Impact of Modern Transformer Architectures on Federated Learning for Remote Sensing

Students: Sebastian Völkers, Kenneth Weitzel, Felix Zailskas

Supervisor: Baris Buyuktas

TECHNISCHE UNIVERSITÄT BERLIN

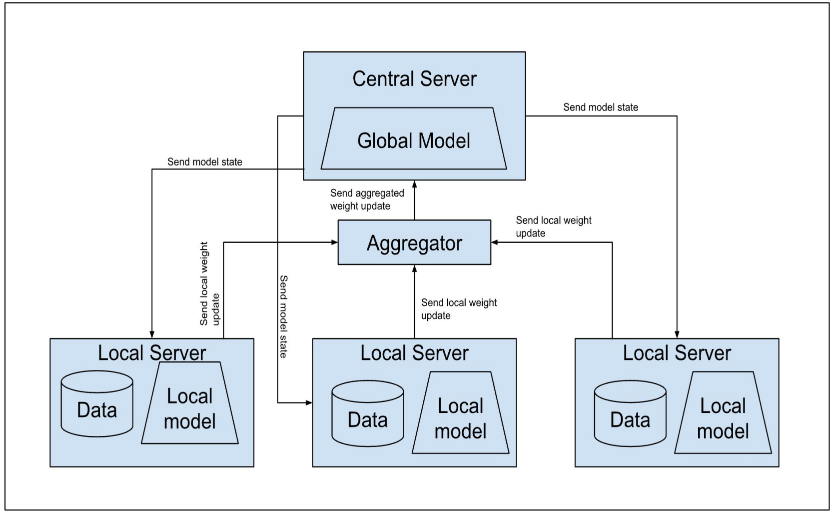
Table of Contents

Motivation

Motivation: Federated Learning

- Remote sensing training datasets can be stored under decentralized databases (i.e., clients) and can be unshared
  - Privacy, commercial interest, legal regulations [1]

Problem: How to train a deep neural network without having direct access to the training data?



Communication round cycle in federated learning.

- Training data in different clients might be not independent and identically distributed (non-IID) due to:
  - Label distribution skew
  - Quantity skew
  - Concept drift
- The presence of non-IID data in federated learning (FL) can reduce the overall performance as it affects the convergence of the global model
- Transformers can address the limitations of training data heterogeneity [2]

[1] Büyüktaş, B., Sümbül, G., & Demir, B. (2023). Federated learning across decentralized and unshared archives for remote sensing image classification. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2311.06141
[2] Qu, L., Zhou, Y., Liang, P. P., Xia, Y., Wang, F., Adeli, E., Li, F., & Rubin, D. L. (2022). Rethinking architecture design for tackling data heterogeneity in federated learning. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr52688.2022.00982

Federated Learning Algorithms

FedAvg [3]

