically attempts to resolve these privacy and communication issues by facilitating collaborative model learning with many edge devices geographically distributed but not exchanging raw information with a central aggregation server (Lee & Jang, 2025). Following the conventional FL model, every involved client trains a local model on its own privately collected data, and only the model parameters need to be shared with the central server to be aggregated on a global basis, resulting in data and privacy locality and a substantial privacy exposure reduction in distributed IoT scenarios (Chang et al., 2024). Such a decentralized scale has proven to show significant practical potential throughout an extensive range of IoT applications, including localized yield prediction in farming systems and autonomous decision-making in resource-constrained edge-IoT networks (Tariq et al., 2025). Moreover, FL frameworks have been more co-designed with edge computing infrastructure to introduce computation physically nearer to data sources, minimizing the transmission latency and facilitating quicker model convergence in bandwidth-restricted distributed IoT implementations.

In spite of its huge privacy-saving benefits, FL is characterized by serious and constant issues that impose tangible impediments to successful application on resource-limited IoT devices. One of the biggest obstacles is that the scale of computation required to train local models poses a significant burden on the resources of client devices because IoT endpoints are often very low-power processors with very limited memory and power resources, and training large-scale neural networks is even computationally intractable (Niu et al., 2024). Also, the model updates between the central server and potentially thousands of participating clients are transmitted many times with significant bandwidth consumption, causing acute communication bottlenecks, especially under the conditions of a wireless network, as the reliability of connections as well as bandwidth are constrained by the network parameters (Liu et al., 2023). All these bottlenecks of computation and communication efficiency are the most severe challenges to scalable and viable FL applications at the IoT edge, and how to tackle them jointly in a single framework remains one of the most open research problems (Cao et al., 2025; Jiang et al., 2023).

Another crucial issue with deployments of FL in practice is that data is not independently and identically distributed (non-IID) between spatially dispersed IoT client devices. In contrast to idealized assumptions of homogeneous data distribution, real-world IoT implementations have noticeable statistical heterogeneity among clients, where each individual device gathers information that is indicative of its localized environment, usage, and sensor properties. This widespread data heterogeneity leads to a situation where client model updates are far apart in the local training phase and makes global model aggregation far more difficult, often yielding worse convergence behavior and greatly decreased end model accuracy than centralized training baselines (Chen et al., 2024). Strong and efficient FL systems to process IoT big data need to be specifically tailored to be able to sustain stable performance even in non-IID data distributions, as well as to meet the high resource requirements of heterogeneous edge devices (Liu et al., 2024; Ma et al., 2022).

Model compression methods have been explored extensively to reduce computation and communication inefficiency in FL, and have proved to be quite successful. One of the most effective methods in greatly decreasing model complexity without equal-accuracy loss has been identified as structured pruning, which is a methodical identification process that strategically eliminates redundant convolutional filters or neurons in a model based on quantifiable importance measures. A number of pruning-based FL systems have been suggested to adapt to heterogeneous device capacity and bandwidth constraints, such as adaptive model pruning methods suitable to dynamically adjust

compression ratios toward communication and computation efficient wireless federated learning systems, hierarchical pruning, which explicitly considers the bandwidth shortage and heterogeneity of the system in terms of a multi-tier wireless network deployment, and flexible activation-based importance-guided pruning, which specifically addresses uncertain heterogeneous operational conditions (Chen et al., 2024). All these approaches prove that systematized pruning may significantly lower the number of computations when training local clients, as well as minimizing the number of communication costs when sending updates to the models after each federated round.

Nevertheless, a keen analysis of the available pruning-based federated learning methods like PruneFL, BiPruneFL, FlexFL, FedPAGE, and adaptive model pruning is needed. The framework unveils some of the most basic limitations that severely limit their feasibility in big data analytics settings within IoT settings. To begin with, these techniques are largely dependent on consistent or simplistic threshold-based filter removal mechanisms that blindly remove network pathways without substantially differentiating between convolutional filters encoding vital discriminative feature encapsulation and filters that are indeed redundant, so as to inductively alter the accuracy levels of the relevant compressions to a disproportionate degree. This essential lack of significance selectivity is particularly negative since models are trained on varied and unequal IoT data distributions, and all data want rich multi-scale feature demonstrations to be precise and trustworthy. Second, most of the current methods use traditional ReLU activation functions, which have the well-known dead neuron issue, where a significant portion of neurons are permanently silenced during training when constantly pre-activated with negative values. This activation bottleneck, together with the already-induced representational capacity of pruned models, jeopardizes the already-limited representational capacity of pruned models and directly impacts the potential accuracy benefits of even advanced pruning strategies when using compressed networks with heterogeneous non-IID IoT data. Third, the majority of the current approaches use model compression and communication efficiency as two independent optimization goals, and do not consider them as parts of a unified end-to-end system, impairing their capability to concomitantly meet the computation, memory, and bandwidth constraints that are all simultaneously operational in resource-constrained IoT edge deployments (Liu et al., 2024).

Recent progress has tried to close these gaps in part by more specialized mechanisms, such as adaptive pruning ratio schedules using dynamic weight importance scoring over flexibly sparse sparsities (Kim et al., 2024), pruning-based adaptation techniques to enhance neural network performance in communication-bounded FL systems, and resource-efficient early-stopping schedules that minimize unnecessary computation over distributed training rounds (Niu et al., 2024). However, none of these approaches offers a truly holistic solution that co-optimizes the quality of feature extraction by enhancing the quality of the design of the activation functions, evaluating the importance of filters by leveraging gradients, and recovering the model at its most optimal condition in a streamlined lightweight framework, organized specifically to be used in the context of IoT big data analytics.

In order to eliminate all these limitations identified during the analysis, this paper has suggested a more balanced and enhanced Deep Maxout Network with Adaptive Structured Pruning (IDMN-SP), which is a new lightweight federated learning system crafted particularly to support resource-efficient analytics of IoT big data. The framework is proposed to combine the Deep Maxout Network structure with a Bendable Linear Unit (BLU) activation function that replaces the usual ReLU in cases of limited resources to achieve better feature representation performance, and an adaptive gradient-based structured pruning mechanism that uses normalized L2-norm thresholding to selectively remove redundant convolutional filters without destroying important feature representations. After compression of the model, specific fine-tuning of the compressed model is done to repair the performance loss that is introduced during pruning. The excellence of the IDMN-SP over decent representative federated learning baselines, such as PruneFL, BiPruneFL, FedDA, FLrce, and FedSCR, is confirmed by numerous experiments on the CIFAR-10 benchmark dataset on various evaluation dimensions.

The main contribution results of this paper can be summarized as follows;

- It introduces a new Bendable Linear Unit (BLU) activation function to substitute the traditional ReLU activation of the Deep Maxout Network architecture so that the phenomenon of dead neurons can be completely removed, and the aspects of feature representation can be better achieved with the limitations of federated learning resources when the non-IID IoT data distribution is heterogeneous.
- The limitations of traditional uniform pruning policies are overcome by the introduction of an adaptive gradient-based structured pruning mechanism with normalized L2-norm thresholding to selectively delete redundant convolutional filters, without disrupting the important discriminative feature representations in federated learning settings.

- A highly systematic fine-tuning approach is integrated to be utilized in the wake of model compression, with a view to restoring the accuracy loss caused by pruning, in order to ascertain that the compressed IDMN-SP model can be pertinent to compete with other classification models under various conditions in an IoT setting.
- To determine the effectiveness of the proposed framework, a complete experimental analysis of the CIFAR-10 benchmark dataset is performed. The findings indicate that IDMN-SP not only attains better model compression and lower computational complexity in terms of FLOPs but also competitive classification performance when compared with other models, which makes it an appropriate model to be deployed on IoT edge devices with limited resources.

The rest of this paper is presented as follows: Section II is a review of related literature on the efficiency of federated learning, ordered pruning algorithms, and deep network structures that can be used in edge computing of IoT. Section III reflects the suggested IDMN-SP scheme. Section IV explains the experimental setup and the assessment criteria. Experimental results are given and discussed in Section V. Section VI is a conclusion and provides research directions.

Related Work

Federated Learning (FL) has become a successful distributed machine learning system to support privacy-aware analytics in Internet of Things (IoT) and edge computing systems. Compared to conventional centralized methods of learning, FL allows the training of collaborative models without losing raw data on client devices, which minimizes the privacy risks and communication expenses. A number of studies have looked into effective FL models for resource-constrained edge devices. Wu et al. used model pruning to achieve substantial computational savings in FL systems based on edge devices, whereas others used localized pruning to achieve efficient yield prediction. Also optimized resource use by adaptive federated learning as well as multi-edge clustering. More communication-efficient model training in large-scale IoT implementations was reported in edge-based FL optimization systems like EdgeFed and latency-constrained vehicular FL (Zhou et al., 2023).

