Attention-Augmented Fedprox for Enhanced Performance on the Emnist Dataset

Shunyu Yao *

Nanjing University of Posts and Telecommunications, Nanjing, China

* Corresponding Author Email: p22000124@njupt.edu.cn

Abstract. The FedProx algorithm is designed to address challenges related to non-independent and identically distributed (non-IID) data and system heterogeneity in federated learning environments. Despite its effectiveness, performance improvements are still possible when applied to the EMNIST dataset. Incorporating attention mechanisms, such as channel and spatial attention, enhances the stability and accuracy of FedProx models. The channel attention mechanism contributes to improved stability without significant changes in accuracy and loss, especially when a multi-layer perceptron structure is integrated. However, the spatial attention mechanism faces challenges with highly heterogeneous data, leading to instability and poor performance. Modifying data partitioning methods, such as using uniformly distributed data or adjusting the Dirichlet distribution, can mitigate these issues and improve the effectiveness of spatial attention. Ultimately, the findings demonstrate that attention mechanisms can optimize FedProx, but their performance is highly dependent on data heterogeneity, with channel attention showing greater robustness in this context. These findings indicate that attention mechanisms can optimize the FedProx algorithm, but their effectiveness is highly contingent on the degree of data heterogeneity, with channel attention demonstrating greater robustness under varying conditions.

Keywords: Fedprox Optimization, Attention Mechanism, Emnist Dataset.

1. Introduction

Federated learning (FL) has emerged as a powerful paradigm for training machine learning models across distributed devices while ensuring data privacy. However, one of the major challenges in FL is the heterogeneity of the data across different clients. Non-independent and identically distributed (non-IID) data, where the data distribution varies significantly across clients, can cause performance degradation in traditional federated learning algorithms. FedProx (Federated Proximal) has been proposed as a solution to improve the stability and convergence of FL algorithms in the presence of system heterogeneity. The core idea of FedProx is to introduce a proximal term in the local loss function to minimize the discrepancies between local and global model updates, allowing the model to converge faster and with more stability despite non-IID data distributions. Despite its effectiveness, the performance of FedProx can still be further improved, particularly in federated learning environments with more complex datasets like EMNIST.

The EMNIST dataset is an extended version of the MNIST dataset. As an extended handwritten character dataset, it is commonly used for recognizing handwritten letters and numbers. In the study, a federated learning benchmark based on the EMNIST dataset (62 classes) was used, with a client configuration of a total of 100 clients, a local training cycle of 5/round, and 10 clients actively participating in each round of communication (10% client selection rate); Non IID data partitioning uses Dirichlet distribution generation (α=0.1) to simulate highly heterogeneous data distributions (IID degree=0.1).

Before using attention mechanisms, a fixed framework is used for evaluating the Fedprox algorithm. The first step is to build the data loading section: the FMNIST dataset is loaded and partitioned to multiple clients, using Dirichlet distribution to partition the data. The second step is to construct a convolutional neural network suitable for image classification tasks for processing image data, and to train and test it locally. The third step is to build the client part: Use the FedProxClient class to simulate the client in federated learning and introduce the FedProx core mechanism during the training process: add a regularization term to the local loss function and adjust it according to the

regularization coefficient u during the subsequent local training process. The forth step is to build the server part: In each round of training, the server randomly selects a subset of clients from all clients to participate in the training. Then use the weighted average method to merge the model updates of all clients into the global model. The fifth step is to perform the federated training process, starting the training loop after loading data and initializing the server. Select a portion of clients to train using FedProx in each round and return the updated model. The server aggregates updates from clients and updates the global model. After each round of training, evaluate the performance of the global model, record the data, and output it as an image at the end [1].

When using attention mechanisms, an additional attention mechanism section is introduced in the template. Compare the generated images with and without channel and spatial attention mechanisms to analyze the impact of these attention mechanisms on the accuracy, loss, and stability of the Fedprox algorithm on the FMNIST dataset. When comparing images is difficult, process the data directly from each round to better compare subtle differences. At the same time, due to the difficulty in observing stability from images, display the changes in stability under different mechanisms by calculating and comparing the standard deviation. Finally, modify and optimize the initial model or mechanism to verify the reasons for the previous non ideal conditions.

