Effective Edge Server Placement for Efficient Federated Clustering

Sungwoong Yeom, Shivani Sanjay Kolekar, Kyungbaek Kim
Department of Artificial Intelligence Convergence
Chonnam National University
Gwangju City, South Korea
yeomsw0421@gmail.com, shivanikolekar@gmail.com, kyungbaekkim@jnu.ac.kr

***Abstract*—Recently, research on federated clustering has been actively studied to improve the performance of federated learning to solve the non-i.i.d issue. Federated clustering makes clusters with members who has similar characteristics of data which is used as inputs of federated learning, and each cluster trains an artificial intelligence model in a federated manner. However, if distances between members of a cluster configured through federated clustering is long in a network, the overhead related to federated learning becomes larger than expected and it may be lose the network cost benefits of federated learning. In this paper, we propose a DTW(Dynamic Time Warping) based federated clustering and MIP(Mixed Integer Programming)-based edge server placement in order to reduce the network overhead of federated learning caused by federated clustering under non-i.i.d setting.**

***Index Terms*—Federated Clustering, Edge Server Placement, Federated Learning, Building Electricity Demand Prediction, LSTM**

I. Introduction

With the increase in renewable electricity generation and decentralization of the market, in the energy field, the building electricity demand forecasting technique is an essential research to balance the electricity demand and supply in buildings and to maintain a stable load on the power grid [1]. The technique generalizes a time-series deep learning neural network by using the collected electricity demand profiles and then redistribute it to buildings. However, the centralized strategy is expensive in terms of communication costs. Also, because the client sends data to the central server, the personal information of client may be infringed.

In order to solve the privacy infringement problem and expensive communication cost problem, a federated learning approach has been adopted [2]. The federated learning is a distributed machine learning technique in which buildings participating in model learning cooperate to learn a global model under the control of a central entity. The federated learning preserves privacy by sharing the weights of model between local buildings and the central entity instead of electricity demand profiles, and reduces communication costs by adjusting the number of rounds for the federated learning [3]. However, if a non-i.i.d(i.e. not independent, equally distributed) problem such as statistical heterogeneity by irregular occupants behavior and weather, the convergence and performance of the global model may be deteriorated.

In order to alleviate the non-i.i.d problem, a clustered federated learning technique was proposed [4]. The clustered federated learning divide into subgroups that show similar electricity consumption patterns in advance, and performs the federated learning between buildings in the same cluster. However, because the electricity demand pattern is markedly changed by the seasons, it is recommended to perform clustering again when the season changes. Accordingly, before federated learning, it is necessary to apply the federated learning approach to clustering in order to preserve the privacy of building residents.

Recently, a federated k-means clustering technique was proposed to extract electricity consumption patterns considering privacy [5]. This technique divides subgroups by sharing the centroid gradient of cluster between each client and central server. However, when the weather and electricity demand patterns become irregular, the performance of conventional federated clustering may be lower. After the federated clustering, the location of the edge server to perform the cluster federated learning is not suitable, and the network delay can be increased. Accordingly, it is necessary to consider the arrangement of edge servers to perform cluster federated learning.

In this paper, we propose the edge server placement technique for communication-efficient federated learning by reducing access delay each cluster. We propose two methods for improving the performance of federated learning. The first method is a federated K-means using DTW[6] for considering an environment where data distortions from electricity consumption patterns can be observed due to irregular resident behavior and weather. The second method is MIP-based edge server placement by cluster to find the optimal location of edge server by minimizing the access delay of global server in federated learning.