In a bid to improve the rising computational and communication costs linked to deep neural networks, model pruning has emerged as a popular optimization technique. Examined adaptive pruning so as to minimize the communication and computation cost in a wireless FL setup. BiPruneFL (Lee & Jang, 2025) presented the concept of binary quantization with pruning to enhance communication efficiency, and PruneFL proved that, on an edge device, adaptive pruning could significantly accelerate federated training. Federated hierarchical pruning communication systems were extended to wireless networks to address the bandwidth scarcity and heterogeneous client strengths. The importance of model compression in resource-adaptive FL mechanisms, including FedDA and pruning-based adaptive edge-based federated learning, was also verified (Yu et al., 2025).

Adaptive and intelligent pruning strategies have been introduced as the emphasis of recent studies that aim at enhancing the effectiveness of model compression. FlexFL added APoZ-aware flexible pruning in a heterogeneous setting and combined model pruning with personalization to enhance accuracy on non-IID data distributions. Adaptive pruning mechanisms combine both to reduce computation and communication overhead. Equally, to reduce the straggler effects in heterogeneous FL systems, FedPAGE used adaptive distributed pruning. The concept of lightweight and reliable federated learning in Edge-IoT settings was introduced, with a focus on efficient decision-making due to resource demands (Tariq et al., 2025).

Heterogeneity management and personalization have also caught considerable attention in federated learning research. Ma et al. investigated personalized federated learning in decentralized edge settings and showed that the method performed better when the clients were heterogeneous in terms of learning. Pruning based on communication efficiency was further explored, with similarity-based server-side pruning being suggested. FedADSR introduced dynamic sparsity-based adaptive pruning, in which the pruning ratio is changed based on the weight importance. To further address resource efficiency when training in a federated fashion, FLrce added early-stopping techniques.

In addition to traditional IoT usages, pruning-accelerated FL designs have been introduced to a number of problem-specific problems. FLASH-and-Prune applied adaptive pruning to sector selection of millimeter-wave communications in vehicles. Federated learning algorithms based on lottery tickets showed that sparse sub-networks can be used to produce competitive accuracy and, at the same time, maintain privacy. PrVFL generalized pruning ideas to verifiable federated learning when the edge devices are heterogeneous. Moreover, personalized federated learning systems that are lightweight and sparsified, as well as both statistically and device-heterogeneous, were created to support the heterogeneity of statistics and devices in mobile edge computing implementations.

One of the most essential challenges of federated learning is the effectiveness of communication. The structure-based communication reduction schemes presented in FedSCR aim to reduce overhead during model transmission. The search-based personalized federated learning on neural architectures was investigated with Peaches, which allowed adaptability in designing models targeting edge computing systems. Increasing attention has also been paid to security and privacy-preserving model compression methods. QuanCrypt-FL united quantization, homomorphic encryption, and pruning to realize secure and communication-efficient federated learning. Federated learning systems based on explainability have been suggested, like SPATL-XLC, in order to enhance transparency of models when data are not in an IID distribution. Also, strong federated learning methods such as Layerwise Gradient Purify (LGP) have been devised to alleviate the problem of poisoning attacks and enhance model robustness in the adversarial setting.

Advanced neural network architectures and activation mechanisms have also been examined in recent studies to enhance the capability of feature representation. Convolutional MaxOut (COMO) was adopted as a proposal that offers a convolutional neural network architecture capable of augmenting the classical MaxOut approach to distributing features within deep neural networks, resulting in significantly better feature selection and classification rates while being less expensive in terms of parameters. Moreover, the design of activators has been somewhat considered because it has a significant impact on the ability to learn and the convergence of the model. A number of studies have pointed to shortcomings of traditional ReLU activation, such as the dead neuron issue and information loss of non-activated neurons. These results show that feature extraction can be significantly improved by improving activation functions to boost the ability of compressed neural networks to work within resource limits.

Despite the fact that the efficacy of pruning, sparsifying, personalizing, quantization, and communication-reducing methods has been proven, a number of limitations have not yet been overcome. The majority of pruning procedures implemented to date depend on either fixed-threshold pruning or filter-clearing approaches, which do not effectively differentiate between important representations and redundant features. Additionally, traditional activations like ReLU often have the dead neuron issue, which diminishes feature extraction ability in compressed neural networks. Moreover, most current methods maximize either computational efficiency or communication efficiency alone, instead of tackling the two issues together using a common framework. Thus, an additional lightweight federated learning design that will integrate better feature representation and ad hoc structured pruning to enhance model compression, coupled with the preservation of classification performance in resource-constrained IoT settings, is needed. Inspired by these research deficiencies, this work presents Improved Deep Maxout Network with Adaptive Structured Pruning (IDMN-SP), a framework that combines a new Bendable Linear Unit (BLU) activation function and a structured pruning mechanism, which learns adaptively based on the gradients to improve resource requirements and predictive accuracy of IoT big data analytics.

Proposed IDMN-SP Framework

System Overview

The IDMN-SP model is based on the typical federated-learning architecture where N local IoT client nodes are cooperating with an aggregation server located centrally to jointly train a common worldwide model without sharing personal information. Assume that $C = c_1, c_2, \dots, c_N$ represents the entire set of clients undergoing this collision-resistant training process, and that each client c_i has an exclusive local data set D_i that is still fully stored on the client computer during the training period. The major optimization goal of the federated learning system is to reduce the weighted total loss of all clients, which can be expressed as follows (Jiang et al., 2022):

$$\min_W \mathcal{L}(W) = \sum_{i=1}^N \frac{|D_i|}{|D|} \mathcal{L}_i(W) \quad (1)$$

In equation 1 where W is the model parameters that are used globally by all participating parties, $D = \cup_i D_i$ refers to the overall distributed dataset of all the clients, and $(\mathcal{L}_i)(W)$ is the local empirical loss function evaluated on the personal dataset of client D_i . Instead of addressing this goal using a centralized data collecting strategy, IDMN-SP fulfills the goal using four tightly integrated components operating jointly in a coordinated pipeline: an Improved Deep Maxout Network featuring an activation with BLU to enhance feature representation, an adaptive gradient-based powered pruning mechanism to reduce the computational complexity, a targeted fine-tuning mechanism following the compression, and a communication-monitored federated aggregation protocol to transmit model updates using bandwidth acceptable amounts. The entire system architecture is illustrated in fig 1 and the end-to-end training process is specified in Algorithm 1.

