Decentralized Federated Learning in Partially Connected Networks with Non-IID Data

Xiaojun Cai, Nanxiang Yu, Mengying Zhao*, Mei Cao, Tingting Zhang, Jianbo Lu
School of Computer Science and Technology, Shandong University, China

***Abstract*—Federated learning is a promising paradigm to enable joint model training across distributed data while preserving data privacy. The distributed data are usually not identically and independently distributed (Non-IID), which brings great challenges for federated learning. There have been existing work proposing to guide model aggregation between similar clients to deal with Non-IID data. But they typically assume a fully connected network topology, while new design issues need to be considered when it comes to a partially connected topology. In this work, we propose a probability-driven gossip framework for partially connected network topology with Non-IID data. The main idea is to discover similarity relationship between non-adjacent clients and guide the model exchange to encourage aggregation between similar clients. We explore cross-node similarity assessment and define probability to guide the model exchange and aggregation. Both similarity and communication cost are considered in the probability-driven gossip. Evaluation shows that the proposed scheme can achieve 13.04%-14.24% improvement in model accuracy, when compared with related work.**

***Index Terms*—Decentralized federated learning, Non-IID data, Probability-driven gossip**

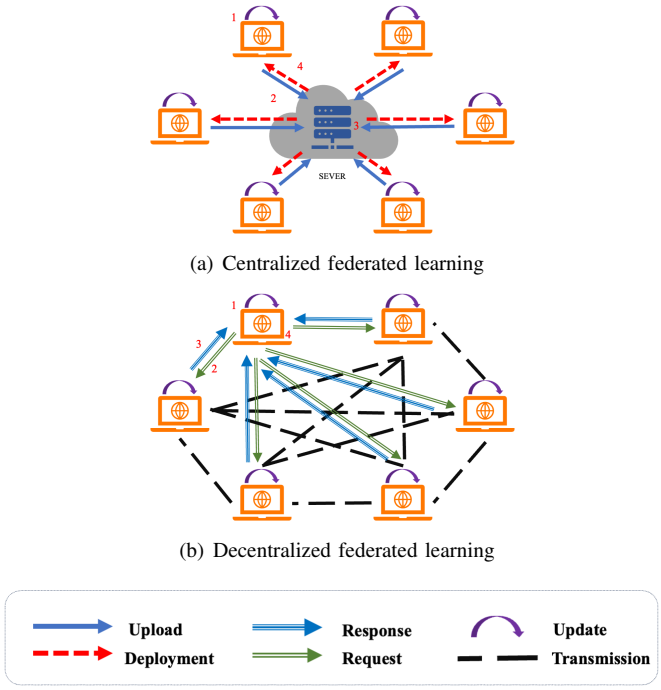
I. Introduction

Federated learning [1] is an emerging machine learning framework that enables deep neural network training across large number of distributed clients in a privacy-preserving manner. As Fig. 1(a) shows, in centralized federated learning, a server deploys a model to selected distributed clients. Then the clients train this model with local private data and upload updated model to the server for aggregation.

A global model can be achieved through this kind of iterative procedure without sharing local data. Federated learning can also be conducted in a decentralized way, as shown in Fig. 1(b). Instead of using a server to orchestrate, clients communicate with neighbours for model exchange and aggregation. Decentralized federated learning is typically more robust to malicious attack or unexpected failures [2] [3].

One challenge federated learning faces is the not identically and independently distributed (Non-IID) data [4], i.e., data distributions in clients are significantly different from each other. This typically leads to local models having various parameter features. The accuracy as well as convergence speed of aggregated model would be highly degraded due to Non-IID data [4] [5]. There are techniques proposed to deal with

This work is supported by Key Research and Development Program of Shandong Province of China (Grant No.2022CXGC020107), National Natural Science Foundation of China (Grant No.61973214), Shandong Provincial Natural Science Foundation (Grant No.ZR2020MF069) and Shandong Provincial Postdoctoral Innovation Project (Grant No.202003005). Mengying Zhao is the corresponding author. (e-mail: zhaomengying@sdu.edu.cn)



(a) Centralized federated learning

(b) Decentralized federated learning

Fig. 1. Centralized and decentralized federated learning.

