Robust Clustered Federated Learning*

Tiandi Ye[1[0000−1111−2222−3333]], Senhui Wei[1[1111−2222−3333−4444]], Cen Chen[1[2222−−3333−4444−5555]], and Ming Gao[1[2222−−3333−4444−5555]]

East China Normal University
{52205903002, 51205903054}@stu.ecnu.edu.cn, cenchen@dase.ecnu.edu.cn, mgao@dase.ecnu.edu.cn

Abstract. [Incorrect reference format. –ytd] [xxx –wsh] [xxx –cc] [Add some text(some examples)to illustrate that when the distributions across all the clients are not such heterogeneous, it is vital to share the more general knowledge, i.e., consider the relations between the clusters. –ytd] The abstract should briefly summarize the contents of the paper in 150–250 words. [Move towards the model where the entropy is small. –ytd] [Contrastive loss or not? Just l2 distance? –ytd]

Keywords: First keyword · Second keyword · Another keyword.

1 Introduction

Federated Learning [1, 2] is a special distributed machine learning paradigm, which consists of a global central server and hundreds or even thousands of clients in general, which enables the federated system to learn a shared model from the data scattered on different devices while protecting users' data privacy. Generally, federated learning training is an iterative optimization process, and the model requires quite a lot of rounds to meets the requirements or converge. Specifically, in each communication round between the global server and the clients includes the following three steps, which are executed in sequence. (1)The global server initiates a neural network model and randomly selects M out of N clients(if available), and then sends the model weights to the selected clients. (2)Before sending the model weights or updates back to the server, the clients update their models by using the private local data in parallel. (3)The global server gather the clients' latest weights or model updates and update the global model. However, federated learning has also encountered many problems in practical applications, such as data heterogeneity across clients, communication bottlenecks, and data privacy issues. [Particularly, many researches have found that data heterogeneity not only slow the convergence speed and is likely to degrade the model performance. –ytd]
Existing methods trying to tackle [or mitigate the effect of –ytd] the data heterogeneity can be classified into the following two categories. [3–5] learns a single global model to fit the overall distribution. In detail, [3] add a proximal

* Supported by organization x.

item to constrain the locally updated model to be not too far away from the global model, preventing the client model from over-fitting its local distribution, which demonstrate to effective in the presence of statistical heterogeneity. [4, 5] decompose the model to base layers and personalization layers aiming to share the global knowledge while preserving the characteristics of the client's local distribution. [Add `MOCHA` and personlizaiton methods fine-tuning. –ytd] [11] fine-tunes the global model on the clients to fit the local distribution.[Haven't read the paper. –ytd]

A line of works [6–10] focus on grouping clients with similar distribution into a cluster and performing intra-cluster FedAvg [2]. [cite some papers. –ytd] [6] introduces a particular feature *LEGLD* that is sensitive to concept shift but robust to class imbalance and iteratively utilizes the feature to bipartition clients into clusters[copy from the origin paper –ytd].

In this paper, we propose a novel and robust clustered federated learning named `RCFL`, which manages to disentangle the cluster-specific knowledge from the cluster-agnostic knowledge.[Describe some other advantages....difficult. –ytd]

The contributions of this paper are as follows:

– We observe that the existing [clustering or / –ytd] methods in personalized federated learning [fail or overlook –ytd] to disentangle the cluster-specific knowledge from cluster-agnostic knowledge.

– We design a novel and [practical or not? –ytd] framework called `RCFL`, which is able to learn a robust and general global encoder, while disentangle the cluster-specific knowledge from cluster-agnostic knowledge.

– We conduct [extensive –ytd] experiments to validate the effectiveness and robustness of our proposed method and the results show that `RCFL` outperforms [all or ? –ytd] the competing baselines.

2 Related Work

2.1 Clustered Federated Learning

For ease of writing, we denote clustered federated learning as `CFL`. [IFCA [19] proposed a solution which provides multiple global models by clustering local clients into multiple clusters. –wsh]

2.2 Contrastive Learning

[Rewrite the following paragraph. –ytd] Unsupervised learning [cite some papers –ytd] has gained popularity, which tries to learn good representations for downstream tasks. In particular, contrastive learning has received extensive attention and research in the field of unsupervised learning. What the contrastive learning does is to increase the similarity between the representations of similar samples and decrease the similarity between the representations of different samples. [Go on ... –ytd] For simplicity, we choose SimCLR[cite it –ytd] as our contrastive learning framework and performs cluster-level cintrastive learning.

Inspired by SimCLR, We define following cluster-level contrastive learning loss, abbreviated to CLL.[Disorder. –ytd]

$$CLL = \frac{\sum_{k=1}^{K} \sum_{i=1}^{|C_k|} \sum_{j>i} sim(z_i, z_j)}{}$$

where $z = R_W(x)$ and $sim(u, v) = \frac{u^T v}{||u||||v||}$.