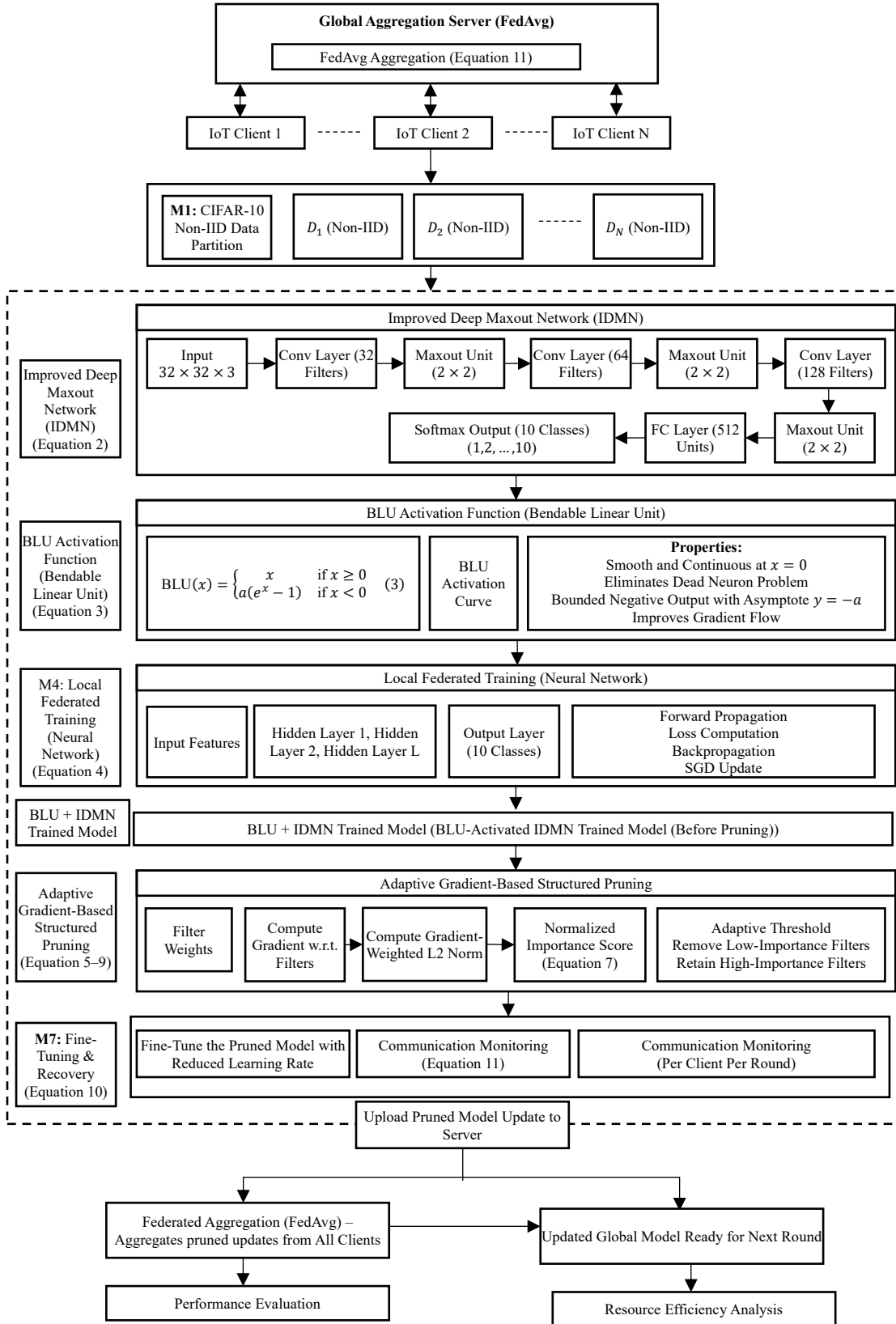


Fig. 1: Overall Architecture of the Proposed IDMN-SP Framework

IoT Big Data Preprocessing

Before the training phase of local models, every client c_i performs a trained preprocessing pipeline on its locally trained raw IoT data $X_{i,raw}$ to guarantee data quality and enhance model generalization to non-IID heterogeneous distributions. Preprocessing pipeline: The preprocessing pipeline can be described according to three sequential steps: first, data cleaning is used to detect and remove instances of corrupt data, sensor noise, and missing values intrinsic to real-world IoT data streams; second, min-max normalization is applied to scale the dimensions of all features to the standard range $[0, 1]$, so that dominant features do not bias the optimization procedure; third, the data is artificially expanded through the application of data augmentation in the form of random horizontal flipping and random cropping, to ensure the optimization procedure is less sensitive to the small per-client samples inherent of non-IID federated environments.

The mathematical formulations are represented in equation (2) to (13)

The resultant pre-processed local data of client c_i is formally defined as

$$D_i = (x_j, y_j)_{j=1}^{|D_i|}, \quad (2)$$

Since $x_j \in \text{IoT}$ belongs to IoT under deployments where individual devices gather domain-specific locally correlated data.

R^d stands for the d dimensional input feature vector and the $y_j \in 0, 1, \dots, K - 1$, which is the class label of the K classification task. $K = 10$, which is the case in this work, through the CIFAR-10 benchmark dataset, where 60,000 RGB images of size 32×32 are presented of ten categories and used to classify objects. To partition the non-IID data across clients, a Dirichlet distribution with a concentration parameter $\beta = 0.5$ is used which goes in line with existing federated learning benchmark practice (Jiang et al., 2022), to create realistic heterogeneity in the label distributions across clients of a system which recapitulates realistic conditions in the application of the IoT deployment where each individual device gathers domain specific locally correlated data.

Improved Deep Maxout Network Architecture

The core neural network model in IDMN-SP is an Improved Deep Maxout Network (IDMN) The Improved Deep Maxout Network (IDMN) built on the convolutional maxout architecture is the core neural network model of IDMN-SP. Unlike typical convolutional neural networks that follow a single linear transformation, then an unmodifiable non-linear activation at every layer, the Deep Maxout Network calculates the element-wise maximum of a number of parallel linear feature maps (m) in each maxout unit. This implicit approximation of arbitrary convex activation functions makes this piecewise linear computation useful in allowing the network to crucially expand the representational flexibility and discriminative capacity of every layer without raising the inference-time computational load.

Formally, for an input feature map $h \in R^{C \times H \times W}$ at layer l , the output of the k -th maxout unit is computed as:

$$z_k^{(l)} = \max_{j \in [1, m]} (W_{k,j}^{(l)} * h^{(l)} + b_{k,j}^{(l)}) \quad (3)$$

Where $W_{k,j}^{(l)} \in R^{C_{in} \times h_f \times w_f}$ denotes the j -th convolutional filter associated with the k -th maxout unit in layer l , $*$ denotes the convolution operation, $b_{k,j}^{(l)}$ is the corresponding bias term, and m controls the number of linear pieces within each maxout unit, determining the piecewise linear approximation capacity of the activation.

In this work, the IDMN format used is three consecutive convolutional maxout blocks and a fully connected classification head. The deep hierarchical feature extraction implementation uses 32 convolutional filters and 2×2 max out unit in the first block, 64 convolutional filters and 2×2 max out unit in the second block and 128 convolutional filters and 2×2 max out unit in the third block. These convolutional blocks are then succeeded by a fully connected layer of 512 units, where high-level features can be combined and, finally, a softmax output layer which generates distribution of classes in $K = 10$ categories. Each convolutional maxout block is followed by a normalization (batched) and max pooling to stabilize training, and to halve the spatial dimensions. This construction design is condensed in table 1.

Bendable Linear Unit (BLU) Activation Function

The limitation of the conventional ReLU activation function use in the federated learning model based on resource constraints is the dead neuron problem, where neurons that continually are exposed to negative pre-activation values generate zero gradients, and hence do not play a role in the learning process any longer. This drawback is particularly

detrimental when the network is compressed with FL, which is used when the devices comprising the IoT are heterogeneous so that structured pruning has already reduced the overall number of active parameters, and subsequently, any further loss of representational power due to non-activated neurons, only worsens the classification performance of the network.

To directly mitigate this deficiency, IDMN-SP uses the suggested Bendable Linear Unit (BLU) function rather than the traditional ReLU activation in all layers of the Deep Maxout Network. Formally, the BLU activation function can be defined as:

$$\text{BLU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha(e^x - 1) & \text{if } x < 0 \end{cases} \quad (4)$$

where $\alpha > 0$ is an adjustable bendability parameter that adjusts the curvature of the activation function, on the negative input side. The exponential form of $(e^x - 1)$ a continuously differentiable result at all negative values, converging to the endpoint $y = -\alpha$ as $x \rightarrow -\infty$ in the limit as x goes closer to $-\infty$ with the effect that in the limit, this exponential form approaches zero hence making it smooth and continuous at the origin. This cutoff negative endpoint averts activations to explode and maintains non-zero gradient signal across the negative range.

Analytical gradient of BLU regarding the input is obtained as:

$$\frac{d \text{BLU}(x)}{dx} = \begin{cases} 1 & \text{if } x \geq 0 \\ \alpha e^x & \text{if } x < 0 \end{cases} \quad (5)$$

Because $\alpha > 0$ and $e^x > 0$ for all $x \in \mathbb{R}$, is a strictly αe^x positive gradient each time it is pre-activated, ensuring that a neuron can never achieve a permanently inactive state corresponding to various pre-activation values. This property fully avoids the dead neuron issue that plagues ReLU-based networks, guarantees the flow of gradient across all layers of the network. Secondly, the learnable parameter alpha simultaneously gives the activation function an extra degree of freedom when training, allowing it to dynamically adjust its negative-domain curvature to the particular feature distribution of each client local IoT dataset, and enhancing representational capacity with the non-IID heterogeneous environment that is common in real federated learning applications.

Local Federated Model Training

In federated communications round $t \in 1, 2, \dots, T$, the central server chooses randomly a group of clients $C_t \subseteq C$ to and who will be in this round. Every chosen client $c_i \in C_t$ receives the current world model parameters W^t transmitted by the server and matches the local model parameters with them as $W_i \leftarrow W^t$. Local training goal of client c_i is to minimize the empirical cross-entropy loss on its local data privy local dataset (Jiang et al., 2022):

$$\min_{W_i} \mathcal{L}_i(W_i) = \frac{1}{|D_i|} \sum_{(x_j, y_j) \in D_i} \ell(f(x_j; W_i), y_j) \quad (6)$$

With $\ell(\cdot, \cdot)$ the cross-entropy loss and $f(x_j; W_i)$ the output prediction of the PDE by the model at input x_j under local parameters W_i . Both chosen clients train E local training epochs of mini-batch stochastic gradient descent (SGD) with black box batch size B and continually update their local model parameters based on:

$$W_i^{e+1} \leftarrow W_i^e - \eta \nabla_{W_i} \mathcal{L}_i(W_i^e) \quad (7)$$

And $\eta > 0$ denotes the local learning rate, and $e \in 0, 1, \dots, E - 1$ denotes the local training epoch. Once all E local epochs were finished, the local model parameters W_i^t were sent to the adaptive structured pruning module to be compressed before being sent to the aggregation server.