Non-IID data in federated learning. For example, a small proxy data set can be constructed and shared to reduce the earth mover's distance between clients [5], or adaptive aggregation can be employed to relieve the accuracy drop of the global model [6]. However, a satisfying global model may still be difficult to obtain due to the inherent data heterogeneity. This motivates personalized federated learning, where clients tend to train customized models through communication with others. In centralized federated learning, the server can collect statistical characteristics of data distribution from clients, and then divide them into clusters according to their similarity so that models can be aggregated only from "similar" clients to relieve impact from Non-IID data [7]. The clustering becomes more difficult when it comes to decentralized federated learning since there is no server responsible for global classification.

To solve this problem, there are researches to employ the gossip framework [8] to randomly communicate with neighbours and figure out the similarity by verifying neighbours' models with its own local data [9]. However, they usually assume a fully connected network topology, where each client can

979-8-3503-4859-0/DATE24/© 2024 EDAA

directly reach every other one. But the topology in reality is not always fully connected. If it is partially connected, the existing techniques face new challenges. Firstly, a node can only directly communicate with its neighbours but the similar ones may be far away. Thus it needs new strategies to explore more nodes to find similar clients. Secondly, for model aggregation, non-adjacent clients need to go through several other clients for model passing. In this case, the clients on the path do not aggregate models. Thus, compared with traditional gossip, clients need to have new behaviors and take different actions accordingly. Thirdly, since communicating with faraway nodes induces higher cost, we need to propose strategies with tradeoff between model accuracy and communication cost, and provide the flexibility of tradeoff according to different scenarios.

In this work, we target at the framework design for decentralized federated learning in partially connected network topology with Non-IID data. We make the contributions as follows.

- We analyze and summarize the difference between fully connected and partially connected network topologies in terms of framework design for decentralized federated learning;
- We propose a probability-driven gossip scheme which enables cross-node similarity assessment, adaptive probability update, as well as tradeoff between model accuracy and communication cost for decentralized federated learning with Non-IID data;
- We evaluate the proposed schemes and compare with state-of-the-art related works.

The rest of the paper is organized as follows. Section 2 presents related work. Section 3 presents details of the proposed probability-driven gossip framework. Section 4 gives evaluation results and discussions. We conclude this paper in Section 5.

II. Related Work

Nowadays, there has been a large amount of work to tackle with the challenge of Non-IID data in federated learning. Hao et al. [10] propose a fair federated learning system using data augmentation to generate pseudo-samples of unseen categories using zero-sample data augmentation to mitigate the statistical heterogeneity of the client's data distribution, thus improving the accuracy and homogeneity of the global model. Karimireddy et al. [11] propose a new stochastic controlled averaging algorithm (SCAFFOLD) that uses control variables (variance reduction) to correct for "client drift" in each client update, allowing the global model to learn in a more accurate way.

In the context of decentralized federated learning, Hegedűs et al. [2] compare gossip learning with traditional federated learning and verify the performance of both to be comparable. But the gossip framework faces great challenges with Non-IID data, where there are usually multiple learning objectives to fit personalized features of clients. T. Dinh et al. [12] take a personalized approach, using the Moreau envelope as the client's regular loss function, by limiting the data between the client and the server to a range that ensures that the trained model adapts to the personalized needs while maintaining accuracy. Listo Zec et al. [9] propose an adaptive clustering algorithm. The proposed algorithm finds suitable partners by each client based on similarity estimation of local tasks, thus allowing soft cluster allocation and communication throughout the graph. Onoszko et al. [13] propose an algorithm for clustering similar nodes based on empirical loss. In the first stage, permanent neighbors are found through random communication. In the second stage, a fixed number of these clients are selected for aggregation. The limitation of this approach is that choosing a fixed number of clients may involve noisy neighbours, resulting in unsatisfactory aggregation performance. Zexi Li et al. [14] also propose an adaptive soft clustering algorithm, which matches the uniform cluster neighbors in the first stage with high confidence. In the second stage, a heuristic based on expectation maximization under the Gaussian mixture model similarity assumption is used for clients to discover more neighbors with similar goals. Edvin et al. [15] propose an approach for privacy-preserving node selection. The approach mitigates the risk of inferential attacks by using secure aggregation while enabling effective collaborator identification. This is achieved through adversarial dobby slot machine optimization that exploits the dependencies between different weapons to achieve a privacy-preserving node selection scheme.