2. Relevant Theories

2.1. Addressing Heterogeneity in Federated Learning

The EMNIST dataset has a non independent and identically distributed data distribution as an extended handwritten character dataset. Different types and quantities of character samples can lead to class bias. In addition, due to the use of Dirichlet to divide the EMNIST dataset based on visual similarity, each category contains fewer types of characters, but each character has a larger sample size. Or some categories contain a large number of samples, while others have fewer samples. All of these have led to a high degree of heterogeneity in data in distributed learning.

2.2. Attention Mechanism

The attention mechanism originated from the study of human vision, focusing on specific parts of information by selecting to ignore certain information. In situations where computing power is limited, using attention mechanisms can better achieve resource allocation and improve algorithm accuracy [2]. Today the common attention mechanisms include channel attention mechanism and spatial attention mechanism.

The channel attention mechanism focuses on the feature importance of different channels. In channel attention, the first step is to calculate the average pooling and maximum pooling outputs of the feature map. In PyTorch, using the avd_pool2d and max_pool2d functions in torch.nn. functional can perform average pooling and max pooling operations.

Average pooling is the process of taking the average of all values within the pooling window to obtain a new value. Its calculation formula is as follows:

$$\text{Average Pooling Output} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{1}$$

(N is the number of elements in the pooling window,x_i is each element in the window)

Maximum pooling is the process of outputting the maximum value within the pooling window. Its calculation formula is as follows:

$$\text{Maximum Pooling Output} = \max(x_1, x_2, \ldots, x_i) \tag{2}$$

Then generate a channel attention coefficient by passing it through two fully connected layers.

Finally, the sigmoid function normalizes the coefficient and outputs a value between 0 and 1, which is used to weight the channels of the input feature map.

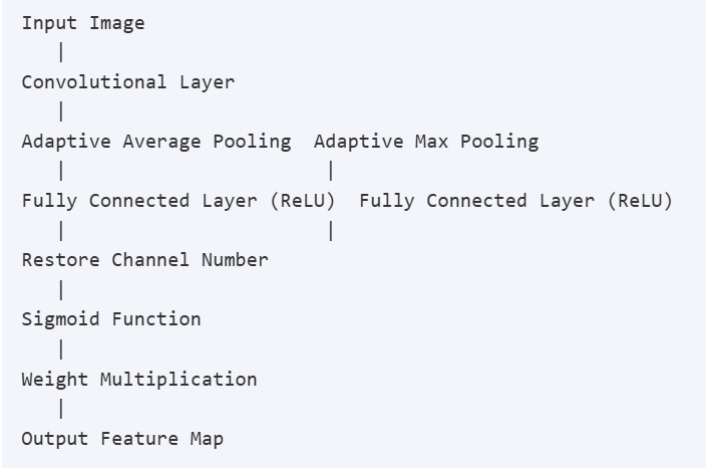


Fig 1. Channel Attention Mechanism Process (Picture credit: Original).

The spatial attention mechanism focuses on the importance of different spatial positions in an image are given in Fig 1 [3]. In channel attention, the first step is to initialize SpatialAttention class with a kernel size of 7, which is an odd number to ensure symmetric convolution operations. Then create two convolutional layers and apply the SpatialAttention module. The output of the convolutional layer is first passed through average pooling and maximum pooling to extract spatial information as shown in Fig 2. Then, connect the results and obtain a single channel feature map through convolution operation. After that, using the Sigmoid activation function to calculate spatial attention weights and multiply element by element with the input feature map to generate a weighted feature map. By multiplying the spatial attention map with the original feature map, important regions can be emphasized, and unimportant regions can be suppressed [4].

