A Parameter Aggregation Strategy on Personalized Federated Learning

Anonymous ACL submission

Abstract

We investigate the parameter aggregation weights of federated learning (FL), simulate a variety of data access scenarios for experiments, and propose a model parameter weight self-learning strategy for horizontal FL. For application use of this study, a personalized FL network structure model based on edge computing is designed.

1 Introduction

We do research on Federated Learning (FL) on Natural Language Processing (NLP), with emotion classification as the basic task. (Huan et al., 2021) has proposed FedBN-PW-CTC, a federated learning-based Chinese text classification model with FedBN as FL structure, which concluded that for non-iid data, the weights of the parameters at the access side have a greater effect on the accuracy of test sets with similar characteristics, and the increase of the weights leads to the increase of the accuracy of the corresponding test sets, so PW is meaningful for non-iid data access.

2 Related Work

Federated learning (FL) was first proposed by Google in 2016, (Hard et al., 2018) did parameter weighting by weighted averaging and applied to keystroke prediction for GBoard. (Li et al., 2018) proposed FedProx to tackle heterogeneity in federated networks. (Li et al., 2021) proposed FedBN, which accelerates the convergence speed of the model and performs better on non-iid data. Huan W. et al. proposed FedBN-PW-CTC, a federated learning-based Chinese text classification model based on FedBN, which has proved the effectiveness of Parameter Weighting (PW) on non-iid datasets. In addition, (Chen et al.) proposesd FedGame, a multi-player game to study how FL participants make action selection decisions under different incentive schemes. We take into account the influence of relevant incentive mechanisms in the subsequent weighted strategy design.

In FL-based NLP research, (Yuanhe et al., 2021) conducted a study on FL for Chinese word separation. In emotion classification task, (Latif et al., 2020) conducted a study on FL-based speech sentiment classification. As for application use, (Abdellatif et al., 2021) proposed FL for non-homogenous data on IoT. (Ma et al., 2020) proposed a design of Smart Home System based on Collaborative Edge Computing and Cloud Computing, which brings inspiration to design the network structure of FL.

3 Models

3.1 Self-learning Bigram-PW Strategy

FedBN has achieved good results in FL on non-iid data through adding batch normalization layer, and FedBN-PW used parameter weighting method according to the amount of data for each client participating in FL, replacing the parameter averaging approach of FedBN itself.

We conduct experiments based on this FedBN-PW-CTC to verify the effect of different weighting ratios on the accuracy of the client model, and find that there will be a better weighting strategy compared to both. Therefore, we propose a model parameter weighting self-learning strategy for a small number of data clients (clients holding much less data than the average), before the local model is accessed, the central model compares the changes of multiple weight values of the local weights and selects the optimal one as the weight of this client model.

A weight comparison strategy, named self-learning bigram-PW Strategy, is proposed here, as shown in Figure 1. We experimentally verified that too large weights are detrimental to the overall performance of the model, and taking into account the time cost problem (FL has a large time consumption, too many training comparisons will waste a

basicWeight = $\frac{1}{n}$ (n is client number)

parameterWeight = $\frac{client\ data\ volume}{sum\ data\ volume}$

AvgAccuracy = train as FedBN-Avg(basicWeight)

curWeight = parameterWeight

maxAccuracy = AvgAccuracy

while (curWeight < basicWeight) {

    PWAccuracy = train as FedBN-PW(curWeight)

  if (PWAccuracy > AvgAccuracy) {

    optimizeWeight = curWeight

    maxAccuracy = PWAccuracy

    break

  }

  curWeight = 2 × curWeight

}

return maxAccuracy, curWeight

Figure 1: Bigram-PW Strategy Algorithm

lot of time), as well as the incentive mechanism of FL among different clients (the fairness between the high volume clients and the low ones). The average weight is used as the threshold of iteration, and the weight of the client is not increased when the threshold is reached. By determining the best weight at one client by means of weight iteration comparison, we can obtain a more optimal weight for the global model for that client of access within an effective training time. Considering the time consumption, we propose an optimization scheme to adopt the weighting directly when the result of weighting is better than Avg, which saves the time cost and computing cost to some extent, and we verify the effectiveness and feasibility of the method through experiments.

