Cooperative Learning via Federated Distillation over Fading Channels

*Jinhyun Ahn, **Osvaldo Simeone, and *Joonhyuk Kang

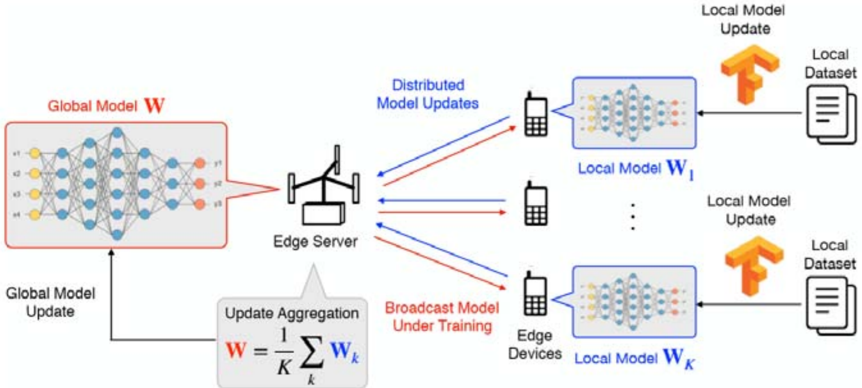
*KAIST, South Korea,

**King's College London, UK

Contents

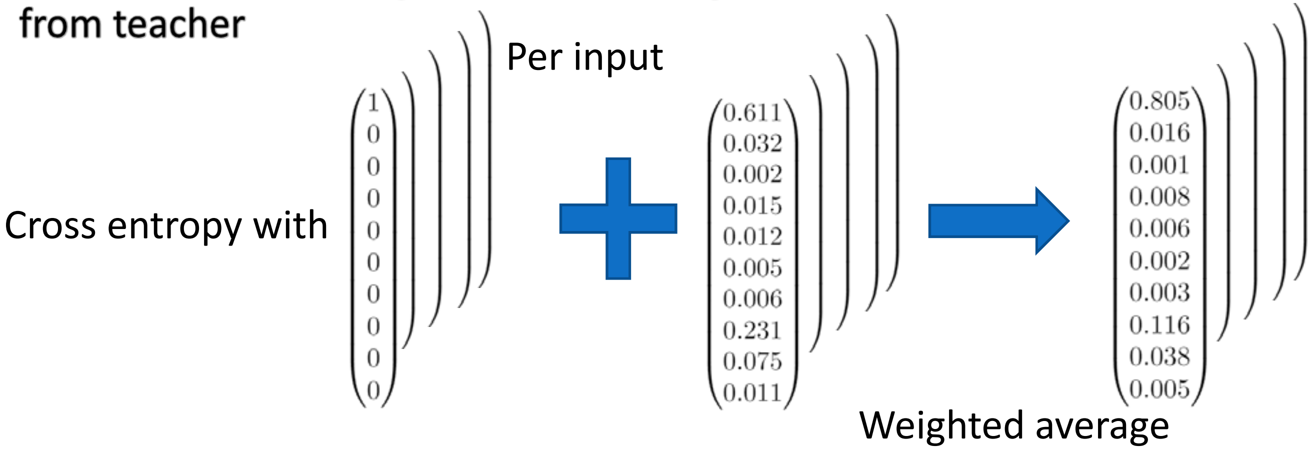
Introduction

- Federated Learning
  - Federated learning (FL), is developed recently, which features distributed learning at edge devices and periodic local-update of model (model coefficients or gradients) averaging at an parameter server (PS)
  - Nevertheless, the updates uploading in FL can be still bandwidth-consuming as an AI model usually comprises millions to billions of parameters [7]
  - A key research issue that is particularly hot recently is to reduce the overhead in update uploading to further accelerate the model training process [8]–[13]
    - Addressing the straggler effect in synchronous update averaging
    - Developing lazily updating algorithm that schedules only those devices with significant updates to save the updating bandwidth
    - Compress gradient vectors by exploiting its inherent sparsity (most of the gradient elements are insignificant and thus can be truncated without harming the model accuracy)



Introduction

- Federated distillation
  - To alleviate this problem, federated distillation (FD) was introduced for classification problems in [14]
  - Distillation for learning model was proposed by Hinton et al. [15]
    - To transfer a knowledge about a learning model, output vectors per inputs are sent from teacher



  - In FD, devices periodically exchange the average output logit vectors per labels instead of local update of model in FL **(less information but lower accuracy gain than FL)**

→ We propose a novel hybrid federated distillation (HFD) scheme that aims at bridging the performance gap between FD and FL