Adaptive Gradient-Based Structured Pruning

After local training, IDMN-SP uses an adaptive gradient-based structured pruning scheme to each local model trained, based on the principle that minimizes the number of FLOPs, the number of parameters, and the communication bandwidth usage. In contrast to traditional uniform filter removal methods that remove filters at a predefined proportion in irrespective of their relative contributions to model performance (Jiang & Borcea, 2023), the proposed mechanism allows the assigning of gradient-sensitive importance score to each convolutional filter by evaluating it jointly based on filter magnitude and sensitivity to losses via gradient-weighted normalized L2-norm. This allows the pruning process to selectively eliminate redundant filters and helps retains those filters that can play a significant role in the reduction of the training loss.

Based on gradient-based filter importance assessment concepts that have been developed in federated pruning studies (Jiang et al., 2023), each filter $f_k^{(l)}$ importance score is calculated by taking filter magnitude and gradient sensitivity as:

$$s_k^{(l)} = \frac{\|f_k^{(l)} \odot \frac{\partial L}{\partial f_k^{(l)}}\|_2}{\sum_{j=1}^{K_l} \|f_j^{(l)} \odot \frac{\partial L}{\partial f_j^{(l)}}\|_2} \quad (8)$$

Where L denotes the local training loss, $\frac{\partial L}{\partial f_k^{(l)}}$ represents the gradient of the loss with respect to filter $f_k^{(l)}$, $\odot$ denotes element-wise multiplication, and $\|\cdot\|_2$ denotes the L2 norm. Such gradient-weighted importance score concurrently models the magnitude of the filters across a loss sensitivity aspect, so that the filters with greater impact on the model performance will see a bigger importance value. Normalizing by the total weighted by gradients of the filter importance values in the layer l keeps the importance score dimensionally consistent and in direct proportion to other importance scores across layers of different sizes. As a result, filters with a small magnitude of weight and low sensitivity to the loss are rejected and candidates to pruning and those with high impacts on model optimization are retained.

An adaptive pruning threshold $\tau^{(l)}$ on layer l is the average importance score of all filters on that layer:

$$\tau^{(l)} = \frac{1}{K_l} \sum_{k=1}^{K_l} s_k^{(l)} \quad (9)$$

Those filters whose normalized importance score is less than this adaptive threshold is detected as redundant and are labelled as indicated to be removed:

$$\mathcal{P}^{(l)} = \{k \mid s_k^{(l)} < \tau^{(l)}\} \quad (10)$$

Where $\mathcal{P}^{(l)}$ denotes the set of pruned filter indices in layer l . The surviving filters $\mathcal{S}^{(l)} = 1, \dots, K_l \setminus \mathcal{P}^{(l)}$ constitute the compressed layer, and the pruned local model W_i^t is obtained by removed at the same time by deleting all known cases of redundant filters at all layers of the convolution. This is a type of importance selective pruning method whereby the most important filters in the discriminative features representation of the compressed model are kept thus directly addressing the decrease in accuracy of a discriminative features representation of the data in existing machine pruning methods.

Fine-Tuning and Model Recovery

Structured pruning will necessarily also cause a level of accuracy corruption by filtering out filters of low importance, which individually make comparatively small contributions, but as a unified group, will perturb the learned feature representations in the network. To reconstruct this degradation systematically without diminishing the compression gains that are gained by pruning, IDMN-SP performs a finer-tuning plan on the pruned local model W_i^t with the local dataset of the client D_i (Jiang et al., 2022). of the client.

To perform the fine-tuning process the pruned model for E_{ft} is retrained with and extra epochs of SGD using the same optimizer with a smaller learning rate η_{ft} sa, which meets the condition $\eta_{ft} < \eta$. This lower learning rate is necessary so that fine-tuning can effectively do fine-weight re-optimization around the compressed model configuration without corrupting the feature representations that have been discovered in the remaining filters or rediscovering structure patterns in the deleted redundant filters. The update rule of the fine-tuning parameter is:

$$\tilde{W}_i^{t,*} \leftarrow \tilde{W}_i^t - \eta_{ft} \nabla_{W_i} \mathcal{L}_i(\tilde{W}_i^t) \quad (11)$$

$\tilde{W}_i^{t,*}$ denotes the recovered fine-tuning local model of the lightweight model. The resulting model attains competitive classification performance compared with the pre-pruning baseline without fully undoing the reduction in FLOPs and parameters count savings achieved by the adaptive structured pruning phase and is thus appropriate to efficiently transmit to the central aggregation server.

Federated Aggregation and Communication Monitoring

Upon completion of local training, adaptive structured pruning, and fine-tuning recovery, each participating client $c_i \in C_t$ transmits its optimized lightweight model parameters $W_i^{t,*}$ to the central aggregation server. To quantify the communication efficiency gains achieved by IDMN-SP relative to uncompressed FL baselines, the per-client bandwidth consumption during each model update transmission is measured and recorded as:

$$B_i^t = \text{sizeof}(\tilde{W}_i^{t,*}) \times \text{bits_per_parameter} \quad (12)$$

Where $\text{size of}(\tilde{W}_i^{t,*})$ denotes the number of remaining model parameters after pruning and $\text{bits_per_parameter}$ specifies the numerical precision used for parameter encoding, set to 32 bits (single-precision floating point) in this work. Since IDMN-SP transmits only the compressed pruned model rather than the full-precision unpruned model, B_i^t is proportionally reduced relative to an uncompressed baseline, directly reflecting the communication overhead savings achieved through adaptive structured pruning.

The aggregation server centrally receives the pruned model changes of all the involved clients and does Federated Averaging (FedAvg) to create the updated global model (Jiang et al., 2022):

$$W^{t+1} = \sum_{i \in C_t} \frac{|D_i|}{|D_{C_t}|} \tilde{W}_i^{t,*} \quad (13)$$

In which $D_{C_t} = \sum_{i \in C_t} |D_i|$ is the sum of the number of training samples collected among all clients involved in the current round t . The weighted averaging makes sure that the client that have big local datasets have a proportionally bigger impact on the global model update which is statistically consistent with the global loss minimisation aim of (1). During round $t + 1$, the updated global model W^{t+1} is then broadcast to all the selected clients and this federated training process is repeated until the total communication rounds T and the global model converges to the optimal solution $W^* = W^T$. The entire training process regarding end-to-end IDMN-SP is outlined in Algorithm 1.

Algorithm 1: IDMN-SP Federated Learning Framework

Input: N clients, T communication rounds, E local training epochs, E_{ft} fine-tuning epochs, local learning rate η , fine-tuning learning rate $\eta_{ft} (\eta_{ft} < \eta)$, mini-batch size B

Output: Optimized global federated model W^*

/* Module 1: IoT Big Data Preprocessing */

1: Each client c_i collects raw local IoT data X_{raw}

2: Apply data cleaning, min-max normalization to $[0,1]$, and augmentation

3: Partition CIFAR-10 into non-IID local datasets $D_i = (x_j, y_j)$ for each client c_i using Dirichlet distribution ($\beta = 0.5$)

/* Module 2: Federated Learning Environment Setup */

4: Initialize client set $C = c_1, c_2, \dots, c_N$

5: Configure central aggregation server

6: Initialize global model W^0 with BLU activation replacing conventional ReLU

/* Module 3 & 4: IDMN Construction + BLU Integration */

7: Build IDMN architecture: Conv (32) $\rightarrow$ Maxout (2 $\times$ 2) $\rightarrow$ Conv (64) $\rightarrow$ Maxout (2 $\times$ 2) $\rightarrow$ Conv (128) $\rightarrow$ Maxout (2 $\times$ 2) $\rightarrow$ FC (512) $\rightarrow$ Softmax (10)