II. Related Work

The federated learning, a distributed machine learning technique, is being activated in a smart grid environment [3]. The performance of model in the federated learning is deteriorated by the non-i.i.d problem. In order to improve the performance of federated learning by alleviating the non-i.i.d issues, the clustering method that groups buildings according to the similarity of electricity consumption patterns is being studied

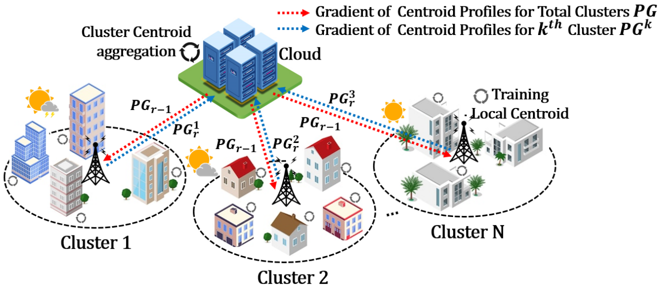
Gradient of Centroid Profiles for Total Clusters PG
Gradient of Centroid Profiles for k^{th} Cluster PG^k
Cluster Centroid aggregation
Cloud
PG_{r-1}
PG_r^1
PG_{r-1}
PG_r^2
PG_{r-1}
PG_r^3
Training
Local Centroid
Cluster 1
Cluster 2
Cluster N

Fig. 1. Federated clustering framework for building electricity consumption pattern extraction

[4]. When a deep neural network trains the consumer patterns identified exclusive to the cluster, the performance of deep learning neural network is greatly improved.

However, because the weights of model are shared instead of data in the federated learning environment, clustering without data collection can be difficult. Although pre-clustered results can be used for the clustered federated learning, new patterns can be observed in the data over time. To extract electricity consumption patterns considering the privacy protection of smart meter data owners, the federated clustering approach has been proposed [5]. This technique applies the federated learning to the k-means++ and considers the centroid initialization of the cluster for improving the federated clustering performance. At this time, the global server of federated learning can improve overall network performance by being located as close as possible to consumers or buildings.

In a edge computing environment, buildings can access an edge server in close proximity within range of a base station [7]. Edge servers can be considered as offloading targets for buildings for the purpose of reducing access latency between building residents and remote clouds. In this paper, we propose federated clustering and MIP-based edge server placement to reduce the network overhead of federated learning in the non-i.i.d environment.

III. System Architecture

Figure 1 shows the federated clustering framework for extracting building electricity demand pattern. Each building can connect to its associated edge servers through a wireless link, and the edge servers and the cloud are connected through optical fiber wired links. In order to form clusters by buildings where similar electricity demand patterns are observed, the cloud performs federated clustering. After federated clustering, for a network cost-effective federated learning task, the number of edge servers is placed in the optimal location as many as the number of clusters.

The framework contains base stations S, edge servers E, buildings B, clusters C, and a cloud. The cloud can be regarded as a data center. Let S = $\{s_1, s_2, ..., s_i\}$, E = $\{e_1, e_2, ..., e_j\}$, C = $\{c_1, c_2, ..., c_n\}$, and B = $\{b_1, b_2, ..., b_m\}$ denote the set of base stations, edge servers, clusters and buildings, respectively. We assume that federated learning task for cluster will be assigned to one of the edge server e on different locations of base stations and each edge server has the same computing resource to process building requests. Also, we assume that each edge server is responsible for a subset of base stations in S to process the building requests and the same base station is not shared between any edge servers.

Algorithm 1: Gradient Sharing-based Federated Clustering with DTW

```
Input: Building profiles $BP$, the number of buildings
       $M$, the number of clusters $N$, the number of
       round $R$, mini-batch size $\mathbb{B}$, learning rate $\gamma$
for $n = 1, 2, ..., N$ do
    Chief randomly selects $BP$ for initializing
     centroid profiles $CP_0^n$
Chief initializes list of building label $L$
for $r = 1, 2, ..., R$ do
    for $m = 1, 2, ..., M$ do
        Worker m selects cluster id $cid$ using DTW :
        $cid = \arg\min_{n \in N} DTW(CP_r, BP_m)$
        Chief updates $cid$ in $L$:
        $L_n = cid$
    Chief & workers convert $CP_r$ to profile gradient
     $PG_r$ during communication between buildings
     and cloud (2),(3)
    for $n = 1, 2, ..., N$ do
        Chief averages $CP_r^n$ using DBA:
        $CP_{r+1}^n = \gamma \frac{1}{\mathbb{B}} DBA(CP_r^n)$
    Chief & workers convert $CP_{r+1}$ to $PG_{r+1}$
     during communication between buildings and
     cloud (2),(3)
Output: list of node label $L$
```

