Research Proposal

Mohammad Maaz Owais, Muhammad Hamza Khawaja, Muhammad Taha, Shazer Ali

1. Introduction

Federated Learning involves training a shared global model using local data and compute on various user devices. Several approaches have been proposed to implement this paradigm starting with FedAvg [6]. However, the system heterogeneity in participating devices poses a significant challenge that needs to be addressed. In developing countries, 57% of population are categorised as low-end users. [8] This has implications for fairness due to introduction of systematic bias, in addition to degradation in model accuracy. Recent works such as FedProx [9] and Hassas [7] have attempted to include slow devices by incorporating partial work and serving a subset model according to device characteristics, respectively. These approaches have mostly been evaluated on simulations using LEAF Benchmark [2]. To the best of our knowledge, none of these works have been evaluated on federated learning systems using real-world devices with a sufficiently large number of users.

To this end, we propose the development of a federated learning system that supports atleast 100 real-world users. We aim to achieve this by building a robust FL system and deploying a suitable application on top of it. In general, the application will leverage a machine learning model that benefits from collaborative learning in a privacy-preserving manner. It will provide the user with an attractive incentive and will leverage the data, generated through the user's interaction with the application, for model training. Therefore, this will provide a conducive platform to concretely evaluate the robustness of Hassas as well as other FL frameworks. Conducting experiments on real-world data in the face of dynamic changes in systems heterogeneity, including state changes, will provide valuable insights that will benefit the community.

2. Related Work

HeteroFL was the first work to challenge the assumption that local models must have the same architecture as the global model. [4] **FedDST** proposes approaches to make on-device computation and in-network communication more efficient. [1] **FedProx** incorporates partial work to include low-end devices [9]. However, these frameworks mostly leverage the **LEAF** benchmark for experimentation. [2] In addition, [3] aims to demonstrate the impact of straggler devices by measuring the impact of cpu resource heterogeneity on training time. The evaluation is conducted using an emulated environment with clients running in Docker containers deployed on a AWS EC2 Virtual Machine instance. The following FL approches perform evaluations using small-scale testbeds of real-world devices. **PruneFL** uses a set of Raspberry Pi devices connected to a central server (PC) and a simulated setting. [**?**] Time measurements from Raspberry Pi devices are used for experiments conducted on the simulated setting. **Hermes** leverages structured pruning to find a small subnetwork for each device and aggregating across overlapping parameters to learn a structured sparse deep neural network. [5] The framework is evaluated on a testbed of 3 Google Pixel smartphones connected to a central server. Similarly, [11] utilized a testbed of 4 devices with different device characteristics to to evaluate their approach using the MNIST dataset. We find few works that have performed evaluations on real-world devices. To facilitate experiments on real devices, **Flower** a highly developed framework has been introduced that can support millions of real world clients using just a few high end GPUs.

3. Design

To our knowledge, our work is the first large-scale FL deployment on real-world user devices that will aid evaluations for academic purposes.

3.1. Android Application

1. Frontend
2. Data Cache
3. Prediction Model

Our application in target will be a specific University dedicated chatbot which handles a user's questions related to Graduate and Undergraduate level applications and academic assistance. Essentially our user will interact with our chatbot over a mobile application, where question will be provided as input. The question will be given as an input to our Deep Neural Network model on the user's device which will then produce an output as answer [10]. We will require user to provides us with the feedback for the query related answer, to improve the model. Our application will perform some pre-processing and then cache each question and answer pair with the feedback given, to later allow for the model to be trained on the device.

3.2. FL Platform

Client Runtime

- Model Training
- Device Analytics
  No personally identifiable information will be logged.
  - Logs
  - Device OS
  - Device Model
  - Device State

- Training time
- Memory profile
- Battery profile

Server Runtime

3.3. Coordination Layer

Device Management The selection and reporting of devices for FL tasks are part of device management. In order to make the device management process inclusive, we add functionality to the mechanism described in the paper **Towards Federated Learning At Scale: System Design**. Devices that meet the eligibility requirements check in with the server for any FL tasks that are open as part of the selection process. After allocating the FL tasks to available devices, the server waits for participants to report updates. To signal the end of a round, the server uses a goal counter and timer. The goal count is split into two quorums, one for high-end devices and the other for low-end devices, both of equal size. The round is marked complete if the goal count is achieved within the timeout value else the round is discarded.