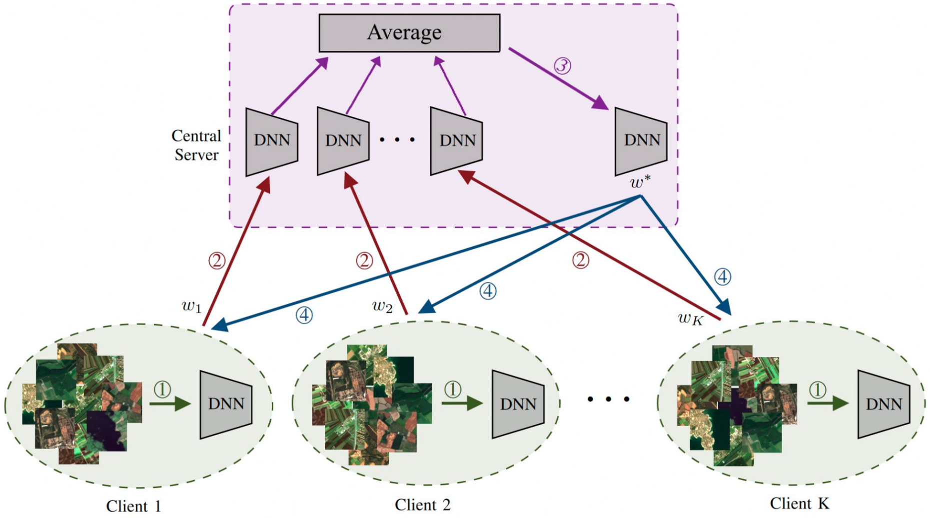
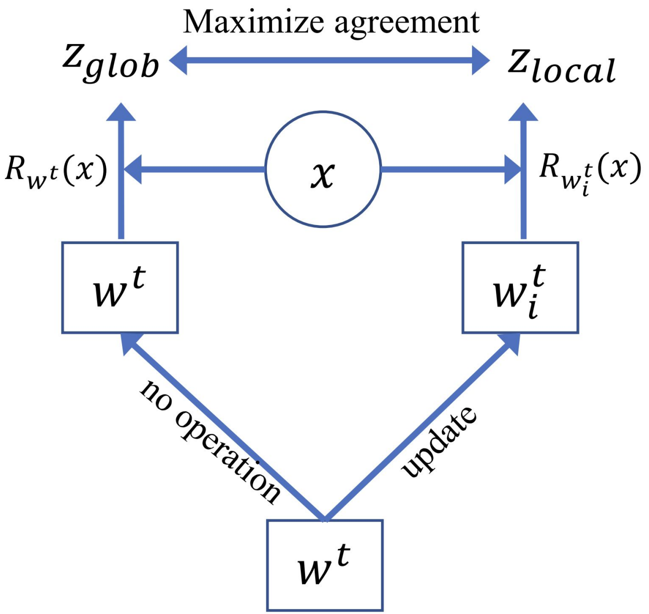


Illustration of FedAvg, Figure adapted from [1]

- Most simple aggregation technique
- Average all client model updates
- Use average as global model update
- Does not address data heterogeneity at all

[1] Büyüktaş, B., Sümbül, G., & Demir, B. (2023). Federated learning across decentralized and unshared archives for remote sensing image classification. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2311.06141
[3] McMahan, H. B., Moore, E. B., Ramage, D., Hampson, S., & Arcas, B. a. Y. (2017). Communication-Efficient Learning of Deep Networks from Decentralized Data. International Conference on Artificial Intelligence and Statistics, 1273–1282. http://proceedings.mlr.press/v54/mcmahan17a/mcmahan17a.pdf

MOON [4]



Feature similarity adjustment in MOON [4]

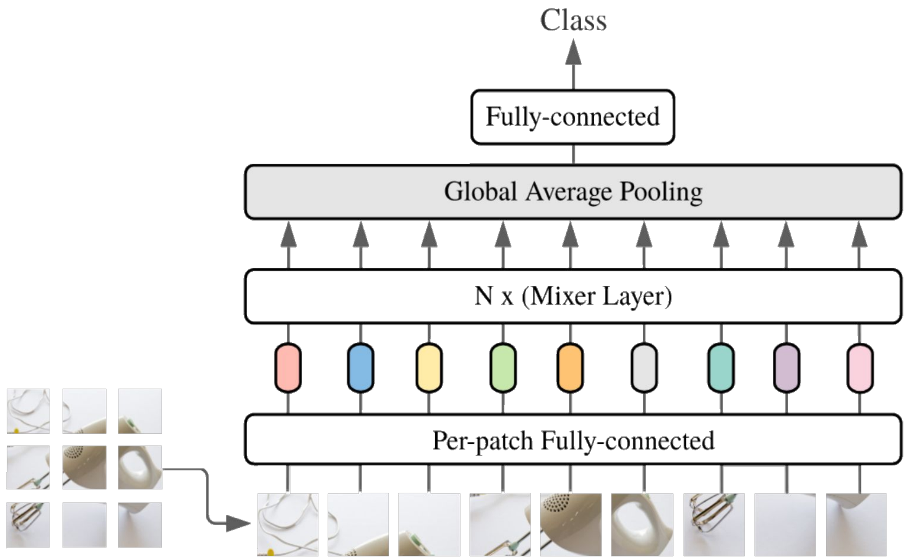
- Local training focused algorithm
- Adds proximal term to local objective function
- Addresses the training data heterogeneity
- Increase similarity between features of global and local model
- Reduce similarity between features of current and previous model