The federated clustering proceeds between cloud and buildings B. The m^{th} building b_m collects the profile of m^{th} building BP_m which have information of building electricity demand and weather. The building profile has three features: electricity demand, indoor temperature, and indoor relative humidity. The building profile can be represented as a matrix in which rows are composed of time index(1h) and columns are features. This building profile is located in each local building and is used as input for federated clustering.

Algorithm 1 shows implementation details of the proposed gradient sharing-based federated clustering with DTW[6]. The input of proposed method is building profiles BP, the number of workers N, the number of clusters N, the number of round R, mini-batch size $\mathbb{B}$, learning rate γ. The cloud initializes the centroid profile CP_0^n for n^{th} cluster by randomly selecting the one of buildings from among the set of buildings B and replacing the profile of m^{th} building BP_m to the initial centroid profile CP_0^n of n^{th} cluster.

Local iteration are implemented in each building. The cluster id cid of the cluster representing the shortest distance among the centroid profile CP_r for N clusters in round r and

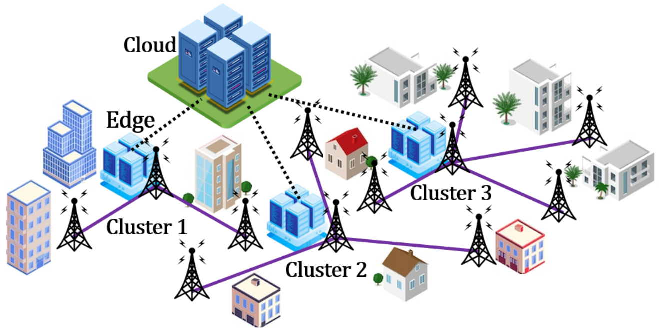

Fig. 2. Edge server placement by federated learning cluster

TABLE I
PERFORMANCE COMPARISON BETWEEN FEDERATED K-MEANS ALGORITHMS

Num Cluster	Method	Avg.DBI	Avg.CHI
5	GSFC_DTW	1.97	13.76
	GSFC	2.33	12.67
6	GSFC_DTW	1.96	14.72
	GSFC	2.14	10.89
7	GSFC_DTW	2.01	14.85
	GSFC	1.83	13.37
8	GSFC_DTW	**1.73**	**14.98**
	GSFC	1.97	11.96

the m^{th} building profile BP_m is selected:

$$cid = \arg\min_{n \in N} DTW(CP_r, BP_m) \tag{1}$$

In order to preserve privacy of clients in federated clustering task, the proposed federated clustering shares the gradient of centroid profile PG_r in round r instead of centroid profile CP_r in round r. Conversion between the gradient of centroid profile PG_r and centroid profiles CP_r in round r is represented as Equations 2 and 3:

$$PG_r = CP_r - CP_{r-1} \tag{2}$$

$$CP_r = CP_{r-1} + PG_r \tag{3}$$

where r denote the r^{th} round, respectively.

Global iteration are implemented in cloud. In order to consider the distortion between centroid profiles of N clusters, the proposed centroid averaging method using DBA[8] in round r is represented as Equations 4:

$$CP_{r+1} = \gamma \frac{1}{\mathbb{B}} DBA(CP_r) \tag{4}$$

The proposed algorithm outputs the list of node id L in which the id of the cluster to which each building belongs is recorded. However, if the buildings belonging to the same cluster are distributed regardless of the climate zone, the overhead of the network may be increased.