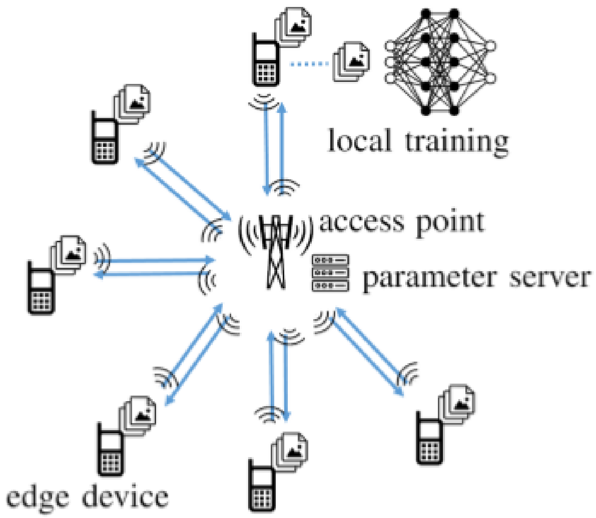
Introduction

- Wireless Implementation of FD and HFD
  - In many practical implementations, however, bandwidth of the communication channel from devices to the PS turns out to be the main bottleneck [16], [17]
  - Recently, a multiple access scheme called “over-the-air” computation (AirComp) is particularly appealing in the scenario as it integrates transmission and computation and allows “one-shot” data aggregation by exploiting the waveform-superposition property of a multi-access channel (MAC) [18-19], [25]
  - There is no previous work about the wireless implementation of FD
  - We propose a communication scheme for the implementation of FD focusing on the quantization and compression
  - Both a conventional digital scheme and an analog scheme are considered for the communication in the uplink and downlink

Problem Definition

- Problem Definition
  - K devices communicate via an Access Point (AP) so as to train a machine learning model that outperforms a model trained solely on the local training set
  - For device k
    - Data set $\mathbb{D}_k$: $(\mathbf{c}, \mathbf{t})$ (vector of covariates, one-hot encoding vector)
    - Trains its own neural network model : $\mathbf{w}_k$, $W \times 1$
    - Neural network produces the logit vector : $\mathbf{s}(\mathbf{c}|\mathbf{w}_k)$

      the probability vector : $\hat{\mathbf{t}}(\mathbf{c}|\mathbf{w}_k)$



$$\hat{\mathbf{t}}(\mathbf{c}|\mathbf{w}_k) = \hat{\mathbf{t}}(\mathbf{s}(\mathbf{c}|\mathbf{w}_k)) = \frac{1}{\sum_{i=1}^{L} e^{s_i}} \begin{bmatrix} e^{s_1} \\ \vdots \\ e^{s_L} \end{bmatrix}$$

Training Protocols (FL)

Algorithm 2 Federated Learning (FL)

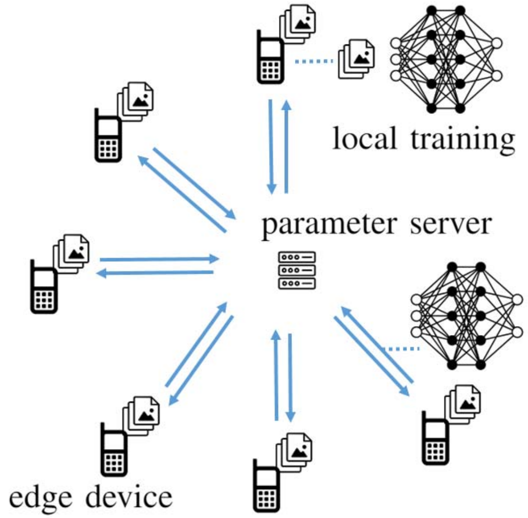
for each iteration $i = 1, \ldots, I$
  for each device $k = 1, \ldots, K$
    download from PS the average weight update

$$\Delta \mathbf{w}_{i-1} = \frac{1}{K} \sum_{k=1}^{K} \Delta \mathbf{w}_{i-1}^{k}$$

    set initial value

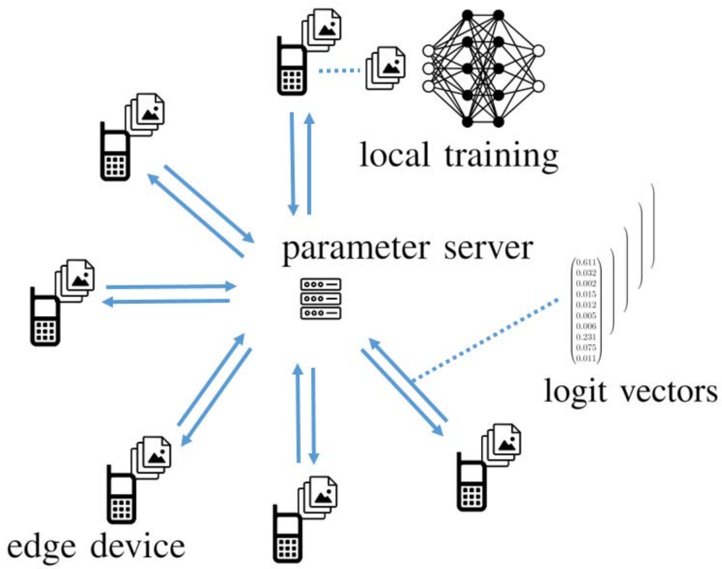
$$\mathbf{w}_i^k = \mathbf{w}_{i-1}^k + \Delta \mathbf{w}_{i-1} - \Delta \mathbf{w}_{i-1}^k \triangleq \mathbf{w}_{i,o}^k$$