[4] Li, Q., He, B., & Song, D. (2021). Model-Contrastive Federated Learning. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10708–10717. https://doi.org/10.1109/cvpr46437.2021.01057

The MLP-Mixer Architecture [5]

[5] Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lučić, M., & Dosovitskiy, A. (2021). MLP-Mixer: an all-MLP architecture for vision. arXiv (Cornell University). https://arxiv.org/pdf/2105.01601.pdf

MLP-Mixer Architecture

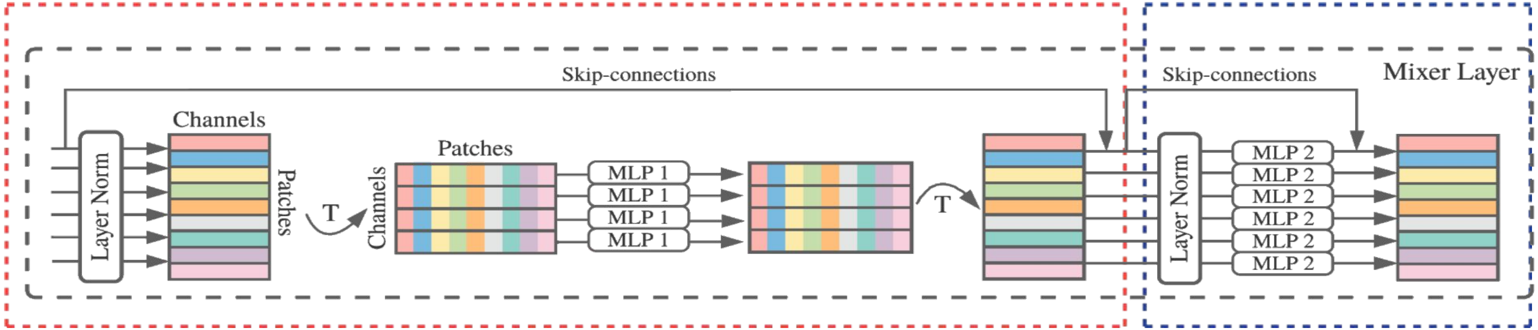


Model architecture of MLP-Mixer [5]

- The aim is to create a computer vision model without convolution and self-attention layers
- MLP-Mixer Only uses multilayer perceptrons repeatedly applied across spatial locations and feature channels

[5] Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lučić, M., & Dosovitskiy, A. (2021). MLP-Mixer: an all-MLP architecture for vision. arXiv (Cornell University). https://arxiv.org/pdf/2105.01601.pdf

MLP-Mixer Architecture



Mixer Layer [5]

- Two parts consists of Token-Mixer and Channel-Mixer
- Token-Mixer: cross-location mixing
- Channel-Mixer: per-location mixing
- 2 MLPs that each use same parameters across inputs
- Promised good computation-accuracy trade-off

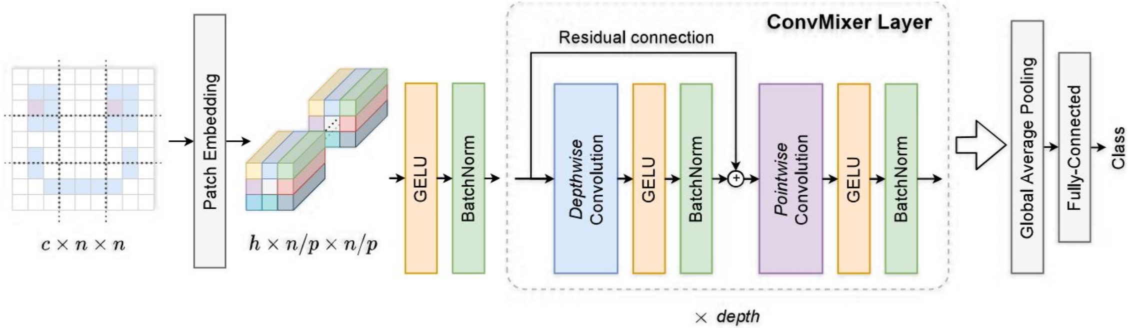
[5] Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lučić, M., & Dosovitskiy, A. (2021). MLP-Mixer: an all-MLP architecture for vision. arXiv (Cornell University). https://arxiv.org/pdf/2105.01601.pdf