3.2 Personalized FL Edge Network

FL can effectively solve the data silo problem and text classification research has a wide range of application scenarios, so we are inspired by the datasets in our experiments with Chinese text sentiment classification as the task. For the experiments with non-homogenous datasets, we design practical application scenarios for the proposed Personalized FL model, which can be used in the sentiment discrimination part of chatbots and the sentiment analysis part of online opinion monitoring. We use Sentence Vector as the base vector and train it at the personal edge side, with personalized labels, such as age, sex, career etc., as the personal vector at the Chinese text classification, called FedBN-PWP-CTC. Due to the existence of dialects, language expressions are closely related to regions, and considering about the large group network structure of edge computing sinks to regions, we add regional information to the output of the edge layer as edge vector, and train in the cloud considering this vector, called FedBN-PWPE-CTC, and the network structure is shown in Figure 2.

4 Experiment

Our dataset is selected from SMP2020-EWECT competition, where there are 2 non-homogenous datasets, the usual training dataset consists of 30,768 randomly extracted datasets from Weibo, and the virus training set consists of 9,606 data, obtained by keyword extraction from COVID-19, with non-iid characteristics compared to the global usual dataset. Both datasets are divided into 6 categories of emotions, surprise, Happy, Neutral, fear, angry and sad.

4.1 Comparison of Avg and Origin PW

Firstly, we compare the accuracy of the model on the same equal data set with data distributed as iid and non-iid on FedBN and FedBN-PW, which is abbreviated as Avg and PW in the subsequent experiments. We simulated client2 as the access of non-iid data, where client0 and client1 have 15,384 usual data and client2 has 9,606 virus data. Compare to iid-data, we find that the PW model was almost ineffective for iid data. For non-iid data, the Avg accuracy has a 0.3% improvement, considering that it is because the weighted weight is closer to Avg. Therefore, we further verify the effect of PW on different proportions of non-iid data access, and we conduct simulations for data access with the proportion of 0.038, 0.072, 0.135, and 0.238 respectively, using FedBN as a comparison experiment, and the experimental results are shown in Figure 3.

PW has a significant effect on the accuracy improvement of the whole model training when the amount of data on one side is insufficient, and the less the amount of data on the access side, the more obvious the effect of PW.

In order to further verify the effect of PW, we conduct tests on different test sets to verify the effect of the model in different application scenarios, where client0 and client held 15,384 data and

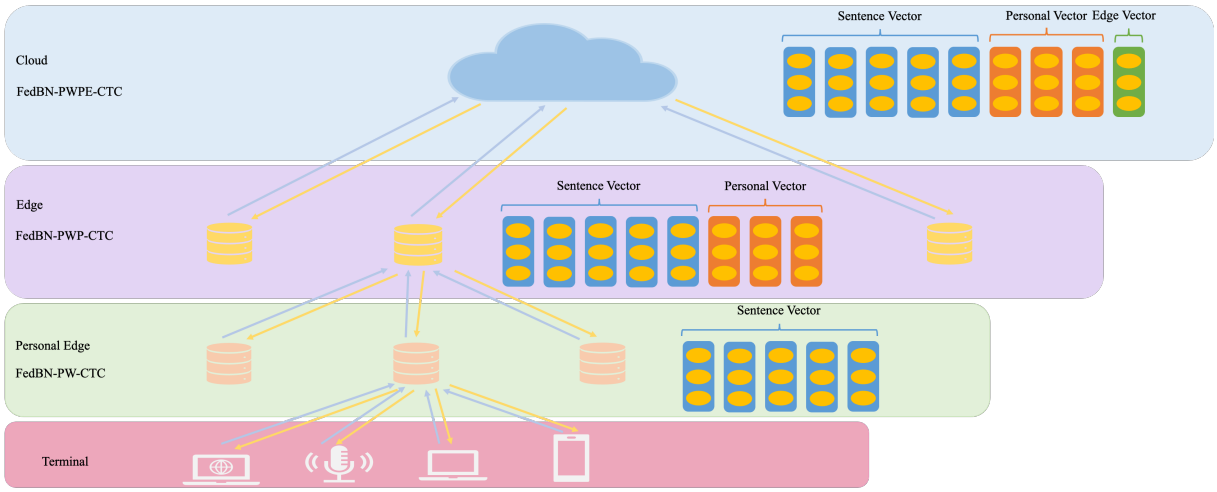

Figure 2: Personalized FL Edge Network Structure

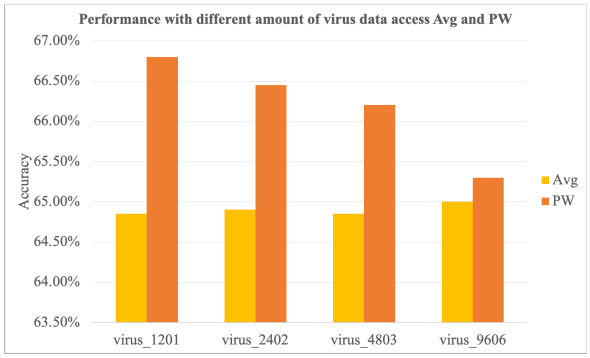

Figure 3: Performance with Different Amount of Data

client2 held 9,606 data. Different test sets were tested on the iid and non-iid training sets, the experimental results are shown in Figure 4. We find