However, the above research focuses on fully connected network topology, where each client can directly communicate with all other nodes. This indicates that all clients can exchange and aggregate models with similar nodes within one single step in a straightforward manner. Usually, network topology in reality is not always fully connected, motivating new strategies for decentralized federated learning with Non-IID data.

III. Method

In this section, we first give the formulation of target problem, and then present the proposed decentralized possibility-driven gossip (DPG) strategy.

A. Problem Formulation

Given a decentralized network with n clients, each client i has their own training data distribution $\mathbf{D}_i(x, y)$ over input features x and labels y. Each client trains its own model with parameters $\mathbf{w}_i$, using a loss function $\mathcal{L}$. The goal is that under T rounds of communication, each client i can derive its own model $\mathbf{w}_i$ with the average loss minimized, as shown in Equation (1).

$$\mathcal{L}^* = \min_{\mathbf{w}_i} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\mathbf{w}_i) \tag{1}$$

B. Main Idea

The main differences between the proposed framework and traditional gossip are as follows. Firstly, since the network is partially connected, we need to explore more than adjacent nodes to find similar clients. So we build an L-hop neighbour list for future communication. Here L is a parameter to decide the range of communication, where all nodes within L hops are candidates for communication. Secondly, instead of randomly choosing a neighbour to communicate, we define a probability

TABLE I
NOTATIONS.

n	number of clients
t	current training round
L	maximum number of hops for model exchange
M_i	set of 1-hop neighbors of Client i
$V_i^{L_0}$	set of L_0-th hop neighbors of Client i
U_i	set of all communicating neighbors of Client i
C_i	set of communication costs for Client i
N_i^t	set of clients selected for aggregation by Client i in Round t
$\mathbf{w}_i$	training model for Client i
j	the number of all communicating neighbors of Client i
$\mathbf{s}_{ik}^t$	similarity of Clients i and k in Round t
$\mathbf{p}_{ik}^t$	probability of Clients i and k in Round t
$\mathbf{c}_{ik}$	communication costs of Clients i and k
ρ	penalty factor of communication cost

vector to guide the candidate selection. The node with higher probability has larger chance to be selected. This is used to guide clients to communicate more with similar nodes within certain distance so that model aggregation can be more efficient. Thirdly, we simultaneously consider similarity and communication cost to update the probability vector.

In the proposed framework, for each client i, it maintains an L-hop neighbour list $\mathbf{U}_i$= [$\mathbf{b}_1$, $\mathbf{b}_2$, ..., $\mathbf{b}_j$] ($j \leq n$), a probability vector $\mathbf{p}_i^t$= [$\mathbf{p}_{i1}^t$, $\mathbf{p}_{i2}^t$, ..., $\mathbf{p}_{ij}^t$] (where $\mathbf{p}_{ik}^t$is the probability of choosing Client k for model aggregation with Client i at Round t), and communication cost vector $\mathbf{C}_i$= [$\mathbf{c}_{i1}$, $\mathbf{c}_{i2}$, ..., $\mathbf{c}_{ij}$] ($j \leq n$). All related notations are summarized in Table I.

Each client i executes the following steps. Step 1 corresponds to initialization. Steps 2-4 make up of one round and are repeated for continuous learning. Detailed procedures are shown in Algorithm 1.

Step 1. Establish the L-hop neighbour list $\mathbf{U}_i$ (Lines 2-9 in Algorithm 1), and build the communication cost vector $\mathbf{C}_i$ (Line 10). Initialize its probability vector $\mathbf{p}_{ik}^0$ as $\frac{1}{j}$ for all k (Lines 14-15).