The ConvMixer Architecture [6]

[6] Trockman, A., & Kolter, J. Z. (2022). Patches are all you need? arXiv (Cornell University). https://doi.org/10.48550/arxiv.2201.09792

ConvMixer Architecture

- Apply linear patch embedding, as in vision transformers [7]
- Built upon MLP-Mixer, with separate spatial and channel-wise mixing, while replacing MLPs by convolutional layers [5]
- Simple model architecture that provides large receptive fields for CNN with prior patch embedding and large kernel sizes
- Provides efficiency in parameters used vs. performance compared to ResNet-152 or DeiT-B
- Parameter efficiency interesting for FL



Model architecture of ConvMixer [6]

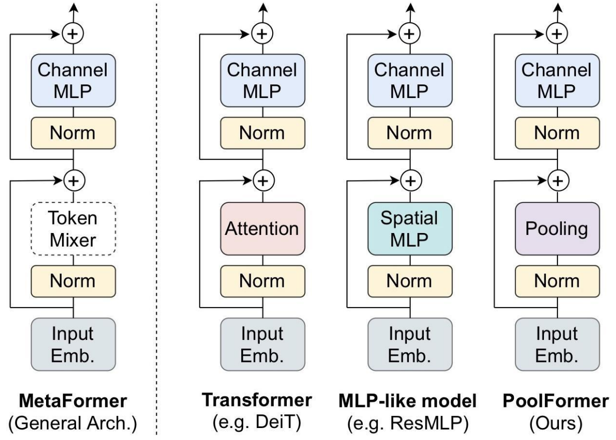
[5] Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lučić, M., & Dosovitskiy, A. (2021). MLP-Mixer: an all-MLP architecture for vision. arXiv (Cornell University). https://arxiv.org/pdf/2105.01601.pdf
[6] Trockman, A., & Kolter, J. Z. (2022). Patches are all you need? arXiv (Cornell University). https://doi.org/10.48550/arxiv.2201.09792
[7] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv (Cornell University). https://openreview.net/pdf?id=YicbFdNTTy

TECHNISCHE UNIVERSITÄT BERLIN

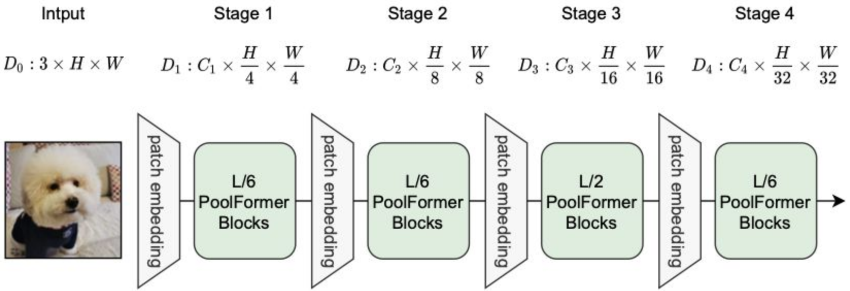
The MetaFormer Architecture [8]

[8] Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., & Yan, S. (2022). MetaFormer is Actually What You Need for Vision. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr52688.2022.01055

MetaFormer Architecture



MetaFormer block compared to other Transformer blocks [8]



PoolFormer architecture [8]

- Success of Transformer architecture attributed to the Attention token mixer
- The MetaFormer block has the same structure as the Transformer block but with variable token mixer
- The PoolFormer block uses a simple (untrainable) average pooling operation as a token mixer
- The full architecture uses 4 stages with L/6, L/6, L/2, L/6 PoolFormer blocks per stage
- Patch embedding is applied before each stage
- PoolFormer consistently outperformed other CV-models on ImageNet1K

[8] Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., & Yan, S. (2022). MetaFormer is Actually What You Need for Vision. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr52688.2022.01055

