A Study of Few-shot Learning on a Federated Setting

Anna Mosolova[1], Elisa Lubrini[1] and Christophe Cerisara[2]

1- Universite de Lorraine - IDMC
Pole Herbert Simon, 13 Rue Michel Ney, 54000 Nancy - France

2- Inria - Dept of Second Author
Address of Second Author's school - France

Abstract. Current restrictions on confidentiality often prevent data from leaving personal devices and being sent to central servers on which central models are trained. The problem of machine learning being hindered by scarcity of data can be tackled using few-shot learning (FSL) techniques. In this paper we explore the application of few-shot learning to a federated dataset, where each node (or device) only holds a limited amount of data. Our experiments are carried out using either an Induction Network, which leverages meta-learning to carry out classification tasks, or a pretrained BART model, by fine-tuning it to each node of the dataset.

1 Introduction

Developing approaches to work with limited amount of data is becoming increasingly important, as legal restrictions make it harder and harder to access private data on personal devices.

Federated learning [1], a technique that has been raising in popularity since it was first introduced by Google in 2016, proposes a solution to the problem, allowing to train models on users' devices so that service providers can benefit from a model that has full access to private data, without the need for them to access such data directly.

Various architectures and methods are possible, both for the training of the single peripheral models and for merging the models into a single one. In this paper we want to propose the use of few-shot learning, a machine learning method used to generalise to a new task based on prior knowledge of other tasks [2].

According to our knowledge, no experiments have been published so far applying few-shot learning to federated datasets and, given the timely relevance of finding optimal solutions to working with federated datasets, this paper aims to contribute to the field by reporting some of our related experiments.

2 Methodology

Our experiments consist in the study of few-shot learning in a federated setting. Concretely, we use the Amazon Review dataset [3] for sentiment analysis by distributing its data across a number of nodes (representing devices in a federated setting) each of which is then used to train a separate model.

We experiment with two different architectures: during a first experiment, each node will be trained using an Induction Network (IN) with meta-learning [4], and in a second experiment the training will consist in fine-tuning pretrained BART models [5].

The reviews are divided in 23 categories according to the kind of product being reviewed. In order to accommodate the needs and limitations of each architecture, the dataset was distributed across the nodes in two different ways, for the training and test of: (1) the IN, and (2) the BART model.

In each of the two experiments, once the models were trained they were used to instantiate two ensemble learning methods: averaging [6] and stacking [7]. The results were then compared to the corresponding baselines: the IN SoTA for the first experiment, and zero-shot [8] learning with BART models for the second experiment.

2.1 Induction Networks

For the training, the samples were split into three sections, each of the three being assigned the reviews from 6 or 7 categories, as shown in Table 1. The remaining 4 categories were used for training.

Node	Ex.	Categories
1.	23105	apparel, office products, automotive, toys games, computer video games, software
2.	18146	grocery, beauty, magazines, jewelry watches, sports outdoors, cell phones service, baby
3.	54333	outdoor living, video, camera photo, health personal care, gourmet food, music
Test	*120*	*books, dvd, electronics, kitchen housewares*

Table 1: Portions of the dataset assigned to each node in the IN experiments.

We reused the implementation of the Induction Network by Zhongyu Chen[1]. After assigning to each node the relative categories (Table 1), each of the nodes was initialized as a separate model and trained only on the examples from the selected category. In order to combine the models and evaluate the resulting quality, we used averaging and stacking.

The first one consisted in averaging the predicted probabilities from all models in order to obtain the final prediction.

For the stacking model we retrained the second node without the *baby* category, so it has seen the examples only from the first 6 topics. Each model was then used to predict the probability distribution for the *baby* domain. After this, they were used as features for training a meta-learner. As a meta-learner we tried several classical machine learning algorithms (Logistic Regression, Decision Trees and Support Vector Machines) as well as simple multi-layer perceptron. Usage of logistic regression as a meta-learner showed the best result which is reported in the next section.

[1] https://github.com/zhongyuchen/few-shot-text-classification

2.2 BART

Since our BART model is pretrained, it is expected to produce good results with less examples, compared to the IN. For this reason, only the 4 categories previously used for testing (Table 1) were used, each to train a separate node with 30 examples (15 positive and 15 negative), as shown in Table 2.

Node	Ex.	Categories
1.	30	books
2.	30	dvd
3.	30	electronics
4.	30	kitchen housewares
Test	*120*	*books, dvd, electronics, kitchen housewares*

Table 2: Portions of the dataset assigned to each node in the BART experiments.

First, we tried to evaluate the model without any training (Zero-shot learning setting), to use the results as a baseline. Subsequently, we fine-tuned 4 separate BART models using 3 support sets from the test data of the Amazon reviews dataset. We applied several combination methods on the resulting nodes. The first one was standard averaging consisting in averaging the probabilities from 4 models and outputting the results based on the final probabilities. The second one was weighted averaging that reused the previously described strategy but assigning to each node its own coefficient of contribution to the final prediction depending on its performance on the development set[2]. The third method was to apply the Federated Averaging strategy[1] during the training of 4 nodes. This strategy consists in training models on separate nodes, then sending their weights to the server, averaging them and sending them back for a new round of training.

3 Experiments

3.1 Experimental Settings

Dataset The dataset used for the experiments was the Amazon Review dataset from [3][3]. For the BART experiments, each node was trained based on 3 labelling criterion (reviews from n stars up being considered positive, with $n = 2$, $n = 4$, and $n = 5$), with each node being fine-tuned on 30 different examples, 10 per labelling criterion (5 positive and 5 negative).

IN baseline As a baseline, we used the original Induction Network introduced in [4] which was trained on the whole training set (19 domains). We compared its results with the results of the nodes which were the IN trained only on

[2]The development set was constructed from 601 examples from the training set of the *automative* domain.