Step 2. Select the communication candidates $\mathbf{N}_i^t$ from $\mathbf{U}_i$ according to the probability vector $\mathbf{p}_i^t$ (Line 19).

Step 3. For each client $k \in \mathbf{N}_i^t$, request for its model and aggregate with model of Client i (Line 20).

Step 4. Calculate similarity between Client i and Client k, denoted as $\mathbf{s}_{ik}^t$. And accordingly update the probability $\mathbf{p}_{ik}^t$ based on $\mathbf{s}_{ik}^t$ and $\mathbf{c}_{ik}$ (Lines 21-25).

Repeat **Steps 2-4**.

In the following, we will give more details of similarity assessment and probability update.

C. Cross-node Similarity Assessment

In partially connected network topology, we need to explore more than adjacent neighbours to discover similar nodes for model aggregation. Thus strategies in fully connected networks cannot be directly applied to partially connected scenarios.

The first question is how to obtain the model of another node in partially connected network. After deciding the communication candidates[1], one node i can broadcast to ask for the model of k. If i cannot find k in its neighbour, its neighbour would help to broadcast until k responds. Then the model of k will be passed to i across in-between nodes. So in addition to traditional actions of *push* and *pull*, in the proposed strategy, nodes need another action of *pass-through* to enable cross-node model exchange.

The second question is how to evaluate the similarity between two clients. We adopt the loss-based scheme, which verifies the model from Client k using local data of Client i, and the loss indicates the similarity between Clients i and k. So the similarity between i and k can be calculated as Equation (2), where $\mathbf{s}_{ik}^t$ represents the similarity of Clients i and k in Round t.

$$\mathbf{s}_{ik}^t = \frac{1}{\mathcal{L}(\mathbf{w}_k)} \tag{2}$$

Since loss values may fall into different ranges during the procedure of federated learning, we use the normalized grade as the final similarity assessment, as shown in Equation (3), where

Algorithm 1 DPG algorithm

```
1: function MAIN
2:     for each client i from n do
3:         U_i=M_i;
4:         L_0=2;  // Initialize L_0 to build L-hop list
5:         while (L_0 ≤ L) do
6:             V_i^{L_0} ← Obtain Client i's L_0-th hop neighbors;
7:             U_i.append(V_i^{L_0});
8:             L_0++;
9:         end while
10:        Compute communication cost C_i in U_i;
11:    end for
12:    for each Round t=1,2,3...T do
13:        for each client i from n do
14:            if Round t=0 then
15:                p_ik^t = 1/j;
16:            else
17:                p_ik^t=p_ik^{t-1};
18:            end if
19:            N_i ← Client selection based on probability
                       p_i^t;
20:            w_i ← Aggregate the models in N_i;
21:            for each node k from N_i do
22:                s_ik^t ← 1/L(w_k);
23:                s̄_ik^t = s_ik^t / Σ_{j=1}^n s_ij^t;
24:                p_ik^t = s̄_ik^t − ρ × c_ik;
25:            end for
26:        end for
27:    end for
28: end function
```

[1]Note that candidates are selected from the L-hop neighbour list. A bigger L indicates wider range for similarity exploration, at the cost of higher communication cost.

m represents the number of clients selected for communication in this round.

$$\bar{\mathbf{s}}_{ik}^t = \frac{\mathbf{s}_{ik}^t}{\sum_{j=1}^{m} \mathbf{s}_{ij}^t} \tag{3}$$

D. Probability-driven Gossip

Given the principle of aggregating models from similar nodes, we can define probability to select nodes for communication during the gossip procedure, where highly similar nodes associate with high probability to be selected. However, in partially connected topology, similar nodes may be far away from each other, indicating high communication cost. Thus in the proposed probability-driven gossip, we simultaneously consider similarity and communication cost. The goal is to have personalized models accurately trained with acceptable communication cost.