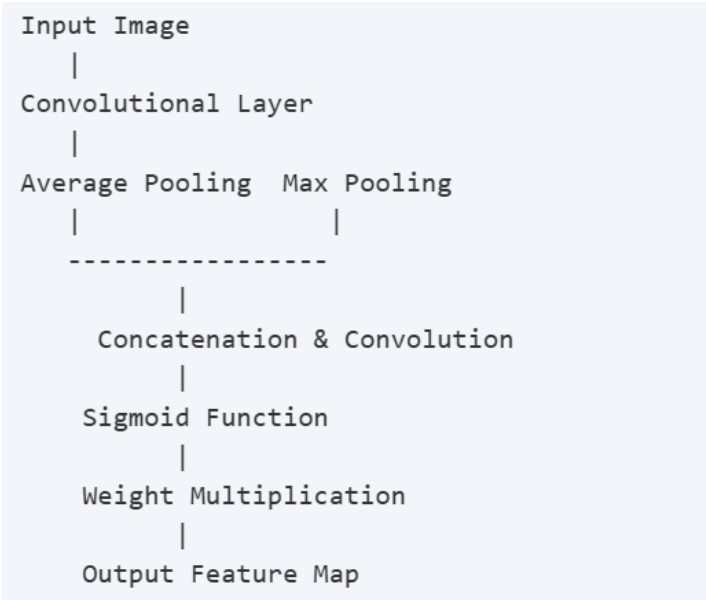


Fig 2. Spatial Attention Mechanism Process (Picture credit: Original).

3. Result Analysis

3.1. Comparative Analysis of Loss and Accuracy in Optimized FedProx Models

Comparing Fig 3 with Fig 4, there was not much change in the results after using the channel attention mechanism. However, when calculating the difference between the two, Fig 5 shows that the initial use of channel attention slightly increases the loss and reduces accuracy. But as the number of rounds increases, the difference in losses gradually decreases, while the difference in accuracy still fluctuates around 3%.

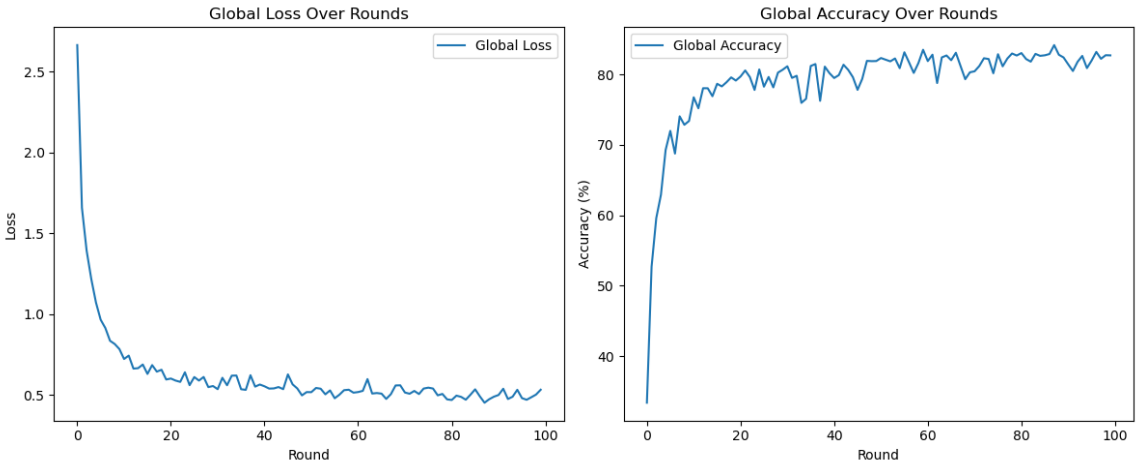


Fig 3. Original results without using attention mechanism (Picture credit: Original).

