Federated Learning for Heterogeneous Devices

Maleeha Masood, Muhammad Nouman Abbasi

Introduction

With the introduction of several data privacy regulations and laws, for instance the GDPR, the topic of secure and private AI has been getting much thought from researchers around the world. However, one main problem that inhibits any advancement in this field is the fact that all traditional machine learning models require large amounts of data, borrowed from its users, for training and validation at the same place and allowed to be accessed by a third party. Thus, to stay in accordance with emerging online privacy laws, we need a way of obtaining user data distributed across the world, without actually 'obtaining' it in order to maintain user privacy. This is where federated learning comes in.

Federated Learning (FL) is a distributed machine learning technique of training a centralized model using decentralized data. The learning involves many clients, that can be mobile or other portable devices, individually training a copy of the current global model that they have received from the central server, thus making sure the data never leaves the client devices. One the individual model has been built, users send only the parameters (e.g. weights) to the central server that then consolidates numerous parameters, that it has received from the numerous individual users, to update its version of the global model. The central server then sends the updated global model to the users for inference.

One of the key features that gives precedence to Federated Learning as compared to the other models is because it addresses the fundamental problems of privacy, ownership, and locality of data [1]. For this reason, it has been labelled as an important innovation that can help machine learning algorithms reach new areas, like medicine, without any privacy concerns like patient confidentiality as per our example. However, the power consuming model training takes place on the end devices, usually mobile phones, and there is an alarming issue of the availability of sufficient excessive resources for training a model.

McMahan and other researchers at Google propose the idea of filtering: sending the global model only to those devices that are plugged, idle and using unmetered Wi-Fi [2]. However, their method simply does not even target low-end mobile devices from the pool because such devices take very long to build models and so are 'dropped'. If one considers a developing country where, while mobile penetration is high, the number of high-end mobile devices is very low, then it is the case that these countries are unable to contribute much to the global model. This adds in bias to the model, while reducing its generality. Another approach of federated learning by Bonawitz [1] provided several solutions for scaling but under the assumption that their models were deployed on devices with at least 2 GB of memory, once again disregarding majority of the mobile devices in the developing regions where 80.2% devices are regraded as low-end (Newzoo 2018 Global Mobile Market Report).

Related Work

Since Federated Learning is a realively new topic, majority of the papers were targeting the performance, efficiency and security of the system. Some of them did try and propose solutions to allow inclusion of heterogeneous devices.

Towards Federated Learning at Scale: System Design suggests a new hierarchal protocol with sub-server aggregators and selectors for the federated system that targets multiple issues regarding the current proposed Federated Learning System. However, the paper does not specifically target model optimization. [a]

Client Selection for Federated Learning with Heterogeneous Resources in Mobile introduces a new protocol, FedCS, that filters out slowly responding parties through a deadline-based approach in an attempt to make the whole system efficient. However, the paper fails to solve our problem: it provides a selection criterion for users that automatically excludes 'slow' devices, like those in developing regions. [b]

Federated Learning: Strategies for Improving Communication Efficiency explored methods to change the model for efficient communications. The paper ignores model performance and efficiency though, which is also equally important in the context of developing regions. [c]

Federated Optimization for Heterogeneous Networks proposes a new averaging algorithm: FedProx that tolerates partial computation. This does benefit our cause and is in line with our problem, however rather than tackling the averaging algorithm, we plan to target the training model. [d]

Adaptive Federated Learning in Resource Constrained Edge Computing Systems proposes an algorithm to determine the frequency of global aggregation so that the available resources are most efficiently used. This paper again targets the improvement of the averaging algorithm while we aim to improve the training model to allow for heterogeneity in the federated learning system. [e]

Proposed Approach

Our primary focus is to optimize machine learning models to run on heterogenous environments in developing countries where mobile device computational power is a constraint. There are several techniques for model optimization, however, our optimization must take into account the distributed nature of federated learning. It must support heterogenous environment and parameter consolidation on a central server.