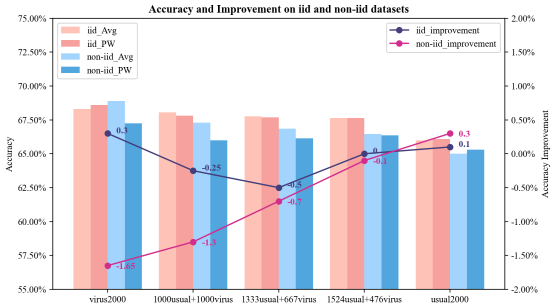

Figure 4: Accuracy of Avg and on Different Datasets

that for iid data, PW does not change significantly compared to Avg on each dataset. For non-iid data, the accuracy of PW is not as high as that of Avg when the proportion of virus data accounts for a certain degree. The analysis here is because the expression of the virus training set itself is closer to the test set of the virus data, which has some correlation with the features of non-iid. However, there is almost no difference in accuracy when the proportion of data in the test set for usual and virus is the same as the proportion for PW, which we define as D.

To further explore this conclusion, we selected the accuracy improvement of PW compared to Avg in the non-iid training set with different data accesses of client2, as shown in Figure 5, test1 to test5 are corresponding to those in the above experiment. When the ratio of usual and virus data

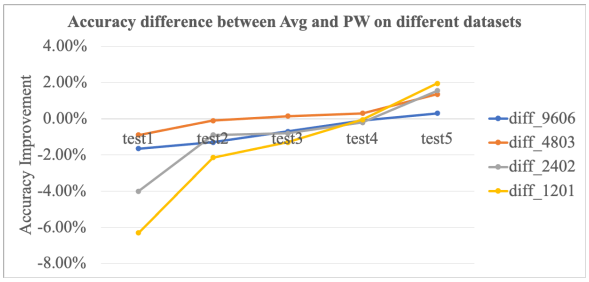

Figure 5: Accuracy Difference between Avg and PW

in the test set reaches D, there is almost no gap between the accuracy of both Avg and PW for different amounts of client2 data, and we can consider D as a threshold value, and PW will perform better than or equal to Avg when the threshold value is reached.

We can obviously find that the performance of PW is very unstable when 1201 virus data are accessed, so we performed different proportional weighting for this training set to verify its effect, and the accuracy change is shown in Figure 6. .

We first verify that FL does work well compared

proprotion	test1	test2	test3	test4	test5	average	variance
1	0.634	0.4965	0.443	0.416	0.359	0.4697	0.87128
0.5	0.6335	0.642	0.6405	0.6385	0.6535	0.6416	0.00436
0.333	**0.6525**	0.6565	0.654	0.654	0.6485	0.6531	0.00069
0.238	0.6455	**0.657**	**0.6605**	0.6625	0.6595	**0.657**	**0.00362**
0.135	0.6255	0.653	0.6565	**0.664**	0.666	0.653	0.02117
0.072	0.599	0.6345	0.6425	0.6535	0.6655	0.639	0.0509
0.038	0.5895	0.635	0.641	0.6535	**0.668**	0.6374	0.07017

Figure 6: Accuracy of Different Parameter Weightings

to local training with a small amount of data access, but it is not true that the higher the percentage of the training set with the same features, the higher the accuracy of the test set, and different weights do affect the performance of the model. Further, it can be found that PW is an effective way in most cases, but the weighting according to the proportion of data we have been proposing is not always the optimal way, and a more optimal strategy can be selected by means of parameter scaling, verifying the necessity of the model parameter weighting self-learning strategy we proposed in 3.1.

4.2 PW Safety Validation

One of the cores of FL lies in data security, and we do not assume that the access client is necessarily honest. Therefore, we simulate the attack access of malicious data to compare the robustness of Avg and PW models to malicious data.