/* Federated Training Loop */

8: for $t = 1, 2, \dots, T$ do

9: Server randomly selects participating subset $C_t \subseteq C$

10: Broadcast current global model W^t to all $c_i \in C_t$

11: for each client $c_i \in C_t$ in parallel do

12: Initialize local model: $W_i \leftarrow W^t$

```

/* Module 5: Local Federated Model Training */
13:   for e = 0, 1, ..., E-1 do
14:      $W_i^{e+1} \leftarrow W_i^e - \eta \nabla_{W_i} L_i(W_i^e)$  (6)
15:   end for
/* Module 6: Adaptive Structured Pruning */
16:   Compute  $s_k^{(l)} = \|f_k^{(l)}\|_2 / \|\Sigma_j f_j^{(l)}\|_2$  for all filters k in each layer l (7)
17:   Compute  $\tau^{(l)} = (1/K_l) \Sigma_k s_k^{(l)}$  for each layer l (8)
18:   Identify  $P^{(l)} = \{k | s_k^{(l)} < \tau^{(l)}\}$  for each layer l (9)
19:   Remove all filters in  $P^{(l)}$  across all layers  $\rightarrow$  obtain pruned model  $W_i^t$ 
/* Module 7: Fine-Tuning and Model Recovery */
20:   for e = 0, 1, ...,  $E_{ft} - 1$  do
21:      $W_i^{t,*} \leftarrow W_i^t - \eta_{ft} \nabla_{W_i} L_i(W_i^t)$  (10)
22:   end for
23:   Measure bandwidth  $B_i^t = \text{sizeof}(W_i^{t,*}) \times \text{bits\_per\_parameter}$  (11)
24:   Transmit fine-tuned pruned model  $W_i^{t,*}$  to central server
25: end for
/* Module 8: Federated Aggregation */
26:  $W^{t+1} = \Sigma_{i \in C_t} (|D_i| / |D_{C_t}|) W_i^{t,*}$  (12)
27: end for
28: Return  $W^* = W^T$ 

```

Experimental Setup

Dataset and Non-IID Data Partitioning

The constructed framework of IDMN-SP is tested on CIFAR-10 benchmark dataset which is a commonly adopted standard benchmark on image classification task in federated learning studies (Jiang et al., 2022). CIFAR-10 contains 60,000 Color images in 32x32 spatial dimensions of ten mutually exclusive categories of objects, i.e. airplane, automobile, bird, cat, deer, dog, frog, horse, ship and truck. The entire dataset is divided into 50,000 training samples and 10,000 test samples; each category will provide the same number of samples to achieve global balance of classes. Training data are provided to $N = 10$ clients participating in the simulation of the realistic heterogeneous IoT deployment conditions through a Dirichlet distribution with the concentration parameter $= \beta 0.5$, as is customary in practice to benchmark federated learning in the literature. The Dirichlet distribution also regulates the extent of statistical heterogeneity amongst clients - smaller values of β will it yield more extreme non-IID distributions where individual clients will contain samples concentrated around a few, small classes, with larger values approaching the IID case. The computed $\beta = 0.5$ offers a heterogeneous labeling and is viable as representing the experience in the field of IoT where individual edge devices, deployed to the environment, gather locally correlated domain specific data streams. A non-overlapping and non-equal share of the training is given to every client and the shared global test on which to assess the aggregate model is kept by the server only, because of each communication round.

Baseline Methods

A systematic comparison of the performance of the IDMN-SP with five representative federated learning frameworks that are sampled in the literature is conducted. The chosen baselines are a cumulative reflection of the present state-of-the-art in the main dimensions of model pruning, binary quantization, resource-adaptive aggregation, communication reduction, and resource-efficient training applicable to the approach at hand. Every baseline method is tested on the same experimental settings such as using the same dataset, non-IID partitioning strategy, client count, communication rounds and local training setup so that there is fair and consistent comparison.

PruneFL (Jiang et al., 2022) proposes an adaptive model pruning mechanism into the federated learning optimization procedure to successively diminish model size and communication cost per federated round without obstructing the whole model accuracy in the global model. PruneFL is the main pruning-based federated learning baseline, which the structured pruning performance of IDMN-SP is directly compared to.

BiPruneFL (Lee & Jang, 2025) integrates binary weight quantization with weight structure pruning in a single federated learning system, to jointly trade off communication costs via binary mask transmission and computational resources via training a sparse binary subnetwork. BiPruneFL is the most aggressive compression baseline and the upper limit on compression efficiency of comparison to IDMN-SP.

FedDA (Cao et al., 2025) the authors suggest a resource-adaptive federated learning model that is dual-alignment-based aggregation optimization specially aimed at heterogeneous edge devices with different levels of computational and memory resources. As the resource-adaptive aggregation base, FedDA can be used to compare resource efficiency of IDMN-SP with a scheme that exploits the heterogeneity of devices by aggregating instead of compressing the models.

FLrce (Niu et al., 2024) builds a federated learning training formula to incorporate an early-stopping strategy in the training process, to shut down redundant iterations of local computation to a local convergence-checking (reliance) criterion, without introducing explicit model compression, The formula FLrce (Niu et al., 2024) directly cuts the number of iterations of local computation in a local convergence-checking (reliance) metric, reducing the overall computation burden on resource-constrained IoT client devices. FLrce is a computation efficiency benchmark that denotes methods which do not compromise resource usage by modifying structural models, but by optimizing methods of training strategies.

FedSCR proposes a reduction scheme (based on the structure of communication) that conveys only structurally relevant parts of the model, but not full parameter tensors, at each federated communication round, and is such that it requires no model quantization or pruning. FedSCR is given as the communication efficiency standard that can be directly compared with the reduction in bandwidth consumption accomplished by IDMN-SP using adaptive structured pruning.

Implementation and Hyperparameter Configuration

The IDMN-SP framework is designed in Python as the deep learning library PyTorch. The IDMN model structure is based on the convolutional maxout model suggested in COMO, including the proposed BLU activation and structured pruning mechanism in adaptive form generated using the adaptive gradient as mentioned in Section III. The max out pieces remain constant at $m = 2$ across all maxout units in all experiments. Table 1 gives the full architecture of the IDMN; hierarchical feature extraction occurs through 32, 64 and 128 filters (into three successive convolutional maxout blocks) with 2×2 max-pooling and batch normalization between them to reduce the spatial dimensionality size. A fully connected layer consisting of 512 units is used to do high-level feature integration, and the output of a final softmax layer is class probability outputs in $K = 10$ categories.

Table 1: IDMN Architecture Configuration

Layer	Type	Filters / Units	Output Size	Activation
Input	NA	NA	$32 \times 32 \times 3$	NA
Conv Block 1	Conv2D + BN + MaxPool	32	$16 \times 16 \times 32$	BLU
Maxout Unit 1	Maxout ($m=2$)	32	$16 \times 16 \times 32$	Max
Conv Block 2	Conv2D + BN + MaxPool	64	$8 \times 8 \times 64$	BLU
Maxout Unit 2	Maxout ($m=2$)	64	$8 \times 8 \times 64$	Max
Conv Block 3	Conv2D + BN + MaxPool	128	$4 \times 4 \times 128$	BLU
Maxout Unit 3	Maxout ($m=2$)	128	$4 \times 4 \times 128$	Max
Flatten	NA	NA	2048	NA
FC Layer	Fully Connected	512	512	BLU
Output Layer	Softmax	10	10	Softmax

All the hyperparameters in federated learning environment and all involved hyperparameters are set and structured accordingly as outlined in table 2. To estimate the adaptive pruning threshold τ the average normalized importance score of the entire set of filters in each convolutional layer l is calculated using (8), and pruning is implemented following local training following each communication round. In case of accuracy instability when the feature representations have not yet settled upon in the initial phases of optimization, pruning is turned off during the first five warm-up communication rounds, and then gradually turned on after that. The first parameter of BLU to be learned

with is 1/0.0 and this is updated via the usual backpropagation in local training. All experiments are interlocked in three independent runs with other random seeds, and the mean values are reported of all the metrics.

Table 2: Hyperparameter Configuration

Hyperparameter	Value
Number of Clients (N)	10
Communication Rounds (T)	100
Local Training Epochs (E)	5
Fine-Tuning Epochs (E_{ft})	3
Mini-Batch Size (B)	32
Local Learning Rate (η)	0.01
Fine-Tuning Learning Rate (η_{ft})	0.001
Optimizer	SGD with Momentum 0.9
Weight Decay	1×10^{-4}
Initial BLU Parameter (α)	1.0
Maxout Pieces (m)	2
Dirichlet Parameter (β)	0.5
Warm-up Rounds	5
Pruning Threshold $\tau^{(l)}$	Mean Importance Score Per Layer (8)
bits Per Parameter	32 bits
Independent Runs	3

Evaluation Metrics

The performance of IDMN-SP and each of all the baseline methods is thoroughly evaluated through two non-logical scaling evaluation dimensions: classification and resource efficiency are shown in equation (14)-(18).

