FLOWER: AN OPEN-SOURCE FEDERATED LEARNING FRAMEWORK FOR BOTH INDUSTRY AND RESEARCH

Daniel J. Beutel [1 2] **Taner Topal** [1 2] **Akhil Mathur** [1 3] **Xinchi Qiu** [1] **Titouan Parcollet** [4]
Pedro Porto Buarque de Gusmão [1] **Nicholas D. Lane** [1]

As mobile devices carrying high-quality sensors become ubiquitous, increasing amounts of data reflecting user activities and preferences are now generated at the edge. Being able to harvest such wealth of information is key to training intelligent systems on real-world statistics. However, due to privacy concerns, client availability, and mobile devices' distributed nature, the common approach of first collecting all users' data followed by performing centralised training is no longer feasible.

To help solve these issues, Federated Learning (FL) emerges as a distributed training technique where thousands of devices collaboratively learn a single model whilst keeping their individual data local. In comparison to centralised learning, FL is still in its infancy and provides fertile ground for research on problems related to low device-availability, hardware heterogeneity, and non-iid data distributions.

Interestingly, most of FL frameworks available today work as simulators and can only run on a single physical machine. This lack of scalability not only prevents us from running realistic large-scale experiments with thousands of devices, but it also delays the transition of FL solutions from research to production.

In this work we present current developments in Flower, a new and scalable FL framework designed to meet both Industry and Research needs. We begin by describing the key ideas behind Flower's architecture which include:

- *ML framework-agnostic:* Flower decouples communication and weight aggregation from local training. This allows clients to use whichever ML framework they prefer.
- *Client-agnostic:* Clients participating in a FL round need not be the same. Flower should then be interoperable with different programming languages, operating systems and hardware settings; thus, allowing for system heterogeneity.
- *Expandable:* Flower should facilitate code integration to incorporate new research ideas.
- *Accessible:* Adapting an existing centralised-training pipeline to federated learning should require little effort.
- *Scalable:* Given that real-world FL tasks require large numbers of clients, Flower should scale to a large number of concurrent clients to allow research on a realistic scale.

We follow by describing how these ideas are implemented in Flower's programming interface. Scalability to a thousand clients is proven through a series of experiments using CIFAR-10 and ResNet-18. An expected increase in accuracy caused by the increase in the number of devices is verified although this gain quickly saturates for more than 50 clients. In fact, scaling from 500 to 1000 clients shows no performance gain, raising the question whether this is due to the simplicity of the dataset or limitations on the aggregation strategy.

Scalability is also verified in terms of data size. We show that Flower can support federated training on web-scale datasets by training a Resnet-18 model on ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012 using 50 clients for over 15 days.

Finally, we present current developments around the Flower Ecosystem including Flower Baselines, a series of scripts dedicated to help reproduce Federated Learning experiments at large scales.

[1]University of Cambridge, UK [2]Adap, Germany [3]Nokia Bell Labs, UK [4]Avignon Université, France. Correspondence to: Pedro Porto Buarque de Gusmão <pp524@cam.ac.uk>.