Experimental Setup

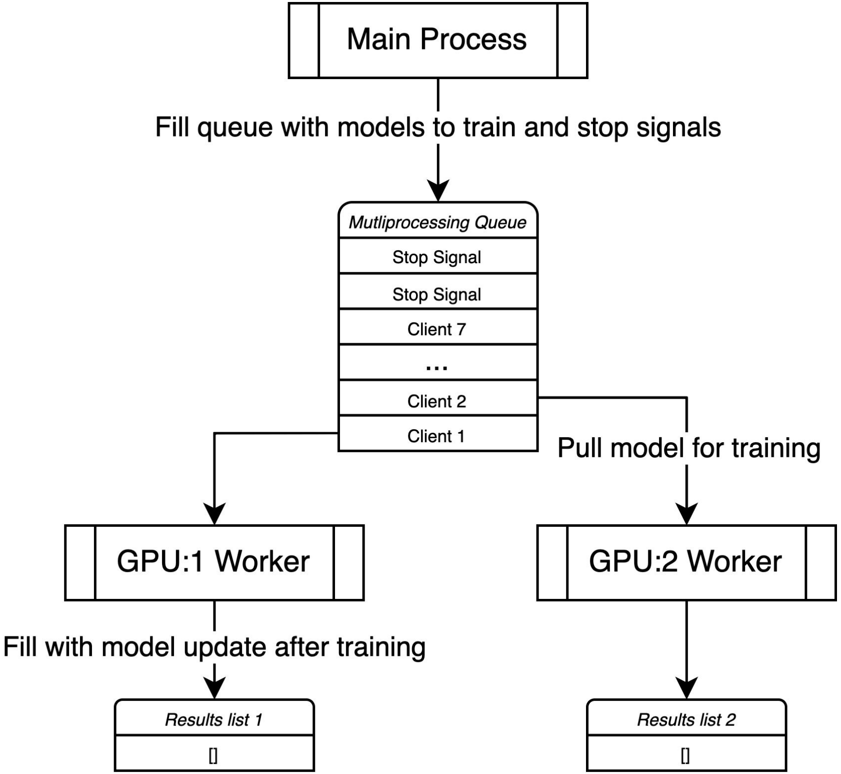
Experiment Setup

- We used the BigEarthNet [9] dataset for multi-label classification
  - Used Countries: Austria, Belgium, Finland, Ireland, Lithuania, Serbia, Switzerland which resulted in seven local FL-clients in each training run
  - Images had 10 channels and 19 class labels
- We used three data decentralization scenarios (DS) to get different levels of non-IID
  - 1. Split all data randomly across all clients (low non-IID)
  - 2. Each client gets data from only one country (medium non-IID)
  - 3. Each client gets data from only one country in one season (high non-IID) (Only used for the sensitivity analysis)
- We compared four models
  - ResNet50, ConvMixer, MLP-Mixer, PoolFormer
- We compared two aggregation algorithms
  - FedAvg, MOON

[9] Sümbül, G., Charfuelàn, M., Demir, B., & Markl, V. (2019). BigEarthNet: A Large-Scale Benchmark Archive for Remote Sensing Image Understanding. IEEE International Geoscience and Remote Sensing Symposium, 5901–5904. https://doi.org/10.1109/igarss.2019.8900532

GPU Parallelization

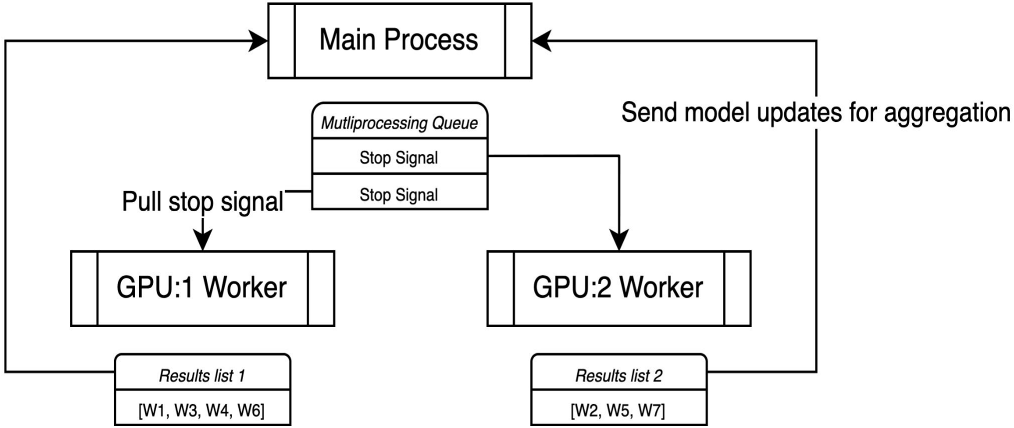
GPU Parallelization



Scheduling of local client training

1. Main process creates process persistent queue and fills it with the models to train
2. Main process adds one stop signal per available GPU to the queue
3. Main process creates one GPU worker process per available GPU
4. GPU workers pull from the model queue and train the next model in parallel
5. Training results are stored in lists within the worker processes