2.3 Multi-task Learning

Multi-task learning (MTL) tries to solve multiple learning tasks jointly, while exploiting commonalities and preserving differences across tasks.

[There are a lot of works that adopts shared-private model architecture to xxx. Some xxx soft weights sharing and some xxx hard weights sharing. –ytd]

In a nutshell, we hope to learn a robust global encoder, which can capture the cluster-sharing knowledge that can be mapped to completely different feature spaces when applied to different cluster models, meanwhile it can be mapped to similar fedature space when apply to intra-cluster models.

3 The RCFL

3.1 Problem Statement

Suppose there are N clients, denoted by u_1, ..., u_N, each with a local dataset $D_i = \{x_j, y_j\}_{j=1}^{N_i}$. The intuition behind the CFL is that there are underlying K different distributions in the dataset $D \triangleq \bigcup_{i\in[N]} D_i$, and the clients can be clustered into K clusters naturally, denoted by $C_1, ..., C_K$. The goal of CFL is to learn K models jointly to cover the K distributions, which can be formulated as

$$\{W^k\} = \arg\min_{\{W^k\}_{i=1}^K} \sum_{k=1}^{K} \sum_{i=1}^{N} \mathbb{1}_{[i\in C_k]} p_i L_i(W_i, D_i) \tag{1}$$

where W^k is the model weights shared by the k-th cluster, $\mathbb{1}_{[i\in C_k]} \in \{0, 1\}$ is an indicator function evaluating to 1 iff client i belongs to cluster C_k, $p_i = \frac{N_i}{\sum_j N_j}$, W_i is the model parameters updated locally on the client and $L_i(\cdot)$ is a general loss function for any supervised learning task.

3.2 Motivation

[Considering xxx –ytd]. The optimization in RCFL can be formulate as

$$\{W_k\} = \arg\min_{\{W_k\}_{i=1}^K} \sum_{k=1}^{K} \sum_{i=1}^{N} \mathbb{1}_{[i\in C_k]} p_i L_i(W_k, D_i) + [constraining... - -ytd] \tag{2}$$

3.3 Local Model Structure

[Define the symbols for each layer in the model. –ytd] Inspired by the research of multi-task learning, we design a cluster-level local-global model, consisting of a globally shared encoder across clusters, a cluster-specific projection layer and output layer, which is more flexible where the former aims to extract more general and cluster-agnostic knowledge and the latter ensures the model could retain more personalized and cluster-specific features.
For ease of presentation, we use $W_g(\cdot)$ to denote the cluster-sharing encoder, $W_p(\cdot)$ to denote the projection layer between the global encoder and the output layer, W_o to denote the output layer. In addition, we use $F_W(\cdot)$ to denote the whole network, i.e., $F_W(\cdot) = W_o(W_p(W_g(x)))$ and $R_W(\cdot)$ to denote the network before the output layer, i.e., $R_W(x) = W_p(W_g(x))$, which is the mapped representation vector of input x.

3.4 Algorithm

There are also some researches[cite some papers, like FedPer and LG-FedAvg –ytd] that use similar shared-private model structure trying to mitigate the negative impact of client heterogeneity in federated learning. However, without further constraints on client's model, general knowledge and personalized features are likely to be entangled into the global encoder. [Describe some negative effects of entanglement. –ytd]

In addition, it is intuitive that the feature space of clients among the same cluster are more similar while that of clients belonging to different clusters are far apart.

Therefore, in order to learn a more robust global encoder we design a novel framework called Robust Cluster Federated Learning (`RCFL`).

In the RCFL framework, through utilizing a public auxiliary dataset located on the server, we force the embeddings of the clients' models among the same cluster to be close to each other and pull the embeddings from different clusters farther away, i.e., disperse them in different subspaces in the feature space.

And the auxiliary dataset is a set of unlabelled samples sampled from the global distribution, which can be obtained from the public data or generated using some generative models[cite some generative model papers –ytd].

An overview of `RCFL` is shown in 1.

Some researches [12, 13] have confirmed that the parameters close to the output layer are more expressive about the conditional distribution of the user's training data, therefore [why design the model architecture like this. –ytd]

4 Experiments

In this section, we ...