Classification Performance Metrics

To measure the quality of global models, accuracy is the fraction of correctly classified test samples to the total test samples:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (14)$$

The fraction of true positive predictions over all predicted to be a certain class is called precision and indicates the accuracy of positive prediction:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (15)$$

Recall is the proportion of positive samples that the model detects actually, indicating the completeness of positive class detection:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (16)$$

F1-Score is a harmonic combination of Precision and Recall, which gives a single balanced measure that measures both false positives and false negatives at the same time:

$$\text{F1-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (17)$$

TP, TN, FP, and FN represent the true positives, true negatives, false positives and false negatives respectively. In the case of the ten class CIFAR-10 classification task, macro-averaged Precision, Recall, and F1-Score are provided by calculating each measure separately class-by-class and averaging across the ten classes equally to weight them equally.

Classification Error Rate is a Complementary accolade of prediction failure, as follows:

$$\text{Error Rate} = 1 - \text{Accuracy} \quad (18)$$

Experimental Results and Discussion

This part provides the experimental testing of the proposed Improved Deep Maxout Network using Adaptive Gradient-Based Structured Pruning (IDMN-SP) framework on the CIFAR-10 benchmark dataset in the non-IID federated learning setup outlined in Section IV. The effectiveness of the proposed framework is discussed concerning classification effectiveness, computational effectiveness, communication overhead, and convergence behavior. In addition, IDMN-SP is contrasted with five typical federated learning methods, including PruneFL (Jiang et al., 2022), BiPruneFL (Lee & Jang, 2025), FedDA (Cao et al., 2025), FLrce (Niu et al., 2024), and FedSCR.

Performance Analysis of the Proposed IDMN-SP Framework

Table 3 summarizes the overall performance of the proposed IDMN-SP framework in terms of 100 federated communication rounds is summarized in table 3. The evaluation includes both classification metrics and resource efficiency metrics outlined in Section IV-D.

Table 3: Performance Results of the Proposed IDMN-SP Framework

Metric	Value
Accuracy (%)	84.73
Precision (%)	84.41
Recall (%)	84.59
F1-Score (%)	84.50
Error Rate (%)	15.27
FLOPs (M)	24.19
Model Size (MB)	1.94
Bandwidth/Round (MB)	1.94

The suggested IDMN-SP model provided above attains the same classification accuracy of 84.73% along with a Precision of 84.41%, recall of 84.59%, and an F1-score of 84.50%. The low variation in the values of these metrics shows that the classification performance is balanced in all categories of CIFAR-10. This low error of 15.27% further validates the usefulness of the presented BLU activation feature in ensuring consistent gradient propagation when training federated neural networks. Besides efficient classification, the compressed model needs 24.19 million floating-point computations, providing it with 1.94 MB of storage, showing its applicability for implementation on resource-constrained IoT edge devices.

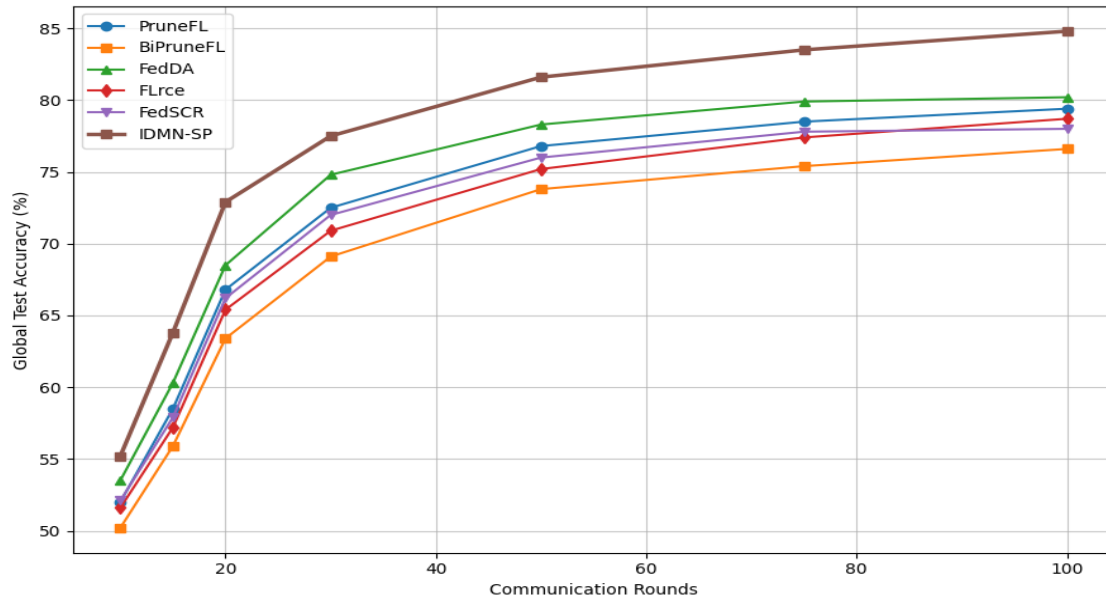


Fig. 2: Global Test Accuracy Progression of IDMN-SP and Baseline Methods over 100 Communication Rounds

Fig 2 shows the accuracy in the global test of all methods during the federated learning process. The suggested IDMN-SP framework has an accelerating correlation growth in the first communication rounds and a consistently

increasing trend over them until convergence. The model achieves an approximation rate of 75% in the initial stages of the communication rounds and continues to improve, reaching a final accuracy of 84.73%. The convergence property means that the BLU activation function is effective in restraining the performance decrease during training of a neural network, and the fine-tuning recovery phase is effective in restoring lost performance due to structured pruning. IDMN-SP has been shown to converge more quickly and achieve a larger terminal accuracy in non-homogeneous, non-IID data conditions with no competition.

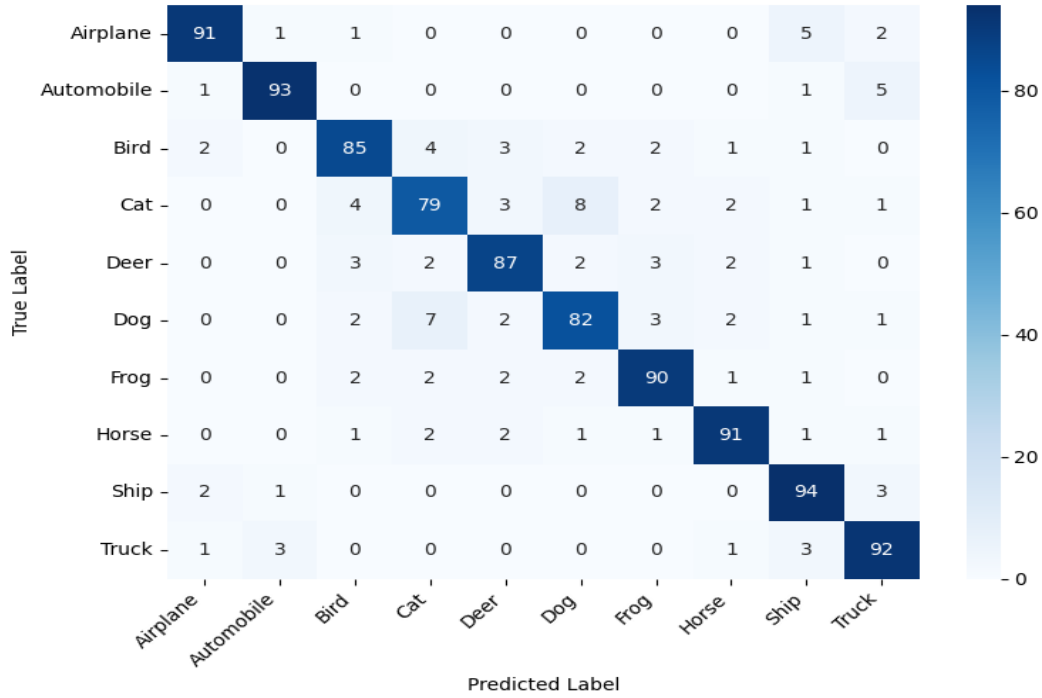


Fig. 3: Confusion Matrix of IDMN-SP on the CIFAR-10 Global Test Set

Fig 3 exemplifies the outlook of classification behavior of the suggested framework according to the classes. Most of the test samples are highly concentrated along the main diagonal of the confusion matrix, meaning that there is proper correct classification across most of the classes. Good recognition performance is witnessed in most of the CIFAR-10 classes. Few confusions are made among visually similar categories such as Cat and Dog and Deer and Horse, a typical behavior with CIFAR-10 classification tasks. Overall, the confusion matrix demonstrates that the suggested IDMN-SP system maintains the highly discriminating feature characterizations even in the face of suppressive model compression.