GPU Parallelization



Worker processes join and return results

6. Once all models are trained the workers pull a stop signal
7. The workers return the results lists to the main process and terminate
8. Main process aggregates the model updates with the specified aggregation algorithm

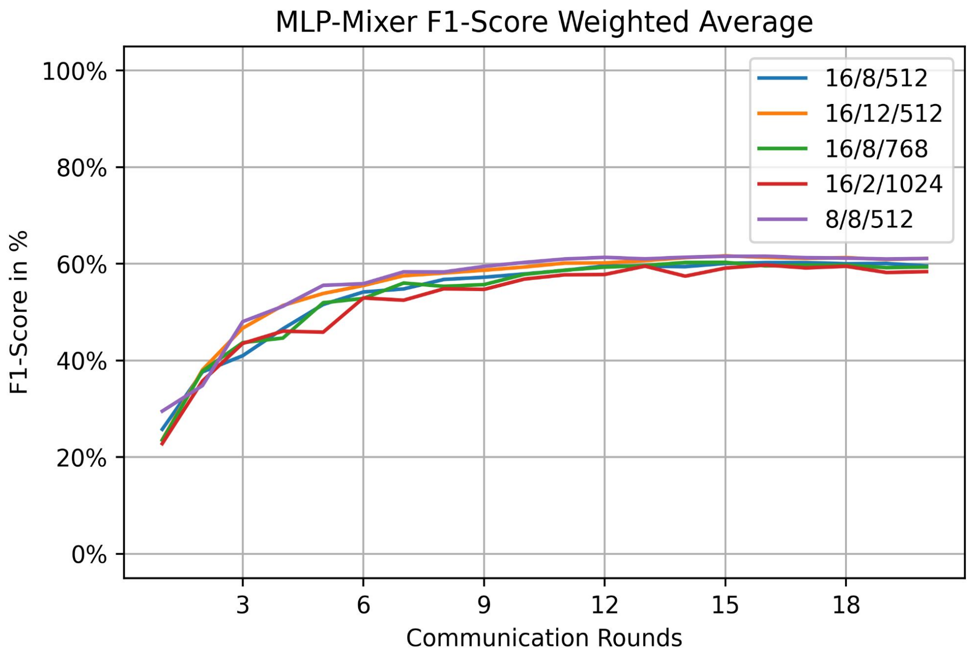
- ➢ When training in parallel the GPU processes slow down
- ➢ For two GPUs we observed a slowdown of ~1s/batch to ~1.5s/batch resulting in no improved training times
- ➢ Hence, we did not use this for training

Experimental Results

Sensitivity Analysis - MLP Mixer

- Tested configurations with (patch size/#blocks/hidden dim.)
  - 16/8/512
  - 16/12/512
  - 16/8/768
  - 16/2/1024
  - 8/8/512
- Model performance similar but patch size 8 performed best and doesn't affect number of parameters much