Algorithm 1: RCFL-Robust Clustered Federated Learning

Input : T, K, learning α, participating ratio ρ, initialize $\{W^k\}_{k=1}^K = \{(W_g^k, W_p^K, W_o^K)\}_{k=1}^K$, auxiliary dataset $\{x_i\}_{i=1}^A$

Output : $\{W^k\}_{k=1}^K$

RunServer()

for *each round* $t = 1, 2, ...$ **do**

  $M = \max(\rho \cdot N, 1)$

  $S_t =$ (random set of M clients)

  Sends $\{W^k\}_{k=1}^K$ to all clients in S_t

  [Distributed Training In Parallel]

  for *each client* $i \in S_t$ **do**

    $W_{i,t+1}, \hat{j} = RunClient()$

  end

  [Contrastive Learning]

  Perfom Constrstive Learning

  for *cluster* $k \in [K]$ **do**

  end

  [Aggregation]

  Divide S_t into K clusters in terms of their cluster estimates, denoted by $C_1, ..., C_K$

  for *cluster* $k \in [K]$ **do**

    $W_p^k = \frac{1}{|C_k|} \sum_{i=1}^M \mathbb{1}_{[i \in C_k]} W_{i,p}$

    $W_o^k = \frac{1}{|C_k|} \sum_{i=1}^M \mathbb{1}_{[i \in C_k]} W_{i,o}$

  end

  $W_g = \frac{1}{|S_t|} \sum_{i=1}^M W_{i,g}$

  [Update Server Models]

  Set $\{W_g^k\}_{k=1}^{k=K} = W_g$

end

RunClient()

Estimate cluster identity $\hat{k} = \arg\min_{k \in [K]} L_i(W^k, D^i)$

Set $W_i \leftarrow W_{\hat{k}}$

for T **do**

  update local model $W_i = W_i - \alpha \nabla_{W_i} L_i(W_i, D_i)$

end

Return W_i and $\hat{j}$ to server

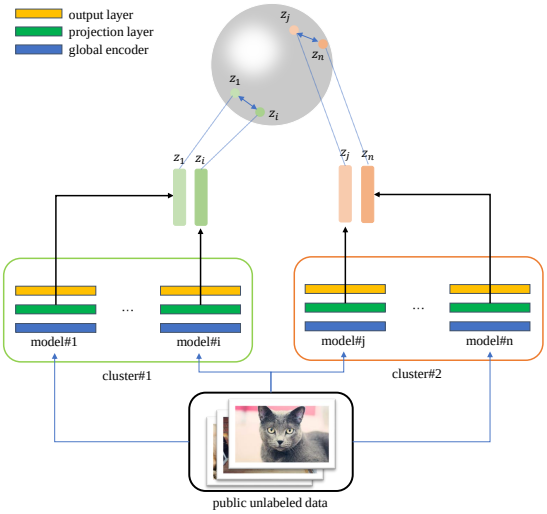

Fig. 1. Overview of RCFL.

4.1 Experimental Setup

Datasets and Models CIFAR10, CIFAR100 and RotatedMNIST.[cite the source –ytd]

Baselines and Evaluation

Implementation and Hyperparameters

4.2 Overall performance

4.3 Algorithm Convergence

4.4 Effects of heterogeneity?

Introduce a metric to quantify the concept shift?

4.5 Effects of the quality [image noise] of the auxiliary unlabelled dataset

4.6 Few-shot clustered federated learning

Use few samples located on the client to determine which distribution the client belongs to.

4.7 Effects of hyper-parameter K, number of clusters

4.8 Figure Illustration

Comparison of baselines in terms of extracted general cluster-agnostic feature and cluster-specific knowledge.

4.9 Pseudo Labels Training on the Sever

Assign the unlabelled samples with pseudo labels.

4.10 Effects of number of layers of the global encoder

Just regard it as a hyper-parameter?

4.11 Replace the contrastive loss with l2-distance based loss

4.12 Reduced the computation overhead on the server

Performing contrastive learning on the global server require $\mathrm{O}(K^2 + M^2)$ computations complexity. Try randomly sample or adopt certain strategy to reduce the computation overhead.

These results demonstrate convincingly that our proposed `RCFL` works.

5 Conclusion

6 Format reference

6.1 A Subsection Sample

Please note that the first paragraph of a section or subsection is not indented. The first paragraph that follows a table, figure, equation etc. does not need an indent, either.

Subsequent paragraphs, however, are indented.

Sample Heading (Third Level) Only two levels of headings should be numbered. Lower level headings remain unnumbered; they are formatted as run-in headings.

Table 1. Table captions should be placed above the tables.

Heading level	Example	Font size and style
Title (centered)	**Lecture Notes**	14 point, bold
1st-level heading	**1 Introduction**	12 point, bold
2nd-level heading	**2.1 Printing Area**	10 point, bold
3rd-level heading	**Run-in Heading in Bold.** Text follows	10 point, bold
4th-level heading	*Lowest Level Heading.* Text follows	10 point, italic

