Federated Learning for Keyword Spotting

David Leroy, Alice Coucke, Thibaut Lavril, Thibault Gisselbrecht and Joseph Dureau
Snips, Paris
{firstname.lastname}@snips.ai

snips

1. Contribution

We propose a practical approach based on federated learning to solve out-of-domain issues with continuously running embedded speech-based models such as wake word detectors.

- we conduct an extensive empirical study of the federated averaging algorithm for the "Hey Snips" wake word based on a crowdsourced dataset that mimics a federation of wake word users.
- we empirically demonstrate that using an adaptive averaging strategy inspired from Adam in place of standard weighted model averaging highly reduces the number of communication rounds required to reach our target performance.

3. The "Hey Snips" Dataset

- **Distributed**: 1.8k distinct contributors
- **Non-iid**: each contributor used their own recording setting, and recorded themselves saying several occurrences of the *Hey Snips* wake word (18% of total utterances) along with negative short sentences from various text sources
- **Unbalanced**: varying amounts of training utterances per user *(mean: 39, standard dev: 32)*.
- train, dev and test splits (see Table 1) contain distinct users - eval accounts for generalization power to unseen users

Train set	Dev set	Test set	Total
1,374 users	200 users	200 users	1,774 users
53,991 utt.	8,337 utt.	7,854 utt.	69,582 utt.

Table 1: Dataset statistics.

Open access for non-commercial use: to promote repeatable research [1]

- `https://research.snips.ai/datasets/keyword-spotting`

4. Model

Acoustic features:

- 20-dimensional log-Mel filterbank energies
- extracted from the input audio every 10ms over a window of 25ms.

Labelling:

- 4 output labels ("Hey", "sni", "ps", and "filler")
- label is at the frame level based on aligner output

Architecture: inspired from [2]

- input window: 32 stacked frames, symmetrically distributed in left and right contexts
- 5 stacked dilated convolutional layers of increasing dilation rate, 2 fully-connected layers followed by softmax ($\sim 200k$ parameters)
- posterior handling [3] generates a confidence score for every frame by combining the smoothed label posteriors
- model triggers when confidence is above threshold τ that defines the operating point of the model

Training:

- trained using cross-entropy on target labels
- threshold τ set for 5 False Alarms per Hour (FAH) on the dev set

2. Federated Optimization [4]

Objective function

- supervised learning objective function $f_i(w) = l(x_i, y_i, w)$ that is the loss function for the prediction on example (x_i, y_i) when using a model described by a real-valued parameter vector w of dimension d.
- datapoints i are partitioned across K users, each user being assigned their own partition $\mathcal{P}_k$, $|\mathcal{P}_k| = n_k$.

$$\min_{w \in \mathbb{R}^d} f(w) \quad \text{where} \quad f(w) \stackrel{\text{def}}{=} \sum_{k=1}^{K} \frac{n_k}{n} \times F_k(w), \text{ with } F_k(w) = \frac{1}{n_k} \sum_{i=1}^{n_k} f_i(w) \tag{1}$$

Optimization procedure : The model is initialized with a given architecture on a central *parameter server* with weights w_0. Once initialized, the parameter server and the user's devices interact synchronously with each other during *communication rounds*. At time $t \in [1, .., T]$:

1. the central model w_{t-1} is shared with a subset of users $\mathcal{S}_t$ randomly selected from the pool of K users (participation ratio C).
2. each user $k \in \mathcal{S}_t$ performs one or several training steps on their local data using mini-batch SGD with a local learning rate η_{local}. The number of steps performed locally is $E \times max(ceil(\frac{n_k}{B}), 1)$, n_k being the number of datapoints available locally, E the number of local epochs and B the local batch size.
3. users from $\mathcal{S}_t$ send back their model updates $w_{t,k}, k \in \mathcal{S}_t$ to the parameter server once local training is finished.
4. the server computes an average model w_t based on the user's individual updates $w_{t,k}, k \in \mathcal{S}_t$, each user's update being weighted by $\frac{n_k}{n_r}$, where $n_r = \sum_{k \in \mathcal{S}_t} n_k \approx C \times \sum_{k=1}^{K} n_k$.