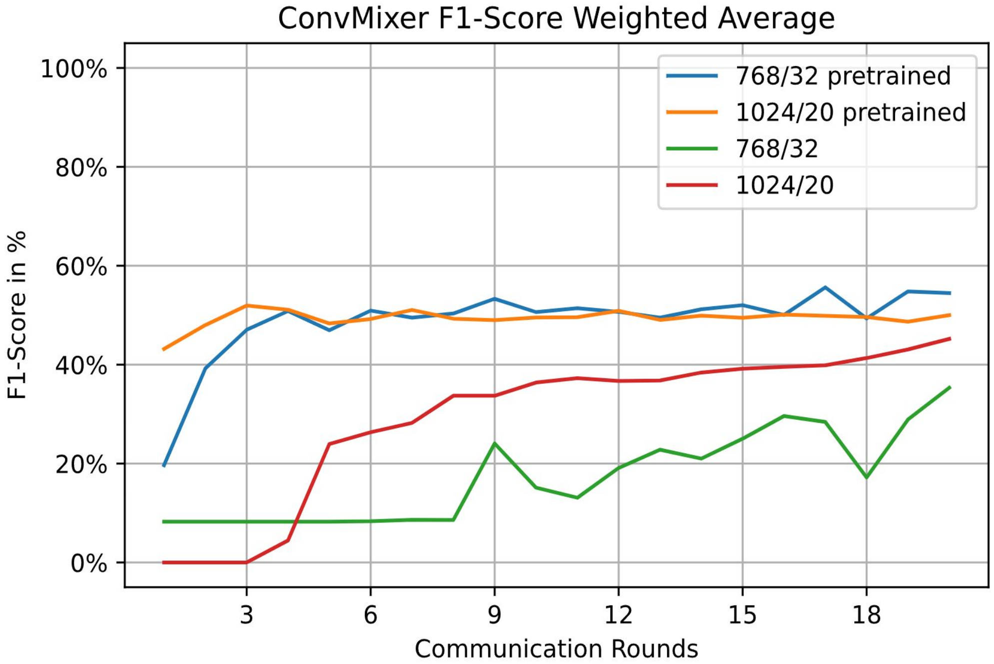
Choice: MLP-Mixer 8/8/512



Sensitivity Analysis - ConvMixer

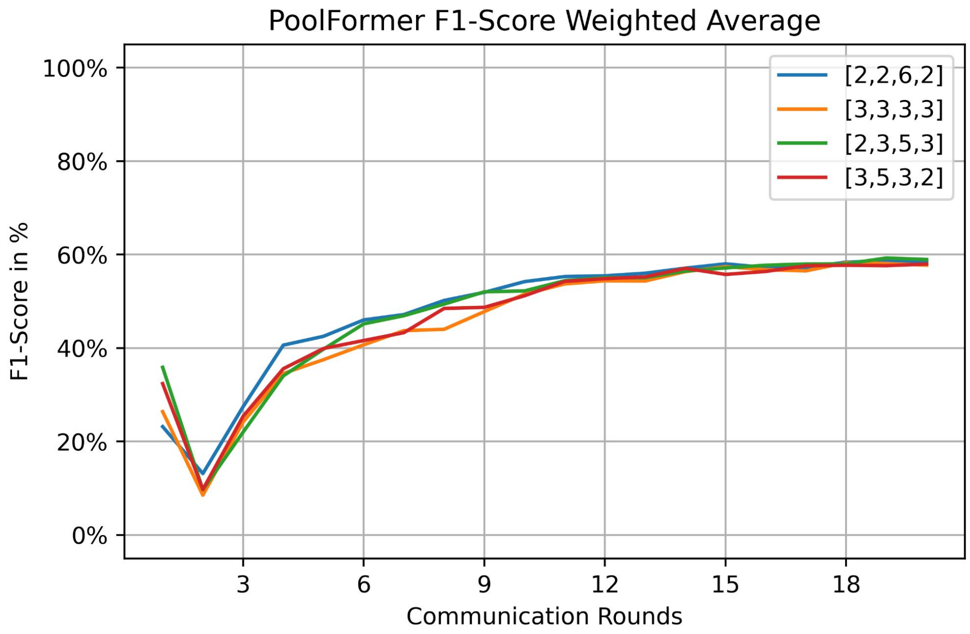
- Tested ConvMixer Configurations (hidden dimension/depth)
  - 768/32
  - 768/32 pretrained
  - 1024/20
  - 1024/20 pretrained
- Pretrained models preformed substantially better

Choice: ConvMixer 768/32 pretrained



Sensitivity Analysis - PoolFormer

- Tested different distributions of PoolFormer blocks across stages (12 available)
  - 2, 2, 6, 2
  - 3, 3, 3, 3
  - 2, 3, 5, 3
  - 3, 5, 3, 2
- Models perform almost the same
- Numerically the paper suggested is best (2, 2, 6, 2)