Failure Detection The device is declared failed if it violates the eligibility criteria during a FL round (unmetered connection, plugged in to power). During the FL round, the failure is detected using the ping and ack mechanism between the server and the device. This aids the server in keeping track of active participants of a FL round.

4. Methods

The tasks we need to accomplish can be broadly categorized as follows.

1. Application Development
2. FL Platform Development
3. User Base Development

4.1. Phase I

We will begin by developing an android application. Simultaneously, we will develop an end-to-end prototype of the FL platform for 4 devices. Once a working prototype is in place, the next step will be to scale it to 20 users and perform thorough end-to-end testing. The primary objective is to ensure efficient, seamless deployment of the FL platform on a sufficiently large number of clients.

4.2. Phase II

Once this is achieved, we will integrate our android application with the FL platform.

4.3. Phase III

Finally, we will make an effort to scale our application to 100+ users.

5. Timeline and Work Division

Timeline and division of work.

References

[1] BIBIKAR, S., VIKALO, H., WANG, Z., AND CHEN, X. Federated dynamic sparse training: Computing less, communicating less, yet learning better. *CoRR abs/2112.09824* (2021).

[2] CALDAS, S., WU, P., LI, T., KONEČNÝ, J., MCMAHAN, H. B., SMITH, V., AND TALWALKAR, A. LEAF: A benchmark for federated settings. *CoRR abs/1812.01097* (2018).

[3] CHAI, Z., FAYYAZ, H., FAYYAZ, Z., ANWAR, A., ZHOU, Y., BARACALDO, N., LUDWIG, H., AND CHENG, Y. Towards taming the resource and data heterogeneity in federated learning. In *2019 USENIX Conference on Operational Machine Learning (OpML 19)* (Santa Clara, CA, May 2019), USENIX Association, pp. 19–21.

[4] DIAO, E., DING, J., AND TAROKH, V. Heterofl: Computation and communication efficient federated learning for heterogeneous clients. *CoRR abs/2010.01264* (2020).

[5] LI, A., SUN, J., LI, P., PU, Y., LI, H., AND CHEN, Y. Hermes: An efficient federated learning framework for heterogeneous mobile clients. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking* (New York, NY, USA, 2021), MobiCom '21, Association for Computing Machinery, p. 420–437.

[6] MCMAHAN, H. B., MOORE, E., RAMAGE, D., AND Y ARCAS, B. A. Federated learning of deep networks using model averaging. *CoRR abs/1602.05629* (2016).

[7] MUNIR, M. T., SAEED, M. M., ALI, M., QAZI, Z. A., AND QAZI, I. A. Fedprune: Towards inclusive federated learning. *CoRR abs/2110.14205* (2021).

[8] NASEER, U., BENSON, T. A., AND NETRAVALI, R. Webmedic: Disentangling the memory-functionality tension for the next billion mobile web users. In *Proceedings of the 22nd International Workshop on Mobile Computing Systems and Applications* (New York, NY, USA, 2021), HotMobile '21, Association for Computing Machinery, p. 71–77.

[9] SAHU, A. K., LI, T., SANJABI, M., ZAHEER, M., TALWALKAR, A., AND SMITH, V. On the convergence of federated optimization in heterogeneous networks. *CoRR abs/1812.06127* (2018).

[10] VAMSI, G. K., RASOOL, A., AND HAJELA, G. Chatbot: A deep neural network based human to machine conversation model. In *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)* (2020), pp. 1–7.

[11] WANG, C., WEI, X., AND ZHOU, P. Optimize scheduling of federated learning on battery-powered mobile devices. In *2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (2020), pp. 212–221.