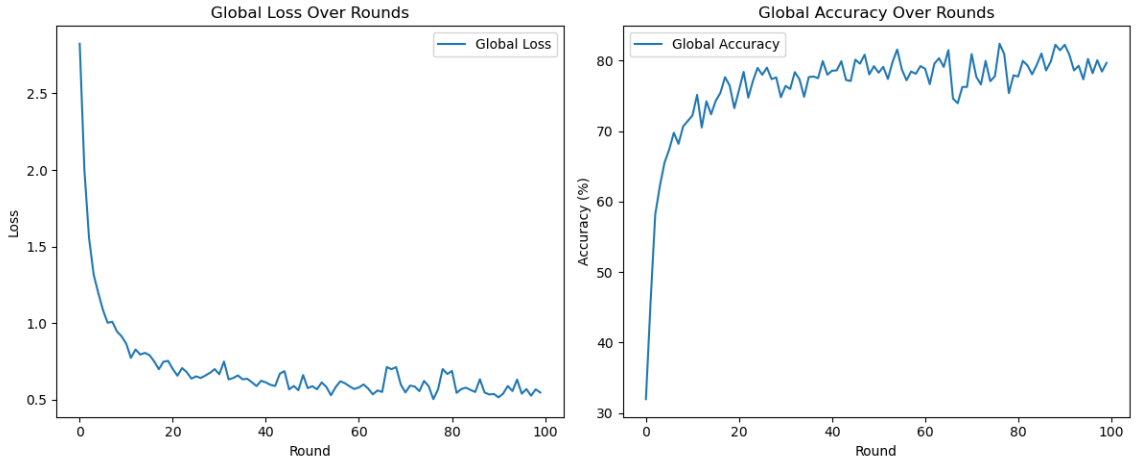


Fig 4. Optimizing results using channel attention mechanism (Picture credit: Original).

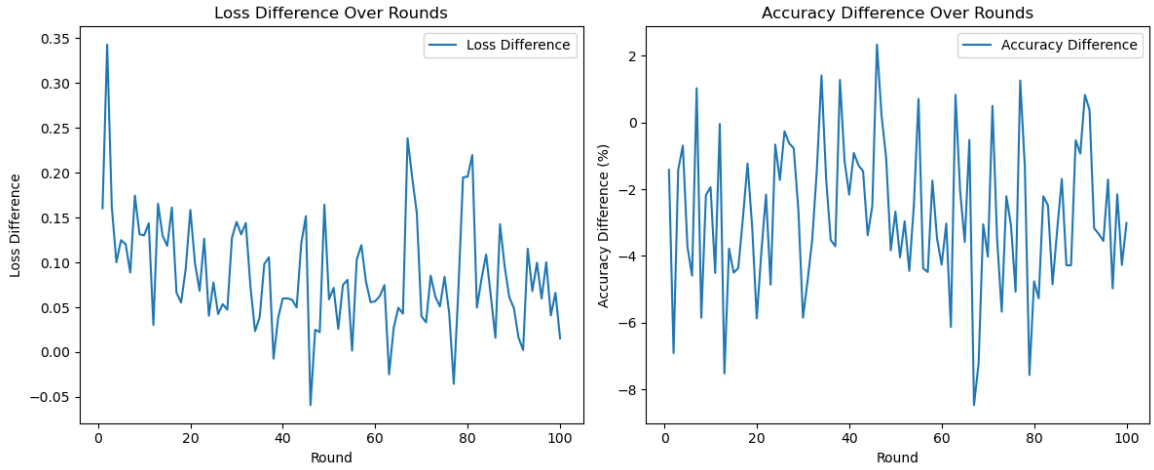


Fig 5. The difference between the data obtained by using channel attention mechanism and the template data (Picture credit: Original).

Comparing Fig 6 with Fig 3, the results showed poor performance after using spatial attention mechanism. After removing the mutated rounds, the global loss of the remaining rounds remained at around 4.0, while the global accuracy remained at around 6%. The accuracy did not improve with the increase of epochs, and the loss did not decrease with the increase of epochs. This indicates that the use of spatial attention mechanisms actually brings more negative effects and is not applicable in this area.