At first, all probabilities are initialized as the same, indicating all nodes in the L-hop neighbour list have the same chance to be selected for model aggregation. Along with the federated learning, probability can be updated according to Equation (4), where ρ is a positive number as a penalty factor for communication cost consideration.

$$\mathbf{p}_{ik}^t = \begin{cases} \bar{\mathbf{s}}_{ik}^t - \rho \times \mathbf{c}_{ik} & \bar{\mathbf{s}}_{ik}^t > \rho \times \mathbf{c}_{ik} \\ 0 & \bar{\mathbf{s}}_{ik}^t \leq \rho \times \mathbf{c}_{ik} \end{cases} \tag{4}$$

The updated probability in this round will be inherited to next round for communication candidate selection. With the defined probability, those clients with high similarity within acceptable range are likely to be selected for model exchange and aggregation. Note that other nodes, e.g., with small similarity or faraway ones, still have some chance to be selected so that broader range of nodes can be explored to figure out similar ones. ρ can be determined according to the optimization objective. If the communication cost is not the main concern and highly accurate models are more desired, ρ can be defined as a small number, and vice versa.

Consider an extreme case that a node having all its 1-hop neighbours dissimilar with it. The proposed cross-node assessment and probability-driven schemes are able to help it explore more nodes to find appropriate ones for aggregation.

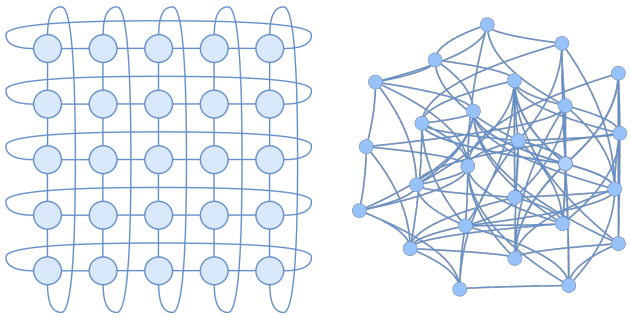
IV. Experiments

In this section, we conduct evaluations to verify the efficacy of the proposed decentralized probability-driven gossip (DPG) scheme, and give discussions.

A. Experimental Settings

Baseline: We compare the proposed DPG with four baselines. The first is Random, which applies traditional gossip framework for random model exchange. The second is Local, where each client generates a model using its local data without communicating with others. The third is decentralized adaptive clustering (DAC) [9], which is proposed for fully connected topology and each client only needs to communicate with its direct adjacent neighbours. The fourth is Oracle, an ideal approach assuming all data distribution information available and only highly similar clients communicate for model aggregation.

Network topology: We considered two network structures. One is 2D-torus, where each point regularly connects to its nodes in four directions. For generality, we also construct a random topology, where connections are randomly defined and number of neighbours of each node is set as 5 to 10. There are totally 100 nodes, i.e., clients in each topology.



(a) 2D-torus (b) Random topology

Fig. 2. Examples of network topologies.

Dataset: We employ Fashion-MNIST as the dataset. To make fair comparisons, we adopt the same rotation transformation as [9]. Specifically, the data in Fashion-MNIST are uniformly divided into four partitions. Each partition has images been rotated by $0°$, $90°$, $180°$ and $270°$, respectively, i.e., four classes. Then images in each partition are further divided into 25 subsets to assign to 25 clients. Thus each client only contain one class of data.