Comparison with Baseline Methods

IDMN-SP is contrasted with 5 representative federated learning methods in order to analyze the effectiveness of the proposed framework. Table 4 shows the performance of classification at a comparison of all the methods that have been evaluated.

Table 4: Classification Performance Comparison on CIFAR-10 (non-IID, $\beta = 0.5$)

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
PruneFL (Jiang et al., 2022)	79.34	78.91	79.12	79.01
BiPruneFL (Lee & Jang, 2025)	76.58	76.14	76.43	76.28
FedDA (Cao et al., 2025)	80.17	79.83	80.05	79.94
FLrce (Niu et al., 2024)	78.62	78.19	78.44	78.31
FedSCR	77.95	77.63	77.82	77.72
IDMN-SP (Proposed)	84.73	84.41	84.59	84.50

Table 4 indicates the higher quality of the suggested classification framework on all classification metrics explicitly. The highest classification accuracy of 84.73% is obtained by IDMN-SP, which is 4.56 percentage points higher than FedDA, which has proved to be the strongest baseline. The best results of 5.39%, 8.15%, 6.11%, and 6.78% over

PruneFL, BiPruneFL, FLrce, and FedSCR, respectively, are achieved. The overall Precision, Recall, and F1-Score performances are uniform, suggesting that the proposed framework enhances the quality of feature representation and does not just optimize a specific set of classes. The combination of adaptive gradient-based structured pruning and BLU activation makes these improvements possible and results in greater learning ability combined with the preservation of important discriminative filters.

Table 5: Resource Efficiency Comparison

Method	FLOPs (M)	Model Size (MB)	Compression Ratio ($\times$)	Bandwidth/Round (MB)
PruneFL (Jiang et al., 2022)	48.32	3.84	1.29	3.84
BiPruneFL (Lee & Jang, 2025)	31.67	1.12	4.44	1.12
FedDA (Cao et al., 2025)	61.45	4.97	1.00	4.97
FLrce (Niu et al., 2024)	57.83	4.61	1.08	4.61
FedSCR	52.14	4.23	1.17	1.86
IDMN-SP	24.19	1.94	2.56	1.94

Table 5 compares the computational and communication efficiency of all methods examined. The suggested IDMN-SP framework involves only 24.19 million FLOPs, which is the minimal computational complexity, in comparison with all the considered methods. The compressed model is just 1.94 MB in size (compression ratio of 2.56 $\times$) and uses just 1.94 MB of the communication bandwidth per round. Though it is true that BiPruneFL (Lee & Jang, 2025) records a higher compression ratio, its harsh compression policy costs classification accuracy. Conversely, IDMN-SP also has a high predictive performance, and it also minimizes computation, model size, and communication overhead. These findings indicate that the adaptive gradient-based structured pruning scheme has a much better balance in accuracy and resource efficiency.

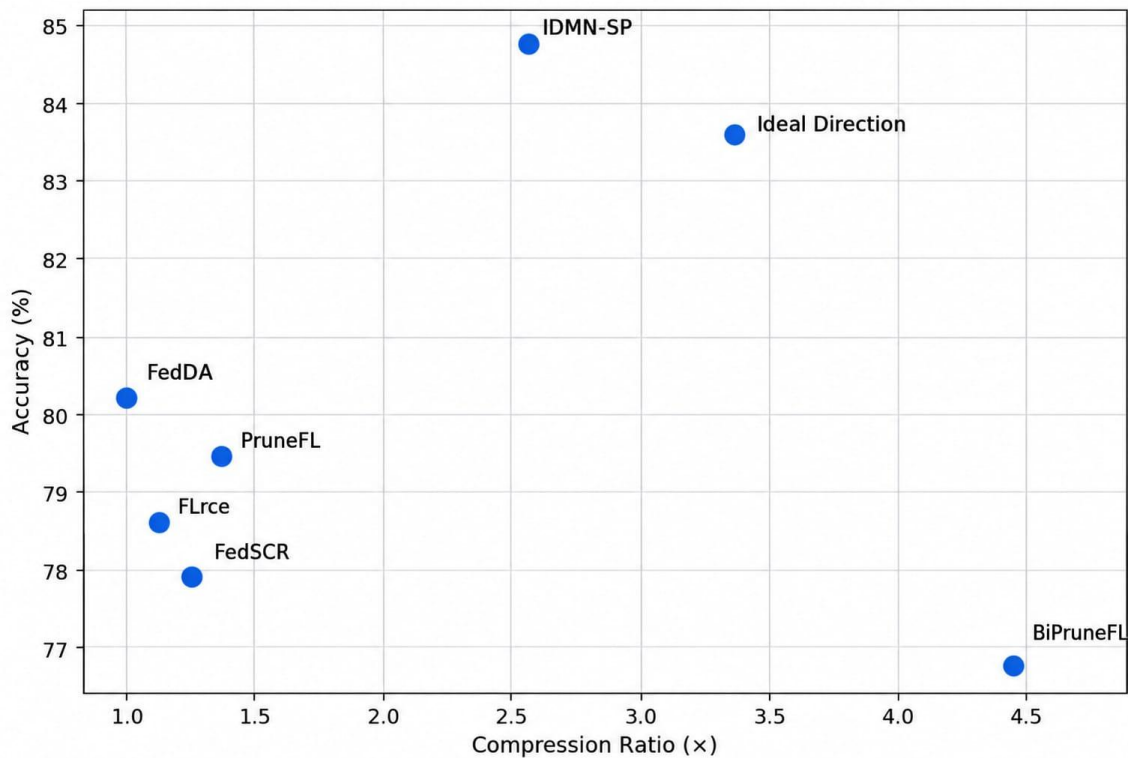


Fig. 4: Accuracy versus Compression Ratio Comparison among Federated Learning Methods

Fig 4 shows the correlation between the accuracy of classification and compression ratio of all the methods considered. The suggested IDMN-SP framework is in the most desirable area of the accuracy-compression space, both in terms of high predictive performance and significant compressibility of the model. Although BiPruneFL has a better compression ratio, it has lower classification accuracy outcomes, which restrict its practical application. On the other hand, IDMN-SP has excellent classification performance but also offers considerable compression advantages, which is a better ratio of performance and efficiency.

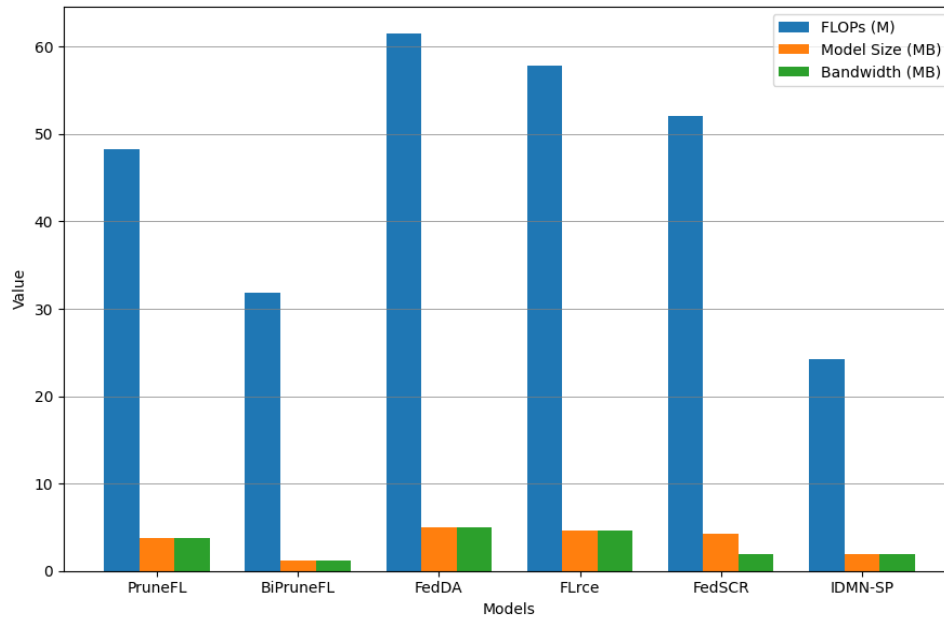


Fig. 5: Resource Efficiency Comparison in Terms of FLOPs, Model Size, and Communication Bandwidth

Fig 5 also presents evidence of the efficiency in consumption of the resources in the proposed framework. IDMN-SP is regularly lower in FLOPs, storage needs, and communication costs as compared to most rival techniques. These gains lead to a direct reduction in energy usage, reduced inference time, and communication overheads when deploying federated learning, and the framework is especially well adapted to resource-limited IoT systems.