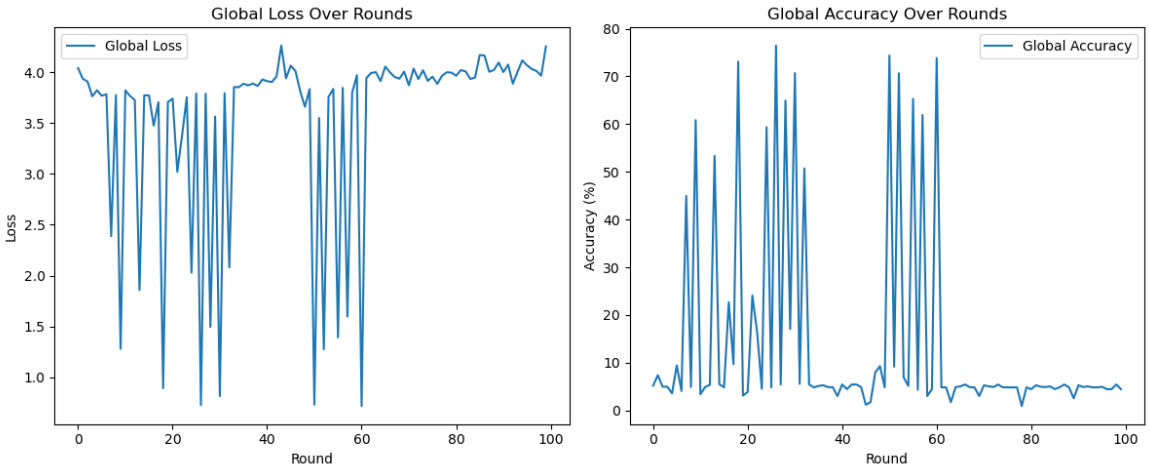


Fig 6. Optimizing results using spatial attention mechanism (Picture credit: Original).

3.2. Stability Comparison: Evaluating the Robustness of the Models

After using channel attention, the stability of the model improved between 30-50 rounds, but decreased between 50-90 rounds. After using spatial attention mechanism, the stability of the model significantly decreased between rounds 5-35 and 45-60.

From the entire 100 rounds, the channel attention mechanism leads to a slight decrease in instability, while the spatial attention mechanism leads to a significant decrease in instability. Calculate the standard deviations of loss and accuracy separately and take the average. The standard deviation of the template is 3.4454, the standard deviation after using channel attention mechanism is 3.7389, and the standard deviation after using spatial attention mechanism is 11.3568.

However, since the ideal accuracy and loss result are a curve, fitting the data first and then calculating the standard deviation of the difference between the ideal value and the actual value can obtain better stability calculation results. Under this processing, the standard deviation of the template is 1.5464, the standard deviation after using channel attention mechanism is 1.4675, and the standard deviation after using spatial attention mechanism is 7.8667. From this perspective, the use of channel attention mechanism slightly improves the stability of the algorithm, while the instability caused by spatial attention remains strong.

3.3. Cause Analysis: Investigating the Impact of Attention Mechanisms on Model Behavior

The use of channel attention mechanism has not significantly improved the performance of Fedprox algorithm. The reason is that the convolutional structure used is relatively simple, and its ability to capture relationships between channels is weak.

The poor performance of spatial attention mechanism is closely related to the heterogeneity of data. When the heterogeneity of data decreases, spatial attention mechanisms can also perform well. Therefore, using uniformly partitioned data instead of Dirichlet partitioned data can avoid the high heterogeneity caused by Dirichlet data partitioning. On this basis, use spatial attention mechanism and compare the results.

3.4. Certainty of Results: Ensuring Robust and Reliable Conclusions

In the process of processing pooling results using channel attention mechanism, using shared multi-layer perceptron structure instead of the original content to improve its performance. The shared multi-layer perceptron structure can learn more complex nonlinear features through multi-layer fully connected layers, thereby better capturing the complex dependencies between channels [2].

Comparing Fig 7 with Fig 3 and Fig 4, the curve is very smooth, and the stability of the results has been greatly improved without causing too much impact on accuracy and loss.