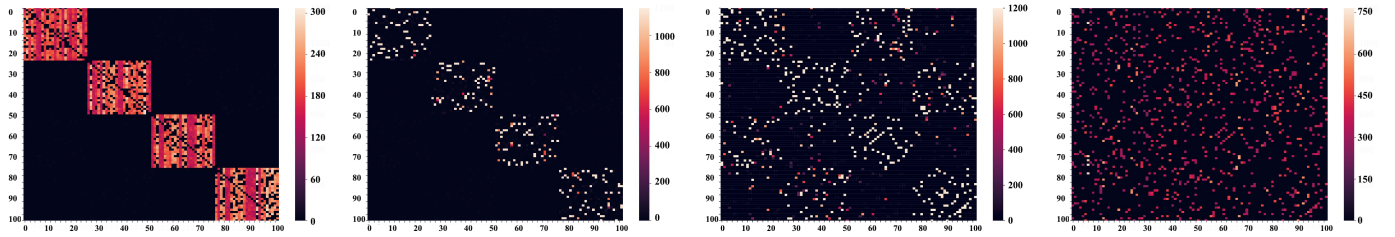
Model: A CNN with Relu activation and maximum pooling is used as target model. The local epoch is set as 8. The Adam optimizer is also applied. The learning rate η is set to be 1×10^{-5}. All tested strategies use FedAvg for model aggregation.

B. Experimental Results

After the federated learning, we evaluate the accuracy of model obtained in each client and show the average accuracy

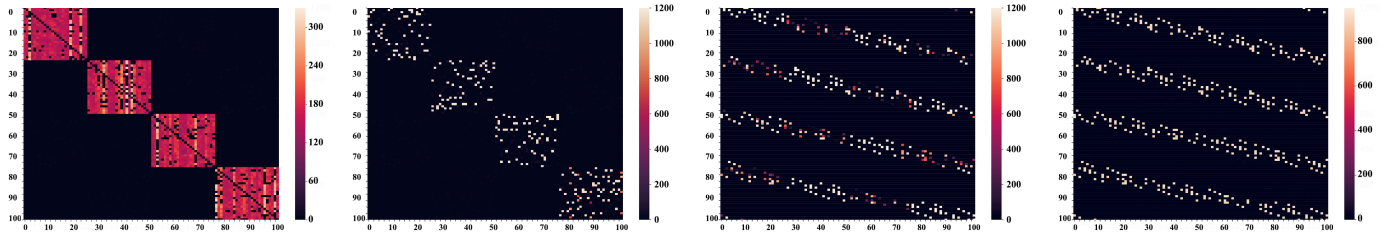
TABLE II
Results in model accuracy.

Method	0°	90°	180°	270°	Mean
			2D-torus		
Random	63.05	63.19	63.66	63.07	63.24
Local	62.46	61.74	58.15	59.38	60.43
DAC	63.82	64.61	65.45	65.58	64.87
DPG	73.74	73.46	74.75	74.51	74.11
Oracle	77.50	78.41	77.81	77.49	77.80
			Random Topology		
Random	66.55	66.02	64.69	64.04	65.32
Local	62.46	61.74	58.15	59.38	60.43
DAC	67.28	67.61	63.70	64.27	65.72
DPG	73.99	74.92	74.28	73.97	74.29
Oracle	76.85	78.43	78.21	76.56	77.51



(a) Oracle (b) DPG (c) DAC (d) Random

Fig. 3. Communication behavior with Random topology.



(a) Oracle (b) DPG (c) DAC (d) Random

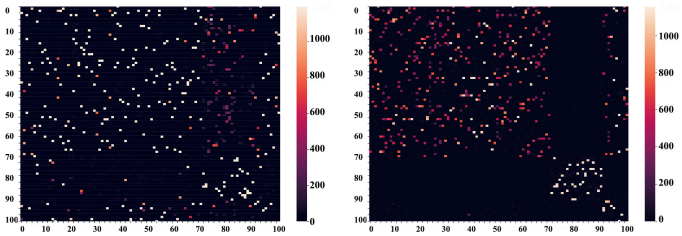
Fig. 4. Communication behavior with 2D-torus.

in each class (0°, 90°, 180° and 270°) in Table II. Results show that the proposed DPG achieves average model accuracy of 74.11% on 2D-torus, which is 14.24% higher than DAC. On random topology, accuracy improvement over DAC is 13.04% on average. This confirms the efficacy of the proposed probability-driven gossip based on cross-node model exchange. Besides, among the first four strategies, DPG has the closest performance to Oracle. They have model accuracies with around 4% difference. This is because that DPG relies on L-hop neighbour list as range of communication, which may not cover all similar clients, and thus degrade the model accuracy to some extent.