Figure 2 shows the proposed edge server placement framework by cluster. The edge server is placed at the location closest to the cluster center among the base station locations. In other words, it is necessary to find a base station location that is the minimum sum of distances between a specific base station and the buildings constituting the n^{th} cluster. In order to optimize the placement of edge server in network, we need to consider the access delay. Because each edge server is located at one of the base stations, the access delay of edge server is proportional to the distance between the base station s and the edge server e. To indicate whether the i^{th} base station s_i will be assigned to the j^{th} edge server e_j, we use a binary decision variable $x_{i,j} \in \{0,1\}$, where $x_{i,j} = 1$ if edge server e_j is assigned to base station s_i; otherwise, where $x_{i,j} = 0$ for all i and j. For each the distance $d_{i,j}$ between i^{th} base station s_i and j^{th} edge server e_j, we compute $\bar{d}_{i,j} = \frac{d_{i,j} - Min(D)}{Max(D) - Min(D)}$ respectively. The proposed MIP-based edge server placement is represented as Equation 5:

$$sid = \arg\min_{i \in I} \sum_{i=1}^{I} \sum_{m=1}^{M} d_{i,m} x_{i,m} \tag{5}$$

where $x_{i,j} \in \{0,1\}$. Each edge server should be assigned to one base station by cluster, and sid is id of the base station where the edge server will be installed. The edge server installed at the optimal location of base station performs the federated learning with buildings in cluster.

IV. EVALUATION

In order to evaluate the proposed edge server placement system, we compare the performance of the proposed MIP-based edge server placement system with federated clustering and random edge server placement system. For the experiment, we use the building electricity demand dataset of CityLearn[9]. In the CityLearn environment, we select 4 climate zones and each climate zones consists of 9 buildings. That is, we use a total of 36 building data sets. Each building dataset consists of electricity demand data measured hourly over a year. We compare the proposed GSFC_DTW(gradient sharing-based federated clustering using DTW) with the GSFC[5]. We set the round of each federated clustering to 20 and the batch size to 20. The primary purpose of clustering is to minimize within-cluster distances and maximize inter-cluster distances. To evaluate the performance of the clustering result, DBI and CHI are used. GSFC_DTW and GSFC use the building electricity demand, the indoor temperature and the indoor relative humidity of building as building profiles to influence federated learning.

Table 1 shows the performance comparison for the federated k-means methods initialized with the same centroid profile according to the number of clusters. In other cases except for the number of clusters of 7, GSFC_DTW is lower than GSFC in DBI which calculated as the ratio of the degree of separation between different groups by comparing the variance within the groups. On the other hand, GSFC_DTW is higher than GSFC in CHI which measures the ratio between cluster variance and intra-cluster variance. When the number of clusters is 8, the lowest DBI and highest CHI are shown. Because of this, buildings in cluster are distributed regardless of the climate

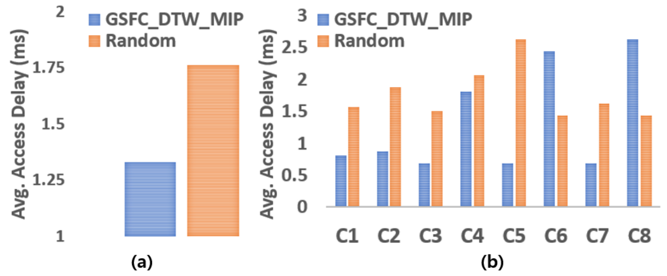

Fig. 3. Comparison of average access delay between different edge server placement algorithms. (a) Average of access delay, (b) Access delay per cluster

zone. Edge servers that perform the federated learning should be placed in optimal locations considering the locations of the buildings constituting each cluster.

Figure 3 shows the comparison of access delay by edge server placement algorithms. In Figure 3(a), the average of access delay is about 0.5ms lower for the proposed GSFC_DTW_MIP method than the random edge server placement method. Because edge servers are randomly placed in 8 out of 36 base stations, the access delay of random edge server placement method is irregular. For better understanding of performance between edge server placement methods, it is necessary to measure the access delay of each cluster. In Figure 3(b), the access delay of the GSFC_DTW_MIP method is lower than the random edge server placement method in the remaining clusters except for the C7(7^{th} cluster) and the C8(8^{th} cluster). Because the locations of buildings in cluster are relatively far away regardless of the climate, clusters C4, C7, and C8 show higher access delay. After edge server placement in network, it is necessary to confirm the performance of clustered federated learning.