    for each iteration of local training
      do SGD update as in (1), for a randomly selected training example $(\mathbf{c}, \mathbf{t}) \in \mathbb{D}_k$
    end
    upload update $\Delta \mathbf{w}_i^k = \mathbf{w}_i^k - \mathbf{w}_{i,o}^k$ to PS



- The weight vectors at each device are initialized to the average weight vectors using the average weight update downloaded from the PS
- Devices carry out a number of local updates using SGD as the update in IL
- Upload the resulting weight vector to the PS

Training Protocols (FD)



- Instead of neural network model parameters, devices exchange the local-averaged logit vector per labels (10 values per 10 classes for MNIST)
- Applies the global-averaged logit vectors per labels for its own local training

Training Protocols (FD)

Algorithm 3 Federated Distillation (FD)

for each iteration $i = 1, \ldots, I$
  for each device $k = 1, \ldots, K$
    download from PS the global-averaged logit vectors for all labels $t = 1, \ldots, L$

$$\mathbf{s}_{i,t} = \frac{1}{K} \sum_{k'=1}^{K} \mathbf{s}_{i,t}^{k'} \tag{2}$$

    obtain the local logit vectors

$$\mathbf{s}_{i,t}^{\backslash k} = \frac{K\mathbf{s}_{i,t} - \mathbf{s}_{i,t}^{k}}{K-1} \tag{3}$$

    initialize $\mathbf{s}_{i+1,t}^{k} := 0$ and $n_{i+1,t}^{k} := 0$ for all labels $t = 1, \ldots, L$
    for each iteration of local training
      do SGD update

$$\mathbf{w}_i^k \leftarrow \mathbf{w}_i^k - \alpha \nabla_{\mathbf{w}_i^k} \left\{ (1-\beta)\,\phi\left(\hat{\mathbf{t}}\left(\mathbf{c}|\mathbf{w}_i^k\right), \mathbf{t}\right) + \beta\phi\left(\hat{\mathbf{t}}\left(\mathbf{c}|\mathbf{w}_i^k\right), \hat{\mathbf{t}}(\mathbf{s}_{i,t}^{\backslash k})\right) \right\} \tag{4}$$

      for a randomly selected training example $(\mathbf{c}, \mathbf{t}) \in \mathbb{D}_k$
      update the logit vector and the label counter

$$\mathbf{s}_{i+1,t}^{k} \leftarrow \mathbf{s}_{i+1,t}^{k} + \mathbf{s}\left(\mathbf{c}|\mathbf{w}_i^k\right)$$

$$n_{i+1,t}^{k} \leftarrow n_{i+1,t}^{k} + 1$$

    end
    upload the local-averaged logit vectors $\mathbf{s}_{i+1,t}^{k} \leftarrow \mathbf{s}_{i+1,t}^{k} / n_{i+1,t}^{k}$ to the PS for all labels $t = 1, \ldots, L$

- In (2) and (3), each device excludes its own information from the averaged logit vectors
- In (4), each device carries out a number of local updates using the averaged logit vectors as a regularizer
- During the local updates, each device computes and uploads the local-averaged logit vectors for all labels to the PS

Training Protocols (HFD)

- The proposed HFD modifies FD by using not only the average logit vector but also the average covariate vector per label, which is shared during a preliminary offline phase
- In the distillation [15],
  - Teacher and students share the same covariates vectors
  - The teacher's knowledge is transferred by sending every logit vectors for all covariates
  - Student uses associated logit vectors for local training of covariates
- In **FD**, the teacher's knowledge is the average logit vectors per labels
- In **HFD**,
  - The teacher's knowledge is average covariate vector and its output logit vectors per labels
  - Updates consist of distillation phase and IL phase
    - Distillation phase : updates over only global averaged covariate vectors using the downloaded logit vectors as regularizer as in FD
    - IL phase : updates over local dataset

Training Protocols (HFD)