[3]The dataset is available at https://github.com/Gorov/DiverseFewShot_Amazon.

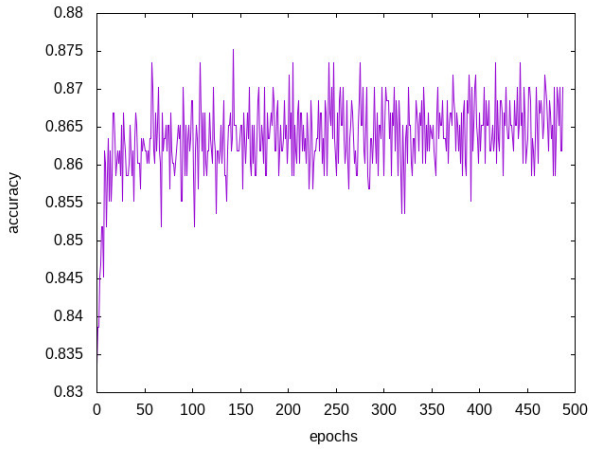

Fig. 1: Evolution of the accuracy during fine-tuning on the development corpus.

subparts of the original dataset. In addition to this, we evaluated their quality in a federated setting by averaging and stacking the predictions of the nodes.

BART general settings BART-MNLI-large[4] from HuggingFace, was used according to different settings: (1) Zero-shot one, (2) training on 12 support sets from the test data, (3) training 4 separate models on the support sets corresponding to one of the test domains and then combining them using several averaging strategies.

weighted averaging For weighted averaging we used the following coefficients: books - 0.1, dvd - 0.3, electronics - 0.4, kitchen housewares - 0.2 that were selected experimentally. The last one is training 4 BART nodes using Federated Averaging.

Hyper-parameters tuning BART's hyper-parameters have been manually tuned with a few trials-and-errors on a development set extracted from the *automotive* subset of the Amazon Reviews corpus. The development set is composed of 30 sentences for fine-tuning, and 600 sentences to compute the sentiment accuracy obtained with the fine-tuned BART-MNLI-large model.

We have tried three fine-tuning strategies: (1) fine-tuning the full BART model, (2) fine-tuning only the first self-attention layer, (3) fine-tuning the inputs, outputs and layer normalization, following [].

The best hyper-parameters were achieved by fine-tuning all BART parameters, with a learning rate of $\lambda = 10^{-4}$ and 100 epochs. Figure 1 shows the evolution of the accuracy on the 600 development sentences during fine-tuning.

[4]https://huggingface.co/facebook/bart-large-mnli (accessed on April, 1, 2021)

We can observe that there does not seem to be any overfitting and that the performances plateau after epoch 100.

3.2 Experimental Results

Table 3 shows the results of all the Induction Network applications including the original one the results of which were reproduced using the implementation by Zhongyu Chen[5].

Model	Mean Accuracy (%)
Node 1	83.5
Node 2	83.3
Node 3	83.1
Averaging	84.6
Stacking	84.6
Reproduced model from [4]	83.9
Result reported in [4] (SoTA)	85.6

Table 3: Results obtained with Induction Network from [4].

Table 4 shows the results obtained with the BART model (with and without fine-tuning). The values between parentheses give the standard deviation of the results after 5 trials.

TODO: report statistical confidence interval at 90% (look for Wald test):

$$\pm 1.96 \sqrt{\frac{p(1-p)}{n = 9000}}$$

Model	Mean accuracy	Books	DVD	Electronics	KH
Zero-Shot Learning	83.0	82.7	80.2	84.3	84.8
Books node	85.1[±0.23]	86.4[±0.2]	83.4[±0.29]	85.0[±0.23]	85.3[±0.33]
Dvd node	85.2[±0.11]	86.8[±0.1]	83.9[±0.34]	85.0[±0.22]	85.5[±0.29]
Electronics node	84.4[±0.39]	84.4[±0.87]	82.1[±0.68]	85.3[±0.23]	86.1[±0.15]
Kitchen housewares node	85.6[±0.12]	86.5[±0.22]	84.8[±0.37]	85.2[±0.22]	86.2[±0.17]
Averaging of 4 nodes	85.9	81.5	84.6	86.2	85.9
Weighted averaging	-	86.8	84.4	86.0	-
Federated Averaging	85.7	86.7	84.4	85.7	86.0

Table 4: Results obtained with BART.

4 Conclusion

In this paper we have explored the combination of few-shot learning (FSL) and federated learning, which is currently a relevant topic given the increasingly rigid

[5]https://github.com/zhongyuchen/few-shot-text-classification

restriction on data protection, since a federated setting simulates the scarcity of data and few-shot learning allows for this scarcity not to be a hindrance.

We explore this possible setting through two main experimental procedures, one by training Induction Networks through meta-learning, and one by fine-tuning BART models, on the task Sentiment Analysis.

We showed that by applying FSL, we could obtain on a federated setting, results that are close to the current SoTA trained on a whole dataset.

References

[1] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

[2] Yaqing Wang, Quanming Yao, James Kwok, and Lionel M. Ni. Generalizing from a few examples: A survey on few-shot learning, 2020.

[3] Mo Yu, Xiaoxiao Guo, Jinfeng Yi, Shiyu Chang, Saloni Potdar, Yu Cheng, Gerald Tesauro, Haoyu Wang, and Bowen Zhou. Diverse few-shot text classification with multiple metrics, 2018.

[4] Ruiying Geng, Binhua Li, Yongbin Li, Xiaodan Zhu, Ping Jian, and Jian Sun. Induction networks for few-shot text classification, 2019.

[5] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, 2019.

[6] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

[7] David H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241 – 259, 1992.

[8] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 951–958, 2009.