We simulate the stable access side with the hybrid dataset of client0 and client1, holding 15384 data respectively, and client2 for 2 types of malicious data. The first one assumes that all 9606 data of client are marked as angry (actually only 2477 are really angry), and the second one assumes that all 9606 data of client2 are misclassified (sad is judged as surprise, happy is judged as angry, neutral is judged as fear, and vice versa). Similarly, we also do a validation for the access of special data, assuming that the data provided by one client is very single, but the access of this client has a certain reference value, we extracted 2556 data with label as happy in client2. The experimental results are shown in Figure 7.

We can see that the PW can have better training results under the attack of malicious data compared to Avg. Here there is a phenomenon that because the reverse data also has some regularity to follow compared to the other two data, this regularity also has some influence on the local parameters, so the overall performance is better than the other two. Through this experiment, we demonstrate the high value of PW for model robustness and non-iid data. In addition, we conduct experiments with different scales of weighting for special data, and the experimental results are shown in Figure 8. It can

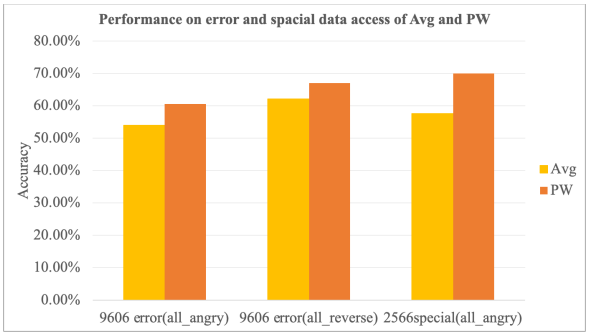

Figure 7: Accuracy on Avg and PW of Malicious Data

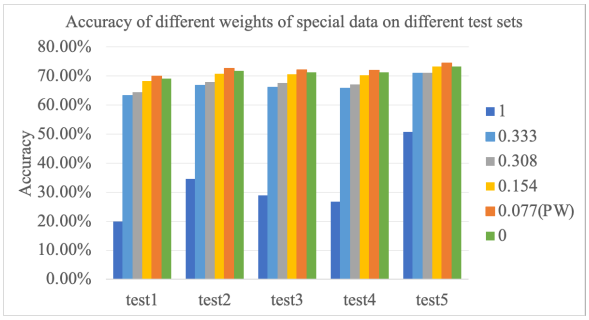

Figure 8: Comparison on Weightings of Special Data

be found that the PW approach weighted by the amount of data outperforms other scales on several proportioned data sets, and its convergence speed is faster. This also verifies the time optimization measure in 3.1, when the data volume weighting is better than Avg, the iteration can be stopped for the reason of saving time cost.

5 Conclusion

We compare several experiments to verify that the FedBN-PW-CTC model outperforms the FedBN model when accessing non-iid data, and the superiority of the model performance becomes more obvious as the accessing client data becomes less. In addition, we propose a model parameter weighted self-learning strategy binary-PW by the performance effect of PW on different test sets, and verify the necessity of this strategy by experiments, and simulate the wrong data and special data to verify the robustness of the model against malicious data attacks and the excellent performance for extreme non-iid data access, further validating the FedBN-PW-CTC model's effectiveness. Finally, we design a Personalized FL networking model based on edge computing for the model for application use.

References

A. A. Abdellatif, N. Mhaisen, A. Mohamed, A. Erbad, M. Guizani, Z. Dawy, and W. Nasreddine. 2021. Communication-efficient hierarchical federated learning for iot heterogeneous systems with imbalanced data.

Z. Chen, Z. Liu, L. N. Kang, H. Yu, Y. Liu, and Q. Yang. A gamified research tool for incentive mechanism design in federated learning.

A. Hard, K. Rao, R. Mathews, Franoise Beaufays, and D. Ramage. 2018. Federated learning for mobile keyboard prediction.

W. Huan, Z. Zerong, L. Ruifang, and G. Sheng. 2021. A federated learning based chinese text classification model with parameter factorization weighting.

S. Latif, S. Khalifa, R. Rana, and R. Jurdak. 2020. Poster abstract: Federated learning for speech emotion recognition applications. In *2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*.

T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V Smith. 2018. Federated optimization in heterogeneous networks.

X. Li, M. Jiang, X. Zhang, M. Kamp, and Q. Dou. 2021. Fedbn: Federated learning on non-iid features via local batch normalization.

Q. Ma, H. Huang, W. Zhang, and M. Qiu. 2020. *Design of Smart Home System Based on Collaborative Edge Computing and Cloud Computing*. Algorithms and Architectures for Parallel Processing.

T. Yuanhe, Chen. Guimin, Q. Han, and Yan S. 2021. Federated chinese word segmentation with global character associations.