C. Discussions

1) Communication behavior: In order to discover communication behaviors during decentralized federated learning, we visualize the number of communications between clients in Fig. 3. Both x and y coordinates represent index of clients. As Fig. 3 (a) shows, ideally, clients only communicate with nodes in the same class, e.g. Clients 0-24, 25-49, 50-74, and 75-99. As depicted in Fig. 3 (b), the proposed DPG basically maintains this trend, but also with a small number of communications with clients from other classes. This is inevitable because it needs to explore at the initial stage and build the similarity information gradually. DAC has more communications between different classes because it lacks of cross-node mechanisms to figure out more similar nodes. Obviously, Random has no patterns of communication (Fig. 3(d)).

Similar conclusion can be obtained from Fig. 4 with 2D-torus topology. Here Random has regular patterns since 2D-torus has particular connection features, making the communication limited into certain set of nodes.



(a) DAC (b) DPG

Fig. 5. Communication cost with more unbalanced setting in Random topology.

2) Test with more unbalanced setting: To further test the performance of DPG, we also conduct another group of evaluation, taking Random topology as an instance. We re-define the four classes to have rotations of 0°, 180°, 350° and 10°, where classes with 0°, 350° and 10° rotations are likely to have similar features while images with 180° are quite different. The number of clients related to four classes are set to be more unbalanced, too, which are 70, 20, 5, and 5, respectively.

The communication behavior is shown in Fig. 5. It can be seen that in DPG, more communications occur among Clients 0-69, and Clients 70-89. There are also significant amount of communications between Clients 90-99 and Clients 0-69. This is because images with rotation of 350° and 10° exhibit close feature with those of 0°, so DPG regards them as similar and guides model exchange among them. As a result, DPG delivers 9.49% higher model accuracy when compared with DAC, as listed in Table III. Fig. 6 compares the model accuracies of the same node in DAC and DPG. DPG shows faster convergence speed and stable accuracy increase during federated learning.

TABLE III
MODEL ACCURACIES WITH MORE UNBALANCED SETTING.

Method	Random topology 0°	180°	350°	10°	Mean
DPG	76.83	70.70	73.71	73.04	73.57
DAC	74.12	51.78	71.83	71.06	67.19

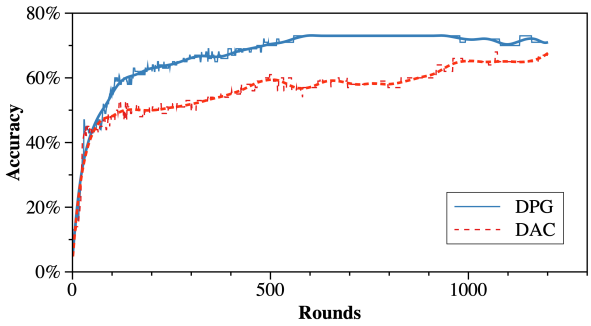


Fig. 6. Model accuracy of DPG and DAC.

3) Tradeoff between model accuracy and communication cost: We implement a strategy which only consider similarity when choosing clients to communicate, noted as DPG-S in Table IV. In DPG-S, when we use 2-hop neighbour list, the average numbers of model communication with 1-hop and 2-hop neighbours are 0.47 and 2.53 per client per round, respectively on average. This leads to an average communication cost of 5.53δ. In the same setting, the proposed DPG delivers 26.76% reduction in communication costs. When 3-hop neighbour list is applied, DPG achieves 9.53% communication cost reduction when compared with DPG-S. Note that the cost include all communications for broadcasting, model request, and also model transfer. Among them, model transfer has the most significant amount of data exchange, dominating the communication cost. So we use cost for model transfer to represent the communication cost, and it is proportional to hop distance.

DPG-S achieves 74.33% of model accuracy on average under tested cases with L=2 and 3. DPG delivers 73.96% on average, which is only 0.5% lower than DPG-S. This confirms that the proposed DPG is able to effectively trade off model accuracy and communication cost.

