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 [10]. 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. [12] This has implications for fairness due to introduction of systematic bias, in addition to degradation in model accuracy. Recent works such as FedProx [13] and Hassas [11] 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 [4]. 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 includes 100+ active 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 challenges the assumption that local models must have the same architecture as the global model. [6] On the other hand, **FedDST** proposes approaches to make on-device computation and in-network communication more efficient. [2] **FedProx** incorporates partial work to include low-end devices [13]. However, these frameworks mostly leverage the **LEAF** benchmark for experimentation. [4] In addition, [5] 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. [7] 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. [9] The framework is evaluated on a testbed of 3 Google Pixel smartphones connected to a central server. Similarly, [15] utilized a testbed of 4 devices with different device characteristics to evaluate their approach using the MNIST dataset. Hence, we find few works that have performed evaluations on real-world devices.

3. Preliminary Design

3.1. Android Application

Personalized Chatbot 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. [14] 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.

Prompt Recommendation System The purpose of this application is to improve the user's search experience. It utilizes APIs from leading search engines such as Google or Microsoft Edge to track the user's search patterns. The application can determine whether the user was satisfied with the search result or not by analyzing their re-attempts with modified queries, which will be provided by the API integrated into the application. The model will continually learn from the user's search patterns and provide the most relevant prompts based on the main category of their query.

Challenges The target audience for our applications consists primarily of low-end smartphone users, who are likely to have limited access to an unmetered network. [8] The training of models on these devices may be hindered by restrictions imposed by the federated learning model, such as requiring the device to be idle and charged. Additionally, the chatbot application requires frequent interaction from users in the form of questions and feedback, which may not be feasible

for all users, especially those outside of academia. The limited RAM on low-end smartphones may also result in the operating system interrupting background processes, causing delays or disconnection from the central server. The Prompt Recommendation System may also encounter limitations with APIs in terms of data access and rate, which can negatively impact the storage and user experience of the application.

3.2. FL Platform

The platform will enable model training on user devices using the data generated by application use. For this purpose, we intend to leverage **FLOWER** as it provides multiple new features that distinguish it from other platforms. [1] We will also consider the possibility of scaling the existing testbed for this purpose. We will decide between the two by weighing their pros and cons after evaluating on a small set of devices. It is imperative to note that these platforms do not include complete protocols for practical deployment that we will have to implement from scratch. We will also require some mechanisms for device analytics to observe devices once we scale to a large number of users.

Client Runtime
- Model Training
- Device Analytics
  We will gather the following information (not exhaustive): logs, device info, training time, memory usage, battery usage, interrupts, proccesses, cpu usage. No personally identifiable information will be collected.

Server Runtime
- Training Plan
- Parameter Aggregation
- Coordination

3.3. Coordination Layer

This layer will be responsible for tackling practical deployment issues at scale. For example, device execution may be interrupted, resource utilization may be ineffective, or device connectivity could be unreliable.

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 [3]. 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.

Assignment	Deadline	Members Responsible
Literature Review	Feb 17	All
Platform Evaluation	Feb 20	Taha, Maaz
FL Deployment (4 Devices)	Feb 26	Hamza, Taha
FL Deployment (20 Devices)	Mar 15	Taha, Hamza
Mid Report	Mar 30	All
Application Prototype	April 1	All, Shazer (Lead)
App Integration	April 5	Shazer, Taha
Testing	April 5	Maaz, Hamza
User-base Development	April 15	All
Baseline Evaluation	April 22	Hamza, Maaz, Taha
Final Report	April 27	All

Table 1: Timeline and Work Division

Failure Detection The device is declared failed if it violates the eligibility criteria during a FL round (unmetered connection, plugged in, idle). 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. Approach

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 using the phones at our disposal. 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. We will extend the user base to friends and family so that we can receive quick feedback, resolve issues, and improve the application before further scaling. 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. We will focus on promotion through college events like Startup Weekend where people from all over the country arrive to observe new and innovative ideas that emerge from within the student body.

5. Resources

https://github.com/taham0/topics-research-project

References

[1] BEUTEL, D. J., TOPAL, T., MATHUR, A., QIU, X., PARCOLLET, T., AND LANE, N. D. Flower: A friendly federated learning research framework. *CoRR abs/2007.14390* (2020).

[2] 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).

[3] BONAWITZ, K. A., EICHNER, H., GRIESKAMP, W., HUBA, D., INGERMAN, A., IVANOV, V., KIDDON, C., KONEČNÝ, J., MAZZOCCHI, S., MCMAHAN, H. B., OVERVELDT, T. V., PETROU, D., RAMAGE, D., AND ROSELANDER, J. Towards federated learning at scale: System design. *CoRR abs/1902.01046* (2019).

[4] 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).

[5] 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.

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

[7] JIANG, Y., WANG, S., KO, B. J., LEE, W., AND TASSIULAS, L. Model pruning enables efficient federated learning on edge devices. *CoRR abs/1909.12326* (2019).

[8] KAMSSU, A. J. Global connectivity through wireless network technology: A possible solution for poor countries. *Int. J. Mob. Commun. 3*, 3 (mar 2005), 249–262.

[9] 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.

[10] 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).

[11] 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).

[12] 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.

[13] 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).

[14] 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.

[15] 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.