Choice: PoolFormer-S12 with distribution 2, 2, 6, 2

Results on BigEarthNet [9]

Algorithm	Architecture	# Parameters	Training Time (s/communication round)	DS1 (F1-score Weighted avg)	DS2 (F1-score Weighted avg)
FedAvg	ResNet-50	23.60 M	767	75.24	47.32
	MLP-Mixer	20.65 M	1447	75.81	62.77
	ConvMixer	20.62 M	2432	72.47	53.66
	PoolFormer	11.24 M	1231	74.09	59.85
MOON	ResNet-50	23.60 M	972	72.82	54.83
	MLP-Mixer	20.65 M	1738	75.12	59.48
	ConvMixer	20.62 M	2683	73.28	58.76
	PoolFormer	11.24 M	1447	74.47	60.26

[9] Sümbül, G., Charfuelàn, M., Demir, B., & Markl, V. (2019). BigEarthNet: A Large-Scale Benchmark Archive for Remote Sensing Image Understanding. IEEE International Geoscience and Remote Sensing Symposium, 5901–5904. https://doi.org/10.1109/igarss.2019.8900532

Conclusions

Guide to use Federated Learning with Remote Sensing Data

	Data Distribution	
#Parameters/Training time	**IID**	**Non-IID**
Relevant	PoolFormer, ResNet	PoolFormer
Less relevant	ResNet, MLP-Mixer	MLP-Mixer, PoolFormer
Aggregation Strategy	FedAvg	MOON
Model Category	Classical CNN	Transformer

Transformer comparison:

- MLP-Mixer: good performing architecture for both IID and non-IID
- PoolFormer: natural choice if a small model size is needed, very good performance
- ConvMixer: generally could not keep up with the other two architectures

Future Research

Future Research Venues

Algorithms & Models:

- Test more architectures and models and how they might interact with different FL algorithms
- Pre-training for other Transformers on non-RS data for RS tasks
- Further insights into the effect of pre-training on ConvMixer

GPU Parallelization:

- Test larger models or larger datasets to see if the issue was a scheduling overhead
- Test if dedicated GPU memory improves results as they accessed the same disk location in parallel

CV Tasks:

- Test if results are reproducible for other CV tasks in RS
    - Land-cover map generation, Image retrieval systems

Thank you for your attention!

Sources

[1] Büyüktaş, B., Sümbül, G., & Demir, B. (2023). Federated learning across decentralized and unshared archives for remote sensing image classification. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2311.06141

[2] Qu, L., Zhou, Y., Liang, P. P., Xia, Y., Wang, F., Adeli, E., Li, F., & Rubin, D. L. (2022). Rethinking architecture design for tackling data heterogeneity in federated learning. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr52688.2022.00982

[3] McMahan, H. B., Moore, E. B., Ramage, D., Hampson, S., & Arcas, B. a. Y. (2017). Communication-Efficient Learning of Deep Networks from Decentralized Data. International Conference on Artificial Intelligence and Statistics, 1273–1282. http://proceedings.mlr.press/v54/mcmahan17a/mcmahan17a.pdf

[4] Li, Q., He, B., & Song, D. (2021). Model-Contrastive Federated Learning. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10708–10717. https://doi.org/10.1109/cvpr46437.2021.01057

[5] Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lučić, M., & Dosovitskiy, A. (2021). MLP-Mixer: an all-MLP architecture for vision. arXiv (Cornell University). https://arxiv.org/pdf/2105.01601.pdf

[6] Trockman, A., & Kolter, J. Z. (2022). Patches are all you need? arXiv (Cornell University). https://doi.org/10.48550/arxiv.2201.09792

[7] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv (Cornell University). https://openreview.net/pdf?id=YicbFdNTTy

[8] Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., & Yan, S. (2022). MetaFormer is Actually What You Need for Vision. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr52688.2022.01055

[9] Sümbül, G., Charfuelàn, M., Demir, B., & Markl, V. (2019). BigEarthNet: A Large-Scale Benchmark Archive for Remote Sensing Image Understanding. IEEE International Geoscience and Remote Sensing Symposium, 5901–5904. https://doi.org/10.1109/igarss.2019.8900532