Federated Learning and Model Editing : connections and open problems

Alessandro Maini s | Alessio Dematti s | Mario Capobianco s | Alessandro Conforti s346511

Abstract

In this report a Federated Learning scenario is studied training a model to recognize images in the CIFAR-100 dataset.

using a the DINO ViT-S/16 pre-trained model as backbone, with an added classification head. We train a centralized version of the model on the whole dataset, first by freezing the backbone and only training the classification head, and then by applying model editing techniques on the whole model. Using this centralized models as references, we train the model in a Federated Learning scenario with a

1. Introduction

Federated Learning is an approach to Deep Learning that aims to train a model using different datasets that are held by different entities, called clients, without making them store their data in centralized dataset.

This approach is promising for application where sharing data is impractical or undesirable, for example for privacy and data protection concerns (e.g healthcare)

2. Centralized baseline

The first thing done is training a centralized baseline on the whole CIFAR-100 dataset, to have a reference to use in evaluating the federated model performance. The model used is obtained by taking the pre-trained backbone DINO ViT/16 and adding a classification head. Only the classification head's parameters are trained, while the backbone's parameters are frozen

2.1. Data preprocessing

Before starting the training, some preprocessing to the data instances of the training set is applied . The first processing is data normalization, where data is normalized so that it has mean equal to 0 and standard deviation equal to 1 on all the three channels, to make the training process easier. Then is applied some data augmentation using random crops and random horizontal flips. This is useful to avoid overfitting, preventing the model from learning things that aren't useful and allowing it to converge faster.

2.2. Hyperparameters search

The hyperparameters of the model are the learning rate, the momentum and the weight decay. To search for the hyperparameters, a validation split of the dataset is produced, with approximately 10% of the dataset being part of it. The scheduler used for the learning rate is the cosine annealing scheduler, as suggested, running the training for 10 rounds, stopping early if the validation accuracy stopped improving significantly for more than five consecutive rounds.

The search for hyperparameters is carried out using a grid seaarch where all the combination of hyperparameters are tested and compared, due to the lack of automated procedures like Gradient Descent that can be exploited

The combinations of hyperparameters tested are:

- Learing rates: [0.1, 0.001, 0.005]
- Momentums : [0.8, 0.9, 0.95]
- Weight decay : [0.001, 0]

The results are plotted in the following figure, where it is shown that the best combination is

2.3. Hyperparameters experimentation

After finding the best combination of hyperparameters, a little experimentation is done changing the learning rate scheduler

2.4. Training results

After selecting the best hyperparameters, the model is trained on the full training set and the results are gathered.

3. Centralized model editing

Model editing is a technique in machine learning where developers try to modify some pre-trained model without retraining it from scratch. The goal of model-editing is to improve the performance of the model on some task that it wasn't originally trained on, with minimal distruption on

its original output. This approach has something in common with Federated Learning: in both cases the goal is to merge models trained on different datasets so that the resulting model is as good as possible in making prediction from all of them. Here, model editing is performed by selecting a subset of parameters that have the least impact on the model output on the original task, so to minimize distruption in the model performance. The parameter's impact on the output is measured using the Fisher score

3.1. Experimentation

The first part of the experimentation on centralized model editing is a warm-up training of the classification head, using the same hyperparameters of the centralized baseline. After that, the a parameters mask is computed on the backbone model. The mask has one corresponding entry for each parameter and is designed so that the parameters that have to be updated have the corresponding entry set to 1, while the others have it set to 0

4. Federated Learning

For the Federated Learning scenario, the algorithm used is Federated Averaging [inserire reference al paper]. The number of clients is fixed to 100, and the fraction of clients that perform computation on each round to 0.1.

At first, each client get a iid shard of the dataset and performs 4 local step before each syncronization round