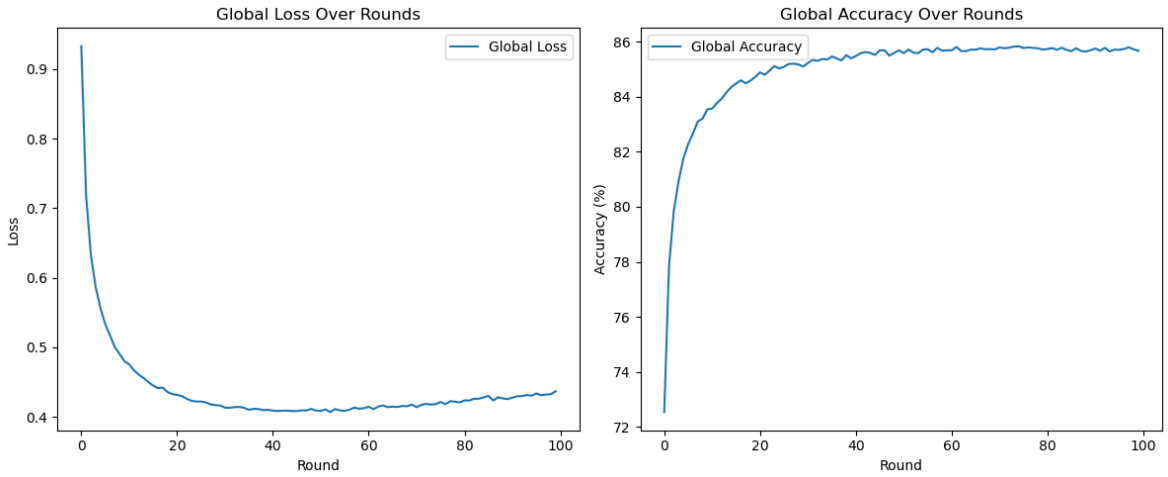


Fig 7. Optimization results of channel attention mechanism after using shared multi-layer perceptron structure (Picture credit: Original).

Due to the high heterogeneity caused by Dirichlet data partitioning, verify the idea by comparing the results using spatial attention mechanism again after reducing heterogeneity.

The first method is changing the value of α in the Dirichlet distribution from 0.1 to 1. When dealing with non independent and identically distributed data, Dirichlet distribution can provide a probability vector for each category, ensuring that samples of each category are allocated proportionally among different clients. Different degrees of data heterogeneity can be simulated by adjusting the hyperparameter $\alpha$of the Dirichlet distribution to control the distribution characteristics of data. As shown in Fig 8, when the heterogeneity of the dataset is reduced, the instability caused by spatial attention mechanism is reduced, and the negative impact on accuracy and loss is also reduced.