Per-coordinate gradient update: the averaging step 4 can be written as global gradient update. This motivates the use of adaptive per-coordinate updates that have proven successful for centralized deep neural networks optimization such as Adam [5].

$$w_t \leftarrow w_{t-1} - \eta_{global} \mathcal{G}_t \text{ where } \mathcal{G}_t = \sum_{k \in \mathcal{S}_t} \frac{n_k}{n} (w_{t-1} - w_{t,k}) \tag{2}$$

5. Results

Evaluation metrics:

- number of communication rounds required to reach early stopping target of 95% recall / 5 FAH on dev set
- at fixed threshold, FAH evaluated on test set negative (hard) and background negative data

Standard setting: training in a centralized fashion e.g mini-batch SGD with data from train set users being randomly shuffled, stopping criterion reached in 400 steps (~ 2 epochs)

Federated setting Best performances are obtained in the following setting:

- user parallelism is set to $C = 10\%$. See Figure 1, the gain of using $C = 50\%$ e.g half of users for each round is insignificant when compared to using 10%, especially in the later stages of convergence.
- the *Adam* global averaging strategy with $\eta_{global} = 0.001$ is used. As seen in Table 2, global adaptive learning rates based on Adam drastically accelerates convergence when compared with standard averaging strategies.
- local training is optimal for $E = 1$ and $B = 20$ with a local learning rate of 0.01. Experiments have shown limited improvements coming from increasing the load of local training as the number of communication rounds required to reach the stopping criterion on the dev set ranges between 63 and 112 communication rounds for $E \in [1, 3]$ and $B \in [20, 50, \infty]$.

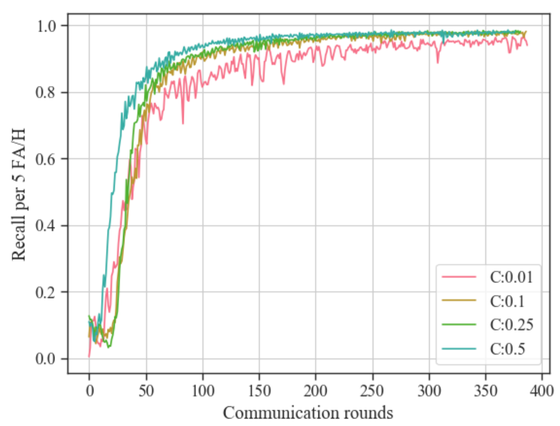

Figure 1: Effect of the share of users involved in each round C on the dev set recall / 5 FAH, *FedSGD*, Adam global averaging, $\eta_{global} = 0.001, \eta_{local} = 0.01$

Avg. Strategy	100 rounds	400 rounds
Standard $\eta_{global} = 1.0$	29.9%	67.3%
Adam $\eta_{global} = 0.001$	93.50%	98.29%

Table 2: Dev set recall / 5 FAH for various averaging strategies - *FedSGD*, $C = 10\%$

The parameters above lead to the following results:

- **Communication rounds**: the number of training steps needed to reach the stopping criterion amounts to approximately 3300 for 100 communication rounds e.g. 8.25 times the number of steps required in the standard setting, with much smaller batches.
- **Communication cost**: the upstream communication cost is 8MB per user over the course of the optimization procedure, amounting to 110GB over the course of the whole optimization process with 1.4k users involved during training.
- **False alarm evaluation**: 3.2 FAH on the negative test data, 3.9 FAH on Librispeech, and respectively 0.2 and 0.6 FAH on our internally-collected news and TV datasets.
- **Later convergence stage**: same parameters, 400 rounds yields 98% recall / 0.5 FAH on the test set for an upload budget of 32 MB per user.

References

[1] Alice Coucke, Mohammed Chlieh, Thibault Gisselbrecht, David Leroy, Mathieu Poumeyrol, and Thibaut Lavril. Efficient keyword spotting using dilated convolutions and gating. *arXiv preprint arXiv:1811.07684*, 2018.
[2] Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. A time delay neural network architecture for efficient modeling of long temporal contexts. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.
[3] Guoguo Chen, Carolina Parada, and Georg Heigold. Small-footprint keyword spotting using deep neural networks. In *Acoustics, speech and signal processing (icassp), 2014 ieee international conference on*, pages 4087–4091. IEEE, 2014.
[4] H. Brendan McMahan, Eider Moore, Daniel Ramage, and Blaise Agüera y Arcas. Federated learning of deep networks using model averaging. *CoRR*, abs/1602.05629, 2016.
[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.