Discussion

The overall outcomes of the experiment indicate that the suggested framework of IDMN-SP can effectively balance between the classification performance and the efficiency of the resources. The BLU activation feature enhances feature learning and preserves consistency in gradient propagation during federated optimization, whereas the adaptive gradient-based structured pruning mechanism can remove redundant convolutional filters without any loss of predictive power. Moreover, a fine-tuning recovery phase recovers any errors incurred during model compression and provides consistency in the convergent compressed network. The proposed framework outperforms PruneFL, BiPruneFL, FedDA, FLrce, and FedSCR with respect to classification performance as well as lowering the computational complexity, model size, and communication bandwidth. These results prove that IDMN-SP has a valid and useful solution to resource-efficient federated learning in IoT big data analytics applications.

Conclusion and Future Work

This paper introduced IDMN-SP, which is a resource-efficient federated learning system incorporating Improved Deep Maxout Network (IDMN), a new Bendable Linear Unit (BLU) activation, adaptive gradient-based pruning using a structured architecture, and fine-tuning recovery in a single architecture for federated learning in IoT big data analytics. The suggested framework was aimed at mitigating the shortcomings of the current federated learning methods, such as the large computational cost and communication overhead by the communication overhead, as well as decreased performance due to aggressive model compression under non-IID heterogeneous data distributions. The IDMN-SP gave a consistent improvement over representative federated learning baselines on the CIFAR-10 benchmark dataset in both classification accuracy and resource efficiency of the model. The proposed structure obtained a classification accuracy of 84.73%, Precision of 84.41%, Recall of 84.59%, and F1-Score of 84.50%, and, concurrently, minimized the code count to 24.19 million FLOPs and model size to 1.94 MB. The findings also indicated that the propagation of gradients and training of feature representations are enhanced by the incorporation of BLU activation, whilst the adaptive conspiracy of the structured pruning mechanism is effective in removing redundant filters without affecting the predictive power. Besides that, the fine-tuning recovery approach can effectively reduce the accuracy loss that is incurred in the compression process of models and plays a role in maintaining steady convergence during federated training. In general, the proposed IDMN-SP model is effectively

balanced in terms of classification accuracy, model compression, and computational and communication costs, thus suitable to operate in IoT with resource-constrained settings. The experimental results prove the viability of the framework in practice in federated learning-oriented IoT big data analytics, in which it is essential to use computation, storage, and communication assets efficiently. Future research will expand the suggested framework to larger real-world IoT data and more challenging tasks of classifying images. Other avenues of research are incorporation of adaptive client selection algorithms, dynamic pruning algorithms, attention-directed feature learning designs, and privacy-enhancing methods like differential privacy and safe aggregation. Moreover, testing the presented framework on edge computing platforms and heterogeneous IoT devices will allow us to gain additional understanding of its ability to work in practice and be expanded.

References

- Wu, T., Song, C., & Zeng, P. (2023). Efficient federated learning on resource-constrained edge devices based on model pruning: T. Wu et al. *Complex & Intelligent Systems*, 9(6), 6999-7013. <https://doi.org/10.1007/s40747-023-01120-5>
- Li, A., Markovic, M., Edwards, P., & Leontidis, G. (2024). Model pruning enables localized and efficient federated learning for yield forecasting and data sharing. *Expert Systems with Applications*, 242, 122847. <https://doi.org/10.1016/j.eswa.2023.122847>
- Mughal, F. R., He, J., Das, B., Dharejo, F. A., Zhu, N., Khan, S. B., & Alzahrani, S. (2024). Adaptive federated learning for resource-constrained IoT devices through edge intelligence and multi-edge clustering. *Scientific Reports*, 14(1), 28746. <https://doi.org/10.1038/s41598-024-78239-z>
- Ye, Y., Li, S., Liu, F., Tang, Y., & Hu, W. (2020). EdgeFed: Optimized federated learning based on edge computing. *IEEE Access*, 8, 209191-209198. <https://doi.org/10.1109/ACCESS.2020.3038287>
- Chang, X., Obaidat, M. S., Ma, J., Xue, X., Yu, Y., & Wu, X. (2024). Efficient federated learning via adaptive model pruning for internet of vehicles with a constrained latency. *IEEE Transactions on Sustainable Computing*, 10(2), 300-316. <https://doi.org/10.1109/TSUSC.2024.3441658>
- Chen, Z., Yi, W., Shin, H., & Nallanathan, A. (2023). Adaptive model pruning for communication and computation efficient wireless federated learning. *IEEE Transactions on Wireless Communications*, 23(7), 7582-7598. <https://doi.org/10.1109/TWC.2023.3342626>
- Lee, S., & Jang, H. (2025). BipruneFl: Computation and communication efficient federated learning with binary quantization and pruning. *IEEE Access*, 13, 42441-42456. <https://doi.org/10.1109/ACCESS.2025.3547627>
- Jiang, Y., Wang, S., Valls, V., Ko, B. J., Lee, W. H., Leung, K. K., & Tassiulas, L. (2022). Model pruning enables efficient federated learning on edge devices. *IEEE Transactions on Neural Networks and Learning Systems*, 34(12), 10374-10386. <https://doi.org/10.1109/TNNLS.2022.3166101>
- L Liu, X., Wang, S., Deng, Y., & Nallanathan, A. (2023). Adaptive federated pruning in hierarchical wireless networks. *IEEE Transactions on Wireless Communications*, 23(6), 5985-5999. <https://doi.org/10.1109/TWC.2023.3329450>
- Cao, S., Wu, H., Wu, X., Ma, R., Wang, D., Han, Z., & Zhang, W. (2025). FedDA: Resource-adaptive federated learning with dual-alignment aggregation optimization for heterogeneous edge devices. *Future Generation Computer Systems*, 163, 107551. <https://doi.org/10.1016/j.future.2024.107551>
- Yu, D., Yuan, Y., Zou, Y., Zhang, X., Liu, Y., Cui, L., & Cheng, X. (2025). Pruning-based adaptive federated learning at the edge. *IEEE Transactions on Computers*, 74(5), 1538-1548. <https://doi.org/10.1109/TC.2025.3533095>
- Chen, Z., Jia, C., Hu, M., Xie, X., Li, A., & Chen, M. (2024). Flexfl: Heterogeneous federated learning via apoz-guided flexible pruning in uncertain scenarios. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 43(11), 4069-4080. <https://doi.org/10.1109/TCAD.2024.3444695>
- Liu, X., Ratnarajah, T., Sellathurai, M., & Eldar, Y. C. (2024). Adaptive model pruning and personalization for federated learning over wireless networks. *IEEE transactions on signal processing*, 72, 4395-4411. <https://doi.org/10.1109/TSP.2024.3459808>
- Jiang, Z., Xu, Y., Xu, H., Wang, Z., Liu, J., Chen, Q., & Qiao, C. (2023). Computation and communication efficient federated learning with adaptive model pruning. *IEEE Transactions on Mobile Computing*, 23(3), 2003-2021. <https://doi.org/10.1109/TMC.2023.3247798>
- Zhou, G., Li, Q., Liu, Y., Zhao, Y., Tan, Q., Yao, S., & Xu, K. (2023). FedPAGE: Pruning adaptively toward global efficiency of heterogeneous federated learning. *IEEE/ACM Transactions on Networking*, 32(3), 1873-1887. <https://doi.org/10.1109/TNET.2023.3328632>

- Tariq, A., Sallabi, F., Serhani, M. A., Barka, E. S., Qayyum, T., Din, I., & Syed, I. (2025). Ai-driven lightweight and trustworthy federated learning framework for decision-making in resource-constrained edge-iot. *IEEE Transactions on Consumer Electronics*, 71(4), 12166-12180. <https://doi.org/10.1109/TCE.2025.3628207>
- Ma, Z., Xu, Y., Xu, H., Liu, J., & Xue, Y. (2022). Like attracts like: Personalized federated learning in decentralized edge computing. *IEEE Transactions on Mobile Computing*, 23(2), 1080-1096. <https://doi.org/10.1109/TMC.2022.3230712>
- Jiang, X., & Borcea, C. (2023, June). Complement sparsification: Low-overhead model pruning for federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 37, No. 7, pp. 8087-8095). <https://doi.org/10.1609/aaai.v37i7.25977>
- Kim, M., Saad, W., Debbah, M., & Hong, C. S. (2024). SpaFL: Communication-efficient federated learning with sparse models and low computational overhead. *Advances in Neural Information Processing Systems*, 37, 86500-86527. <https://doi.org/10.52202/079017-2747>
- Niu, Z., Dong, H., Qin, A. K., & Gu, T. (2024). Flrce: Resource-efficient federated learning with early-stopping strategy. *IEEE Transactions on Mobile Computing*, 23(12), 14514-14529. <https://doi.org/10.1109/TMC.2024.3447000>