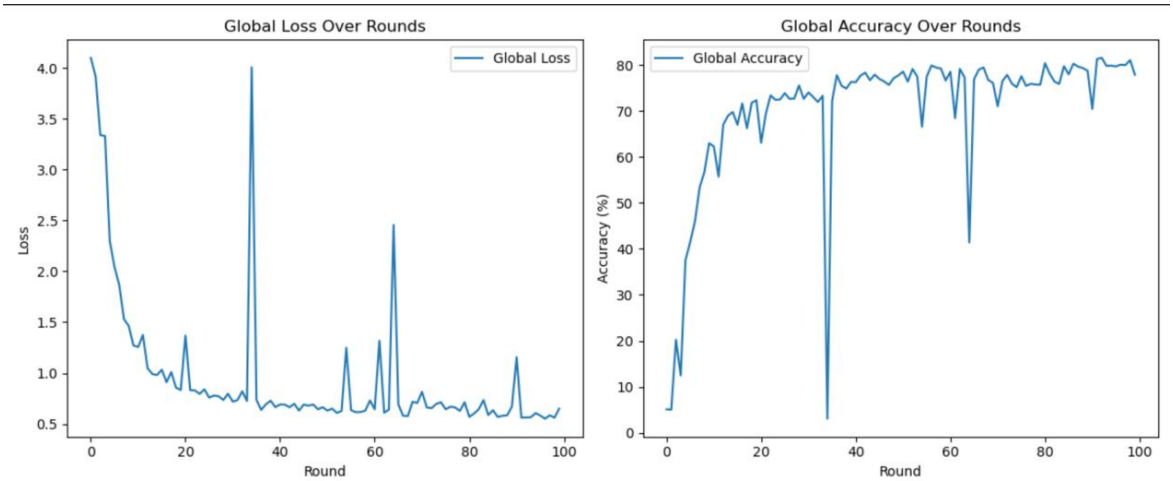


Fig 8. Change the value of α in the Dirichlet distribution from 0.1 to 1 (Picture credit: Original).

The second method is using uniformly partitioned data instead of Dirichlet data partitioning and reusing spatial attention mechanism resulted in significant improvement.

According to Fig 9 and Fig 10, the spatial attention mechanism improves the stability of the algorithm but does not bring significant changes in accuracy and loss. The accuracy and loss curve jitter in the image have been significantly reduced, becoming a smooth curve in some round intervals, which means that the stability has been improved compared to before.

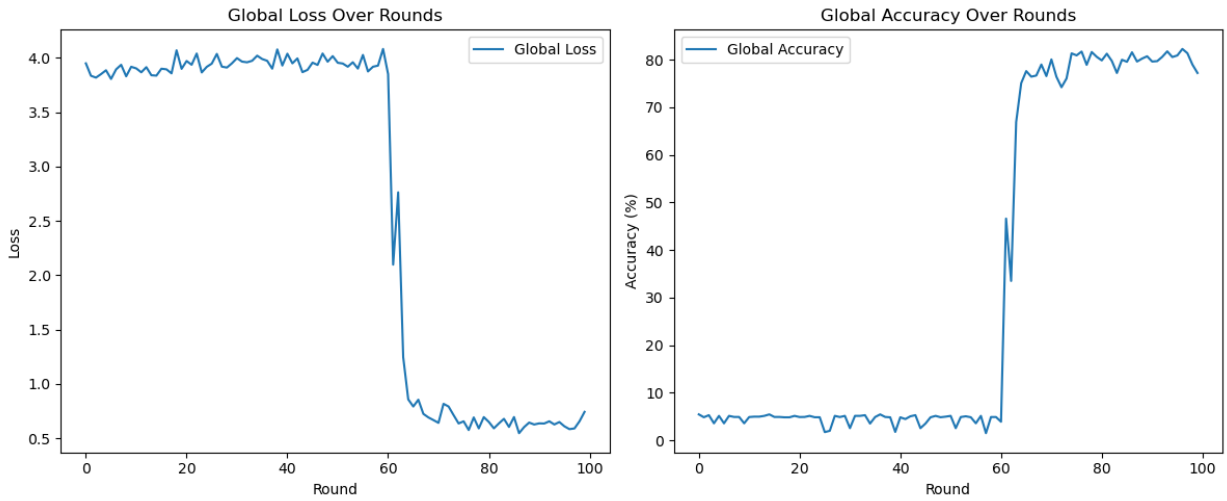


Fig 9. Results of using uniform data partitioning without spatial attention mechanism (Photo credit: Original).

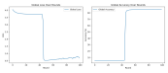


Fig 10. Results of using uniform data partitioning with spatial attention mechanism (Picture credit: Original).

4. Challenges

Regardless of the use of channel attention and spatial attention mechanisms, the accuracy improvement and loss reduction of the Fedprox algorithm based on the FMNIST dataset are limited [5-6]. The high heterogeneity of the dataset has led to the failure of simple convolution implementation of channel attention mechanisms to achieve ideal results, and new solutions need to be sought for global optimization to improve stability. Additionally, spatial attention mechanisms are not suitable for datasets with high heterogeneity. Only when the heterogeneity of the dataset is low can spatial attention mechanism bring better performance [7-8]. However, due to the high heterogeneity of many datasets in reality, and the fact that spatial attention mechanisms can only perform well within a specific range, it is a challenge to add additional content to improve its performance on highly heterogeneous datasets and expand its usage scope [9-10].

