Few-Shot Learning Text Classification in Federated Environments

Nehal Muthukumar
Caterpillar India Engineering Solutions Private Limited
Chennai, India
nehalmuthu@gmail.com

Abstract— With increasing privacy restrictions on personal devices, data is often prevented from leaving these devices and reaching the central servers where critical operations are carried out to research and train models better. It is crucial to improve such data and fine-tune the performance of machine learning models in use today. Here, few-shot learning is applicable with federated datasets, where each device holds a limited amount of data only. In this paper, we carry out experiments using an Induction Network (using meta-learning for classification tasks) or a pre-trained BART model by customising it for each node in our dataset. The former helps with a solution devoid of recurrent communication among the node and also slashes computational costs significantly. The latter is bias-free and works efficiently with sparse data. Furthermore, upon reviewing the literature, it was observed that there is a lack of research work entailing the application of few-shot learning algorithms coupled on natural language processing tasks in a federated setting. We attempt to address this gap with this research.

Keywords— Federated Learning, Few-Shot Learning, Deep Learning, Text Classification.

I. Introduction

Developing approaches to work with limited amount of data is becoming increasingly important, as legal measures are nowadays raising the confidentiality of private data on personal devices. Additionally, BERT-like models are becoming a standard approach to carry out NLP tasks and they allow fine-tuning with small amounts of data.

Federated learning [1], a technique that has been raising in popularity since it was first introduced by Google in 2016, proposes 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 or gather all the data in a single dataset. This would result in multiple models being trained, each on its corresponding device, and then joined in a single model that is then sent back to each device for fine-tuning.

However, given the frequency of communication between nodes required during the learning process, a consistent limitation is that local devices are expected to have a relatively high amount of computing power, local memory and a high bandwidth connection [2].

Another downside is the bias that each node can have towards the whole dataset, given the heterogeneity of local datasets. This heterogeneity is also time-bound in that distribution within the dataset can vary with time [2].

Additionally, federated learning is often hindered by the scarcity of data on some devices [2].

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 (FSL), a machine learning method used to generalise to a new task based on prior knowledge of other tasks [3].

To tackle the aforementioned problems affecting the federated learning setting, we experiment with two FSL architectures: the Induction Network and a pretrained BART model. The first offers a solution that does not require frequent communication between nodes and is constituted by a relatively lightweight model, alleviating the bandwidth, memory, and power requirements of the local devices. The second proposes an alternative that works on devices with small amounts of data and is less sensitive to bias compared to other solutions thanks to its relatively stable general weights that are not easily affected by smaller local models.

According to our knowledge, no experiments have been published so far applying FSL architectures on NLP tasks in federated settings, 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 experiments with their relative results.

II. Related Work

There exist a standard procedure to evaluate a Few-shot learning system based on the Amazon Review dataset proposed in [4]. Several models were already tested with this dataset, like [4], [5] and others.

Current state-of-the-art result for the Amazon Review dataset is obtained in the paper by R. Geng et al. [6]. They introduce the Induction Network, a system based on several modules that is capable of predicting a new class after fine-tuning on only several examples from it. This model uses the concept of classes examples and queries which are compared with each class in order to identify the target class. It consists of 3 sub-parts: an Encoder module, an Induction module and a Relation module. An Encoder model is a biRNN with self-attention which is used to encode the class examples and the queries. Then an Induction module, which is implemented as the dynamic routing algorithm with one output capsule, squeezes representation of each class into one vector which is then called a class vector. After this, the Relation module computes the similarity score between query representation and each class representations using the neural tensor layer. The

2021 Smart Technologies, Communication and Robotics (STCR) | 978-1-6654-1806-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/STCR51658.2021.9588833

978-1-6654-1806-5/21/$31.00 ©2021 IEEE

main contribution of this paper is the algorithm that allows us to add quickly a new class into our algorithm without long training and having a lot of examples for it.

One of the recent models that showed a near SoTA performance on a wide range of NLP tasks is the BART model introduced in [7]. This model is an autoencoder built using a standard Transformer architecture to restore the previously corrupted text.

III. Methodology

Our experiments consist in the study of few-shot learning in a federated setting. We distribute the 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) [6], and in a second experiment the training will consist in fine-tuning pretrained BART models [7].

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 (Section IV-A1), and (2) the BART model (Section IV-A2).

In each of the two experiments, once the models were trained, they were used to instantiate two ensemble learning methods: averaging [1] and stacking [8]. The results were then compared to the corresponding baselines: the IN SoTA for the first experiment, and zero-shot [9] learning with BART models for the second experiment.

IV. Experiments

A. Experimental Settings

All the experiments were conducted using the Amazon Review dataset

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 and the remaining 4 categories were used for testing.

As a baseline, we used the original Induction Network introduced in [6] which was trained on the whole training set (19 domains).