The LSTM network for federated learning takes previous 10 hours of the electricity demand profile sequence and two additional input vectors which consist of temperature and humidity of each corresponding hour. With 3 LSTM layers and a linear layer, the network predicts the expected future electricity demand of the next 1 hour. Figure 4 shows the comparison of RMSE(Root Mean Square Error) error between CA(centralized algorithm) and federated learning algorithms. The error rate of the proposed GSFC_DTW_FL is similar with the CA and lower than the basic FL.

V. Conclusion

In this paper, we propose an edge server placement technique considering access delay by each cluster for network efficient federated clustering. We propose two methods for this technique. The first method is a federated K-means algorithm using dynamic time warping for electricity consumer clustering in an environment where data distortions from electricity consumption patterns can be observed due to irregular resident behavior and weather. The second method is mixed integer programming to find the optimal edge server placement among edge servers and minimizing the access delay of edge server.

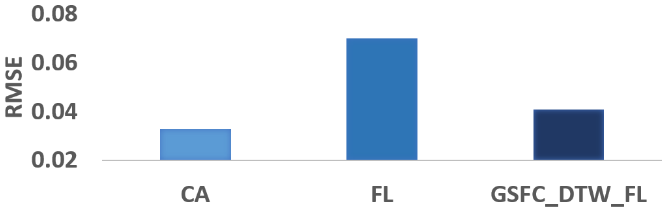

Fig. 4. Comparison of RMSE between centralized algorithm and federated leaning algorithms

Acknowledgment

This research was supported by the Bio & Medical Technology Development Program of the National Research Foundation (NRF)& funded by the Korean government (MSIT) (NRF-2019M3E5D1A02067961)

References

[1] S. Aslam, H. Herodotou, S. M. Mohsin, N. Javaid, N. Ashraf, and S. Aslam, "A survey on deep learning methods for power load and renewable energy forecasting in smart microgrids," *Renewable and Sustainable Energy Reviews*, vol. 144, p. 110992, 2021.

[2] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.

[3] B. Luo, X. Li, S. Wang, J. Huang, and L. Tassiulas, "Cost-effective federated learning design," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, 2021, pp. 1–10.

[4] Y. L. Tun, K. Thar, C. M. Thwal, and C. S. Hong, "Federated learning based energy demand prediction with clustered aggregation," in *2021 IEEE International Conference on Big Data and Smart Computing (BigComp)*, 2021, pp. 164–167.

[5] Y. Wang, M. Jia, N. Gao, L. Von Krannichfeldt, M. Sun, and G. Hug, "Federated clustering for electricity consumption pattern extraction," *IEEE Transactions on Smart Grid*, vol. 13, no. 3, pp. 2425–2439, 2022.

[6] M. Müller, "Dynamic time warping," *Information retrieval for music and motion*, pp. 69–84, 2007.

[7] M. Patel, B. Naughton, C. Chan, N. Sprecher, S. Abeta, A. Neal *et al.*, "Mobile-edge computing introductory technical white paper," *White paper, mobile-edge computing (MEC) industry initiative*, vol. 29, pp. 854–864, 2014.

[8] F. Petitjean, A. Ketterlin, and P. Gançarski, "A global averaging method for dynamic time warping, with applications to clustering," *Pattern recognition*, vol. 44, no. 3, pp. 678–693, 2011.

[9] J. R. Vázquez-Canteli, J. Kämpf, G. Henze, and Z. Nagy, "Citylearn v1. 0: An openai gym environment for demand response with deep reinforcement learning," in *Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 2019, pp. 356–357.