Federated and Incremental Learning
Student project handout

QIAI Lab

July 2020

1 Requirements

- Python 3.6
- Knowledge in Deep-learning
- Affinity with PyTorch is a plus

2 Scope

Machine learning methods offer particularly powerful technologies to infer structural information from given digital data; still, the majority of current applications restrict to the classical batch setting: data are given prior to training, hence meta-parameter optimisation and model selection can be based on the full data set, and training can rely on the assumption that the data and its underlying structure are static. Incremental learning, in contrast, refers to the situation of continuous model adaptation based on a constantly arriving data stream.

We want to combine federal learning with incremental learning. Institutions may involve several round of federal learning with incremental added dataset. But this is a very novel directions, we may need to design the experiments yourself, applying some techniques for solving the catastrophic forgetting problems.

2.1 Retina dataset

The Diabetic Retinopathy 35,126 training samples , 10,906 validation samples and 42,670 test samples [1]. A clinician has rated the presence of diabetic retinopathy in each image on a scale of 0 to 4, according to the following scale: 0 - No DR, 1 - Mild, 2 - Moderate, 3 - Severe and 4 - Proliferative DR.

[1] https://www.kaggle.com/c/diabetic-retinopathy-detection/data

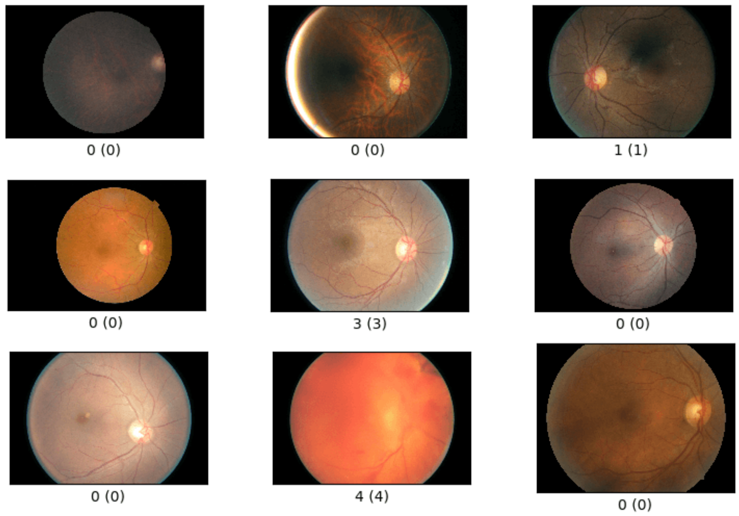

Figure 1: Samples of the retina dataset

3 Preliminary results

By Liangqiong Qu

Experimental setup 1 Three rounds of federal learning (homogenous distributions across sites). Simulated 4 sites.

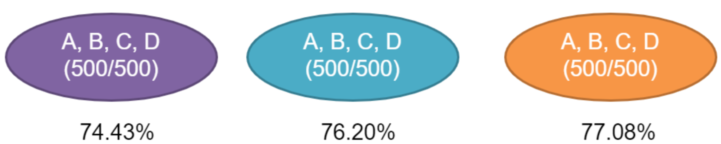

Figure 2: Results of the three rounds

Note: 77.22% (baseline results with union of all dataset), so 77.06 is not bad.

Experimental setup 2 Two rounds have variation data distributions (one: 80% positive, 20% negative, two: 20% positive, 80 negative).

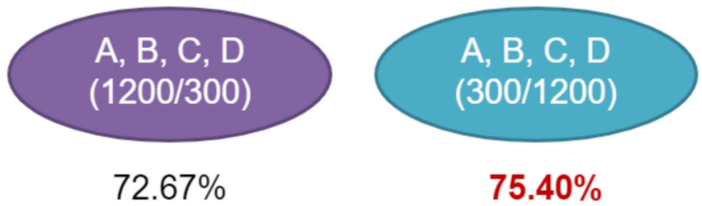

Figure 3: Results of the two rounds

Note: decrease from 77.06% to 75.40%

Experimental setup 3 Introducing distillation loss between previous round model and current round model

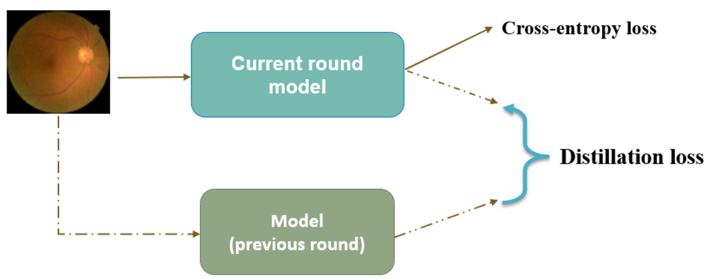

Figure 4: Distillation Loss

Given x trained for a specific classification task, the final layer of the network is typically a softmax in the form:

$$y_i(\mathbf{x}|t) = \frac{e^{\frac{z_i(\mathbf{x})}{t}}}{\sum_j e^{\frac{z_j(\mathbf{x})}{t}}} \tag{1}$$