For our experiments we reused the implementation of the Induction Network by Zhongyu Chen. Each node with the assigned to it categories was initialized as a separate model and trained only on the examples from the selected category. Then these models were combined.

The first methodology that we used was averaging which consisted in averaging the predicted probabilities from all models in order to obtain the final prediction.

The second approach was stacking for which 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 Section IV-B.

2) BART

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). Each node was trained on only one of the 4 testing categories, and tested on all of them.

As a baseline for the BART model we used a zero-shot learning strategy, namely we evaluated the model on the test set without any prior training.

The BART-MNLI model that we used for our experiments was trained for the natural language inference task, so we transformed each training and test sample into two phrases of the following pattern: a) <source sentence>. This text is positive or b) <source sentence>. This text is negative. By obtaining the probabilities of the second sentence being an entailment, we consider the first sentence to be positive if the probability of the phrase a is higher than of the phrase b, otherwise we say that this sentence is negative.

BART-MNLI-large from HuggingFace, was used according to different settings: (1) training on 12 support sets from the test data, (2) training 4 separate models on the support sets corresponding to one of the test domains and then combining them using several averaging strategies, (3) training 4 separate models with applying the Federated Averaging strategy [1]. For the second setting we used standard and weighted averaging. For the first option we averaged the predicted probabilities from 4 models in order to obtain the final prediction. For the second one, before computing the final prediction, we assign to each node its coefficient of contribution depending on its performance on the development set. After several trials, we used the following coefficients: books - 0.1, dvd - 0.3, electronics - 0.4, kitchen housewares - 0.2.

As for the BART's hyperparameters, they have been manually tuned with a few trials-and-errors on another development set. 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 [10]. The best hyper-parameters were achieved by fine-tuning all BART parameters, with a learning rate of $\lambda = 10{-}4$ and 100 epochs.

B. Experimental Results

Table II shows the baselines for the experiments: the result of the Induction Network reported in the [6], the reproduced result of this algorithm and the Zero-shot learning setting of the BART model. Table I shows the results of all the experiments with the Induction Network. Table III shows the results obtained with the BART model. The values between parentheses give the standard deviation of the results after 5 trials.

TABLE I. IN RESULTS

Model	Accuracy (%)
Node 1	83.5
Node 2	83.3
Node 3	83.1
Averaging	84.6
Stacking	84.6

TABLE II. BASELINES

Model	Accuracy (%)
IN SoTA[a]	85.6
IN Repl[b]	83.9
ZSL	83.0

[a] Published results from [6]

[b] Reproduced model following the SoTA architecture [6]

TABLE III. BART RESULTS

Model	Accuracy (%)
Node 1. (Books)	85.1[±0.23]
Node 2. (Dvd)	85.2[±0.11]
Node 3. (Electronics)	84.4[±0.39]
Node 4. (Kitchen H.)	85.6[±0.12]
Average	85.9
Weighted Avg	85.8
Federated Avg	85.7
Stacking	-

C. Analysis

The results show that FSL techniques can be successful when training on a limited amount of data. The models derived from Induction Networks, both through Stacking and Averaging, perform better than our replica of the SoTA model (although their published results are reported to be higher than the ones we could reproduce). The models derived from BART models also seem promising when compared to the ZSL baseline.

The results show that the BART model, making use of its prior knowledge derived from its pretraining phase, is able to perform better than the baseline with only 30 examples and 100 epochs (as opposed to the IN needing 10.000 epochs and 95584 training examples). This also contributes to its stability, which, as mentioned before, is something other federated learning models commonly lack.

V. CONCLUSION

In this paper we employed two FSL architectures to solve a language classification task in a federated settings. Each architecture was chosen to tackle a specific weakness of federated learning, namely (1) the frequency of communication between nodes required during the learning process, requiring local devices to have a relatively high amount of computing power, local memory and a high bandwidth connection, (2) the bias that each node can have towards the whole dataset, and (3) the need for large amounts of training data.

We proposed a solution to each of the problems 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 and we demonstrated that by applying FSL on a federated setting, results that are close to the current SoTA trained on a whole dataset can be obtained.

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] Edited by: Peter Kairouz and H. Brendan McMahan. Advances and open problems in federated learning. Foundations and Trends in Machine Learning, 14(1):–, 2021.

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

[4] 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.

[5] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A simple neural attentive meta-learner. In International Conference on Learning Representations, 2018.

[6] Ruiying Geng, Binhua Li, Yongbin Li, Xiaodan Zhu, Ping Jian, and Jian Sun. Induction networks for few-shot text classification. In Proceedings of the 2019 EMNLP-IJCNLP, pages 3904–3913, Hong Kong, China, November 2019. Association for Computational Linguistics.

[7] 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.

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

[9] 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.

[10] Kevin Lu, Aditya Grover, Pieter Abbeel, and Igor Mordatch. Pre-trained transformers as universal computation engines. arXiv preprint arXiv:2103.05247, 2021.