5. Conclusion

In this study, the performance of the FedProx algorithm, particularly in federated learning scenarios with the EMNIST dataset, was enhanced by incorporating attention mechanisms, namely channel and spatial attention. The channel attention mechanism, while having a limited impact on

accuracy and loss reduction, significantly contributed to the stability of the FedProx model. This mechanism, especially when paired with a multi-layer perceptron structure, allowed for better performance stability, particularly in scenarios where federated learning networks exhibit non-IID data distributions. The results highlight that, despite its modest impact on direct accuracy improvements, the channel attention mechanism plays a crucial role in mitigating instability during training. This is particularly beneficial in environments where training involves diverse and heterogeneous data across multiple clients, allowing the model to maintain consistency and minimize fluctuations in performance over time.

On the other hand, the spatial attention mechanism encountered considerable challenges when applied to highly heterogeneous datasets. In cases with a higher degree of data distribution skewness, such as those simulated using the Dirichlet distribution, the spatial attention mechanism failed to yield improvements in either accuracy or loss. Instead, it introduced instability, as evidenced by poor performance during training rounds, even with the application of different partitioning strategies like uniform data partitioning. The findings emphasize that spatial attention mechanisms are highly sensitive to the data heterogeneity present in federated learning environments. For spatial attention to be effective, the dataset must exhibit lower heterogeneity, which is often not the case in practical federated learning applications. This underscores the need for further investigation into methods for reducing data heterogeneity or developing more robust attention mechanisms that can perform well across a broader range of datasets. Ultimately, while attention mechanisms can optimize FedProx, their effectiveness is contingent on the specific characteristics of the dataset, and more targeted solutions are required for improving stability and performance in real-world federated learning applications.

References

[1] Li, T., et al.: 'Federated Optimization in Heterogeneous Networks'. Proceedings of Machine Learning and Systems (MLSys), 2020, 2, pp. 429–450

[2] Niu, Z., Zhong, G., Yu, H., et al.: 'A review on the attention mechanism of deep learning', Neurocomputing, 2021, 452, pp. 48–62

[3] Xu, L., et al.: 'A Novel Global Spatial Attention Mechanism in Convolutional Neural Network for Medical Image Classification'. arXiv preprint, 2020, available at: https://arxiv.org/abs/2007.15185 (accessed 31 July 2020)

[4] Murtagh, F.: 'Multilayer perceptrons for classification and regression', Neurocomputing, 1991, 2, (5–6), pp. 183–197

[5] Jiang, L., et al.: 'PFedAtt: Attention-based Personalized Federated Learning on Heterogeneous Clients'. Proceedings of the AAAI Conference on Artificial Intelligence, 2021, 35, pp. 7865–7873 .

[6] Wang, J., et al.: 'Knowledge Generation and Distillation for Road Segmentation in Intelligent Transportation Systems'. IEEE Transactions on Intelligent Transportation Systems, 2025.

[7] Zhu, X., et al.: 'RMER-DT: Robust Multimodal Emotion Recognition in Conversational Contexts Based on Diffusion and Transformers'. Information Fusion, 2025, 103268.

[8] Wang, R., et al.: 'RAFT: Robust Adversarial Fusion Transformer for Multimodal Sentiment Analysis'. Array, 2025, 100445.

[9] Chen, Z., et al.: 'FLAMe: Federated Learning with Attention Mechanism using Spatio-Temporal Features'. IEEE Sensors Journal, 2025, 25, (1), pp. 123–135 .

[10] Li, Y., et al.: 'Accelerating Federated Learning Through Attention on Local Model Updates'. Proc. 36th Conference on Neural Information Processing Systems (NeurIPS), Virtual, 2022, pp. 1–15.