Possible methods:

- **Binary weights:** Using weights which are constrained to only two possible values can help models to become less computationally expensive. BinaryConnect is an example that enables faster computations for training models on low-power devices. Courbariaux and others achieved obtain near state-of-the-art results on permutation-invariant MNIST, CIFAR-10 and SVHN datasets. [4]
- **Model pruning:** Model pruning involves eliminating unnecessary weight values to achieve a smaller model while minimizing the loss in accuracy of the original model. This reduced version is better optimized for training on low-end devices.
- **Freezing model:** Freezing a model means 'locking' weights in some layers of the neural network to accelerate model training. Interesting progress has been achieved using this technique; Freezeout: Accelerate Training by Progressively Freezing Layers [5] and Fast Deep Learning Training through Intelligently Freezing Layers [6], but little related work has been done in the context of FL.

There are several other techniques for model optimization. However, majority of the current methods, for example the OpenVino Toolkit by Intel and TensorFlow Model Optimization Toolkit, concentrate primarily on *inference* at the edge. We need a solution for optimizing model for *training* at the edge. Although there are a number of solutions for federated learning in heterogenous environment, a lot of them do not target optimizing the training model. Rather, they complicate the averaging algorithm. Thus, we have proposed methods that facilitate the training at edges without effecting the averaging algorithm.

Timeline and Division of Work

Literature Review	22 Feb 2020	Maleeha and Nouman
Collect Data	29 Feb 2020	Maleeha and Nouman
Pinpoint the feasible solution	7 March 2020	
Setting up Programming Environment	14 March 2020	Maleeha and Nouman
Build a prototype	28 March 2020	Maleeha and Nouman
Mid Report	31 March 2020	Maleeha and Nouman
Testing	7 April 2020	Maleeha and Nouman
Evaluate	21 April 2020	Maleeha and Nouman
Final Report	28 April 2020	Maleeha and Nouman

Presentation	5 May 2020	Maleeha and Nouman

GitHub Repository Link

https://github.com/MNoumanAbbasi/CS678-ResearchProject

References:

[1] Bonawitz et. al, Towards Federated Learning at Scale: System Design. *arXiv:1902.01046, 2019.*

[2] Mcmahan et. al, Communication-Efficient Learning of Deep Networks from Decentralized Data. *AISTATS 2017.*

[3] Courbariaux, Matthieu and Bengio, Yoshua and David, Jean-Pierre, BinaryConnect: Training Deep Neural Networks with binary weights during propagations. *Curran Associates, Inc., 2015.*

[4] Brock, A., Lim, T., Ritchie, J. M., & Weston, N. J. (2017). FreezeOut: Accelerate Training by Progressively Freezing Layers. *Paper presented at NIPS 2017 Workshop on Optimization, Long Beach, United States.*

[5] Xiao, Xueli & Mudiyanselage, Thosini & Ji, Chunyan & Hu, Jie & Pan, Yi. (2019). Fast Deep Learning Training through Intelligently Freezing Layers. *1225-1232. 10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00205.*

[a] Bonawitz et. al, Towards Federated Learning at Scale: System Design. *arXiv:1902.01046, 2019.*

[b] Takayuki Nishio and Ryo Yonetani. Client selection for federated learning with heterogeneous resources in mobile edge. *arXiv preprint arXiv:1804.08333*, 2018.

[c] Jakub Konecˇny`, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. [17] Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[d] Li, Tian & Sahu, Anit & Zaheer, Manzil & Sanjabi, Maziar & Talwalkar, Ameet & Smith, Federated Optimization for Heterogeneous Networks. *Virginia, 2019.*

[e] Wang, Shiqiang & Tuor, Tiffany & Salonidis, Theodoros & Leung, Kin & Makaya, Christian & He, Ting & Chan, Kevin. Adaptive Federated Learning in Resource Constrained Edge Computing Systems. IEEE Journal on Selected Areas in Communications. PP. 1-1. 10.1109/JSAC.2019.2904348, 2019.