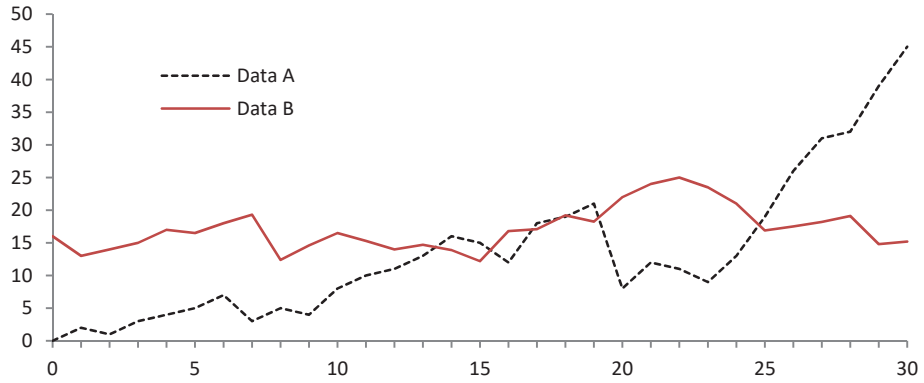

Fig. 2. A figure caption is always placed below the illustration. Please note that short captions are centered, while long ones are justified by the macro package automatically.

Sample Heading (Fourth Level) The contribution should contain no more than four levels of headings. Table 1 gives a summary of all heading levels.
Displayed equations are centered and set on a separate line.

$$x + y = z \tag{3}$$

Please try to avoid rasterized images for line-art diagrams and schemas. Whenever possible, use vector graphics instead (see Fig. 2).

Theorem 1. *This is a sample theorem. The run-in heading is set in bold, while the following text appears in italics. Definitions, lemmas, propositions, and corollaries are styled the same way.*

Proof. Proofs, examples, and remarks have the initial word in italics, while the following text appears in normal font.

For citations of references, we prefer the use of square brackets and consecutive numbers. Citations using labels or the author/year convention are also acceptable. The following bibliography provides a sample reference list with entries for journal articles [14], an LNCS chapter [15], a book [16], proceedings without editors [17], and a homepage [18]. Multiple citations are grouped [14–16], [14, 16–18].

References

1. Kairouz P, McMahan H B, Avent B, et al. Advances and open problems in federated learning[J]. arXiv preprint arXiv:1912.04977, 2019.
2. Kamp M, Adilova L, Sicking J, et al. Efficient decentralized deep learning by dynamic model averaging[C]//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Cham, 2018: 393-409.
3. Li T, Sahu A K, Zaheer M, et al. Federated optimization in heterogeneous networks[J]. arXiv preprint arXiv:1812.06127, 2018.
4. Arivazhagan M G, Aggarwal V, Singh A K, et al. Federated learning with personalization layers[J]. arXiv preprint arXiv:1912.00818, 2019.
5. Liang P P, Liu T, Ziyin L, et al. Think locally, act globally: Federated learning with local and global representations[J]. arXiv preprint arXiv:2001.01523, 2020.
6. Fu Y, Liu X, Tang S, et al. CIC-FL: Enabling Class Imbalance-Aware Clustered Federated Learning over Shifted Distributions[C]//International Conference on Database Systems for Advanced Applications. Springer, Cham, 2021: 37-52.
7. Xie M, Long G, Shen T, et al. Multi-center federated learning[J]. arXiv preprint arXiv:2108.08647, 2021.
8. Sattler F, Müller K R, Samek W. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints[J]. IEEE transactions on neural networks and learning systems, 2020.
9. Sattler F, Müller K R, Wiegand T, et al. On the byzantine robustness of clustered federated learning[C]//ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020: 8861-8865.
10. Duan M, Liu D, Ji X, et al. FedGroup: Efficient Federated Learning via Decomposed Similarity-Based Clustering[J].
11. Zhao Y, Li M, Lai L, et al. Federated learning with non-iid data[J]. arXiv preprint arXiv:1806.00582, 2018.
12. Anand R, Mehrotra K G, Mohan C K, et al. An improved algorithm for neural network classification of imbalanced training sets[J]. IEEE Transactions on Neural Networks, 1993, 4(6): 962-969.
13. Wang L, Xu S, Wang X, et al. Eavesdrop the composition proportion of training labels in federated learning[J]. arXiv preprint arXiv:1910.06044, 2019.
14. Author, F.: Article title. Journal **2**(5), 99–110 (2016)
15. Author, F., Author, S.: Title of a proceedings paper. In: Editor, F., Editor, S. (eds.) CONFERENCE 2016, LNCS, vol. 9999, pp. 1–13. Springer, Heidelberg (2016). https://doi.org/10.10007/1234567890
16. Author, F., Author, S., Author, T.: Book title. 2nd edn. Publisher, Location (1999)
17. Author, A.-B.: Contribution title. In: 9th International Proceedings on Proceedings, pp. 1–2. Publisher, Location (2010)
18. LNCS Homepage, http://www.springer.com/lncs. Last accessed 4 Oct 2017
19. Ghosh A, Chung J, Yin D, et al. An efficient framework for clustered federated learning[J]. arXiv preprint arXiv:2006.04088, 2020.