Enabling On-Device Training of Speech Recognition Models with Federated Dropout

Dhruv Guliani (dguliani@google.com), Lillian Zhou, Changwan Ryu, Tien-Ju Yang, Harry Zhang, Yonghui Xiao, Françoise Beaufays, Giovanni Motta

Introduction
End-to-end neural ASR models can be trained using Federated Learning (FL) [4] to preserve user privacy by removing the need to send raw user data to servers. FL optimization proceeds in synchronous rounds [5], sending a set of clients (devices) copies of a model for local training and aggregating model updates after optimization.

Costs Associated with FL
ASR models are much larger than models previously trained with FL [6, 7, 8, 9] and are often containing over 100M parameters. The following are costs associated with FL:

- Communication costs: sending and aggregating model updates after optimization.
- On-Device costs (CPU and memory usage).

Federated Dropout
Federated Dropout (FD) reduces both communication and on-device computation costs by reducing the size of models trained on clients. FD leverages the insight from domain adaptation in Table 1: showing:

- Communication costs: sending and aggregating models, dealing with heterogeneous network dynamics, etc.
- On-Device costs (CPU and memory usage).

Non-Streaming Conformer Results
Figure 2 shows:

- FD can provide a quality/cost trade-off as WER gets slightly worse with increasing FD rate
- Higher FD rates usually converge slower
- PR is slightly worse than PCPR, but usable if engineering resources are limited
- Higher report goals can improve convergence speed and quality. Further improvements are hypothesized to be possible with hardware tuning.

Domain adaptation in Table 1 shows:

- FD is effective with production-grade exps.
- PR performs very well, minor quality loss
- Higher dropout rates (50% and greater) result in degradation in MF WER from the No-MF Baseline.

Streaming Conformer Results
Table 1: Streaming Conformer Initial Setup

Per-layer dropout ratios (Figure 4) show:

- It is possible to achieve better quality or lower cost with per-layer FD

Table 2: Quality of Sub-Models
Submodels were sampled (50%) with 50% dropout from two streaming conformers, one trained under FL with FD and one without, with the result that:

- FD enabled sub-models to achieve a much lower WER with lower variance.

Model & Dataset
We use two variations of the Conformer architecture shown in Figure 1: a Non-Streaming Conformer [2] with 119M parameters and a Streaming Conformer [3] with 137M parameters. The Non-Streaming Conformer is trained from scratch under FL using a speaker-split Librispeech corpus [4]. The Streaming Conformer is first trained on a production-grade Multi-Domain (MD) dataset containing 37k hours of audio server-side, and then trained on Medium-Form (MF) data from a new domain under FL. We call these tasks "training from scratch" and "domain-adaptation" respectively.

Feed Forward layers are the largest parts of both models, making up 60% of the Non-Streaming Conformer and 55% of the Streaming Conformer. Our application of federated dropout is hence limited to just these layers for this investigation.

Conclusion
Federated Learning is key to user privacy and ensures that raw user data never leave the device. To leverage this, we must be able to fit model training onto edge devices. End-to-end neural ASR models can contain well over 100 million parameters, creating significant communication and computation cost hurdles on the edge. We argued that Federated Dropout is a promising technique to reduce this cost and explored various configurations to improve its effectiveness. We illustrated a usable quality/cost trade-off allowing for client model size reduction between 6-22%, with WER improvements in a domain adaptation setting ranging from 34-3% respectively. We also showed that FD causes capable sub-models to form within the full model, allowing the same model to be downsampled for inference. We hope this work inspires deeper investigations and applications of both model size reduction and sub-model training.

Per-Layer FD
We explore making FD more effective by varying the amount of dropout applied across different layers. We target layers for additional dropout using the idea that certain layers may be ambient, or less important to the model’s performance [22], with the aim of improving upon the results of uniform FD.

Sub-Model Evaluations
Other properties of FD are also investigated by sampling and evaluating sub-models from the full size model after training. Sub-models are obtained by removing activations and corresponding neurons in the same way as the FD procedure and are evaluated without any further training.

Method
The training from scratch task is used to study the general characteristics of FD with ASR. Note that in Algorithm 1, two edge cases exist:

- all maps in ρ are unique: Per-Client-Per-Round (PCPR)
- all maps in ρ are the same: Per-Round (PR)

We explore dropout rates, report goals, PCPR vs PR FD, and comment on convergence time and quality.

Domain Adaptation
The domain adaptation task is a more realistic setting for federated training of ASR models, wherein a well-trained server-side model is adapted to a new domain with FL on edge devices.

Training from Scratch
We explore making FD more effective by varying the amount of dropout applied across different layers. We target layers for additional dropout using the idea that certain layers may be ambient, or less important to the model’s performance [22], with the aim of improving upon the results of uniform FD.

Sub-Model Evaluations
Other properties of FD are also investigated by sampling and evaluating sub-models from the full size model after training. Sub-models are obtained by removing activations and corresponding neurons in the same way as the FD procedure and are evaluated without any further training.

Key References