Knowledge distillation consists of training a network on a different set (different than the dataset used to train the model previously) using as loss function the softmax output of the current distilled network $\mathbf{y}(\mathbf{x}|t)$ and the previous round network output $\hat{\mathbf{y}}(\mathbf{x}|t)$ using a certain value of softmax temperature t:

$$\mathcal{L}_{KD}(\mathbf{x}|t) = \sum_i \hat{y}_i(\mathbf{x}|t) \log y_i(\mathbf{x}|t) \tag{2}$$

If ground truth is available for the transfer set, the process can be strengthened by adding to the loss the cross entropy between the output of the distilled model (computed with $t = 1$) and the known label $\bar{y}$:

$$\mathcal{L}_{KD}(\mathbf{x}|t) = -t^2 \sum_i \hat{y}_i(\mathbf{x}|t) \log y_i(\mathbf{x}|t) \tag{3}$$

Finally, we can define the final loss as a weighted combination of the distillation loss and a typical cross entropy loss $\mathcal{L}_{CE}$:

$$\mathcal{L} = \alpha \mathcal{L}_{CE} + (1-\alpha)\mathcal{L}_{KD} \tag{4}$$

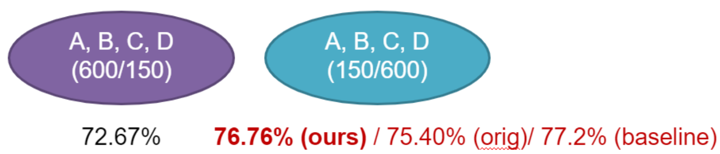

Figure 5: Distillation Loss

4 Project

The first step is to take over the federated-learning code from *Liangqiong Qu* (`https://github.com/jbdel/federated_learning`) and change it to make it work on a single machine (i.e. simulate the training procedure for n sites but on a single machine). Secondly, download and preprocess the retina dataset and replicate the results shown in this handout.

So far, we only played with class imbalances on a single dataset. The goal of this project is to:

Introduce new classes Using the ophthalmic database of 5,000 patients with age, color fundus photographs from left and right eyes and doctors' diagnostic keywords from doctors (in short, ODIR-5K [2]).

They classify patient into eight labels including normal (N), diabetes (D), glaucoma (G), cataract (C), AMD (A), hypertension (H), myopia (M) and other diseases/abnormalities (O) based on both eye images and additionally patient age.

[2] `https://odir2019.grand-challenge.org/dataset/`

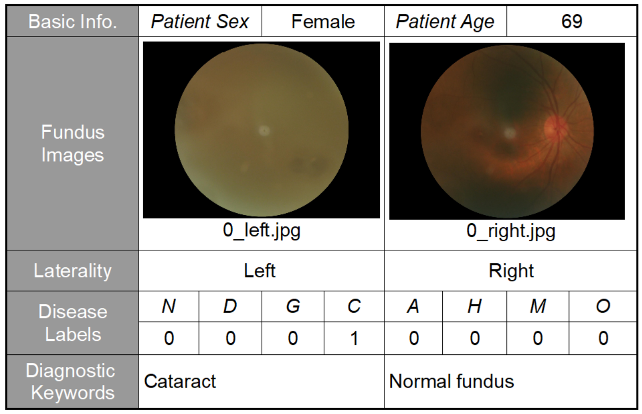

Figure 6: ODIR dataset

Use out of domain images What if we use photos from another dataset (another distribution). The REFUGE challenge [3] offers 1200 fundus images from 2 types of devices.

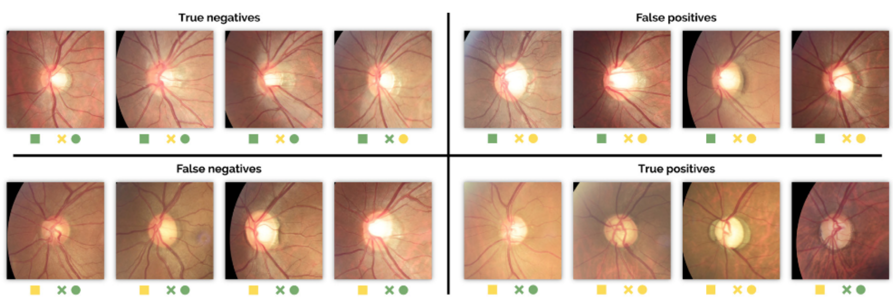

Figure 7: REFUGE dataset

5 Contacts

- Professor Rubin - `dlrubin@stanford.edu`
- Liangqiong Qu - `liangqi@stanford.edu`
- Jean-Benoit Delbrouck - `Jean-Benoit.DELBROUCK@umons.ac.be`

[3] `https://refuge.grand-challenge.org/Home2020/`

6 References

- Incremental Learning of Object Detectors Without Catastrophic Forgetting (ICCV2017)
- Overcoming catastrophic forgetting in neural networks (EWC) (PNAS2017)
- Continual Learning Through Synaptic Intelligence (ICML2017)
- Gradient Episodic Memory for Continual Learning (NIPS2017)
- Continual Learning with Deep Generative Replay (NIPS2017)
- Learning without forgetting (ECCV2016)