TABLE IV
AVERAGE COMMUNICATION CONSUMPTION FOR EACH CLIENT IN EACH ROUND.

Method	1-hop	2-hop	3-hop	Comm.cost	Reduction
			$L=2$		
DPG-S	0.47	2.53	0	5.53δ	–
DPG	0.49	1.78	0	4.05δ	26.76%
			$L=3$		
DPG-S	0.43	1.64	0.93	6.50δ	–
DPG	0.32	1.58	0.80	5.88δ	9.53%

V. CONCLUSION

In this paper, we propose a probability-driven gossip framework for decentralized federated learning with partially connected network topology. Unlike fully connected topology, where clients can directly communicate with all other nodes, in partially connected topology, similar nodes may be located far away. So we propose a cross-node similarity assessment scheme to enable looking for similar nodes beyond adjacent neighbours. Besides, we define probability by simultaneously considering similarity and communication cost, and use probability to guide model exchange and aggregation, so that models of similar clients can be aggregated with acceptable communication overhead. We conduct several groups of evaluations to compare the proposed scheme with both traditional gossip framework and state-of-the-art strategies. Results show that the proposed probability-driven gossip can deliver 13.04%-14.24% improvement in average model accuracy, when compared with state-of-the-art work.

REFERENCES

[1] Brendan McMahan, Eider Moore, Daniel Ramage, et al. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[2] István Hegedűs, Gábor Danner, and Márk Jelasity. Gossip learning as a decentralized alternative to federated learning. In *IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS)*, pages 74–90. Springer International Publishing, 2019.

[3] Dongxiao Yu, Zongrui Zou, Shuzhen Chen, et al. Decentralized parallel sgd with privacy preservation in vehicular networks. *IEEE Transactions on Vehicular Technology*, 70(6):5211–5220, 2021.

[4] Peter Kairouz, H Brendan McMahan, Brendan Avent, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[5] Yue Zhao, Meng Li, Liangzhen Lai, et al. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.

[6] Yongheng Deng, Feng Lyu, Ju Ren, et al. Improving federated learning with quality-aware user incentive and auto-weighted model aggregation. *IEEE Transactions on Parallel and Distributed Systems*, 33(12):4515–4529, 2022.

[7] Lei Yang, Jiaming Huang, Wanyu Lin, et al. Personalized federated learning on non-iid data via group-based meta-learning. *ACM Transactions on Knowledge Discovery from Data*, 17(4):1–20, 2023.

[8] Róbert Ormándi, István Hegedűs, and Márk Jelasity. Gossip learning with linear models on fully distributed data. *Concurrency and Computation: Practice and Experience*, 25(4):556–571, 2013.

[9] Edvin Listo Zec, Ebba Ekblom, Martin Willbo, et al. Decentralized adaptive clustering of deep nets is beneficial for client collaboration. In *International Workshop on Trustworthy Federated Learning*, pages 59–71. Springer, 2022.

[10] Weituo Hao, Mostafa El-Khamy, Jungwon Lee, et al. Towards fair federated learning with zero-shot data augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3310–3319, 2021.

[11] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, et al. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR, 2020.

[12] Canh T Dinh, Nguyen Tran, Josh Nguyen, et al. Personalized federated learning with moreau envelopes. *Advances in Neural Information Processing Systems*, 33:21394–21405, 2020.

[13] Noa Onoszko, Gustav Karlsson, Olof Mogren, et al. Decentralized federated learning of deep neural networks on non-iid data. *arXiv preprint arXiv:2107.08517*, 2021.

[14] Zexi Li, Jiaxun Lu, Shuang Luo, et al. Towards effective clustered federated learning: A peer-to-peer framework with adaptive neighbor matching. *IEEE Transactions on Big Data*, 2022.

[15] Edvin Listo Zec, Johan Östman, Olof Mogren, et al. Private node selection in personalized decentralized learning. *arXiv e-prints*, pages arXiv–2301, 2023.