Prior to the global iterations

- Obtain the local averaged covariate vectors

$$\tilde{\mathbf{c}}_t^k \quad t = 1, \ldots, L$$

- Download the global averaged covariate vectors exclude its own information

$$\tilde{\mathbf{c}}_t = \frac{1}{K} \sum_{k'=1}^{K} \tilde{\mathbf{c}}_t^{k'}$$

$$\tilde{\mathbf{c}}_t^{\backslash k} = \frac{K\tilde{\mathbf{c}}_t - \tilde{\mathbf{c}}_t^k}{K-1}$$

Algorithm 4 Hybrid Federated Distillation (HFD)

for each device $k = 1, \ldots, K$
  for each iteration $i = 1, \ldots, I$
    download from PS the global-averaged logit vectors (5) for all labels $t = 1, \ldots, L$
    obtain the logit vectors (6)
    for each iteration of the distillation phase of local training
      do SGD update as in (7) for a data point $(\tilde{\mathbf{c}}_t^{\backslash k}, \mathbf{t})$ for a randomly chosen label t
    end
    for each iteration of the IL phase of local training
      do SGD update as in (3) for a randomly selected training example $(\mathbf{c}, \mathbf{t}) \in \mathbb{D}_k$
    end
    upload the logit vectors

$$\mathbf{s}_{i+1,t}^k = \mathbf{s}\left(\tilde{\mathbf{c}}_t^k \mid \mathbf{w}_i^k\right)$$

    to the PS for all labels $t = 1, \ldots, L$

- As in FD, each device downloads the global averaged logit vectors (and exclude)
- At the distillation phase, does SGD updates with the covariate vectors using the logit vectors as a regularizer for a randomly chosen label
- At the IL phase, each device does SGD updates with its own local dataset
- After the local updates, computes and uploads the output logit vectors of local averaged covariate vectors per labels

Wireless Cooperative Training

- Proposed four wireless implementations of FL and FD/HFD
  - Digital (D) or analog (A) communication in uplink and downlink
  - digital-digital (D-D) / digital-analog (D-A)
  - analog-digital (A-D) / analog-analog (A-A)
- Digital transmission for both uplink and downlink is based on separate source-channel coding
  - UL: Equal resource allocation to devices, sparsification and quantization(FD/HFD)
  - DL: Broadcast after compression and quantization
- Analog transmission implements joint source-channel coding through over-the-air computing
  - UL: Simultaneous transmission in uncoded manner
  - DL: Broadcast → Consider scaling factor and AMP algorithm at each device

Wireless Cooperative Training

Channel Model

- During each information exchange phase of the i-th global iteration, devices share a fading uplink multiple-access channel: The received signal is

$$\mathbf{y}_i = \sum_{k=1}^{K} h_i^k \mathbf{x}_i^k + \mathbf{z}_i$$

  - h_i^k : quasi-static fading channel from the device k to the AP
  - $\mathbf{x}_i^k$: $T_U \times 1$ signal transmitted by the device k
  - $\mathbf{z}_i$: $T_U \times 1$ noise vector with i.i.d. $\mathcal{CN}(0,1)$ entries
  - Each device k has a power constraint $\mathrm{E}\left[\|\mathbf{x}_i^k\|_2^2\right]/T_U \le P_U$

- The AP can broadcast to all device in downlink so that the received signal is

$$\mathbf{y}_i^k = g_i^k \mathbf{x}_i + \mathbf{z}_i^k$$

  - g_i^k : quasi-static fading channel from the AP to the device k
  - $\mathbf{x}_i$: $T_D \times 1$ signal transmitted by the AP
  - $\mathbf{z}_i^k$: $T_D \times 1$ noise vector with i.i.d. $\mathcal{CN}(0,1)$ entries
  - The AP has a power constraint $\mathrm{E}\left[\|\mathbf{x}_i\|_2^2\right]/T_D \le P_D$

Wireless Cooperative Training

- Performance Comparison
  - 10 devices train a 6- layer CNN to carry out image classification based on subsets of the MNIST data set available at each device
  - The distributions of datatset are i.i.d.
    - Randomly select disjoint sets of 64 samples from the 60,000 training MNIST examples, and allocate each set to a device
  - Channel fading: Rician fading
  - Number of global iteration: 10
  - Learning rate: 0.001
  - Number of quantization bits: 16
  - Sparsfication level for analog transmission: $q = 4T/5$
  - $T_U = T_D = T$
  - $P_D = P_U + 10 \text{ dB}$

Wireless Cooperative Training

Performance Comparison

- Number of channel uses varies under $P_U = 0$ dB
- FD and HFD significantly outperform FL at low values of T that is, with limited spectral resources
- HFD is seen to uniformly improve over FD
- The A-A scheme is clearly preferable over the alternatives

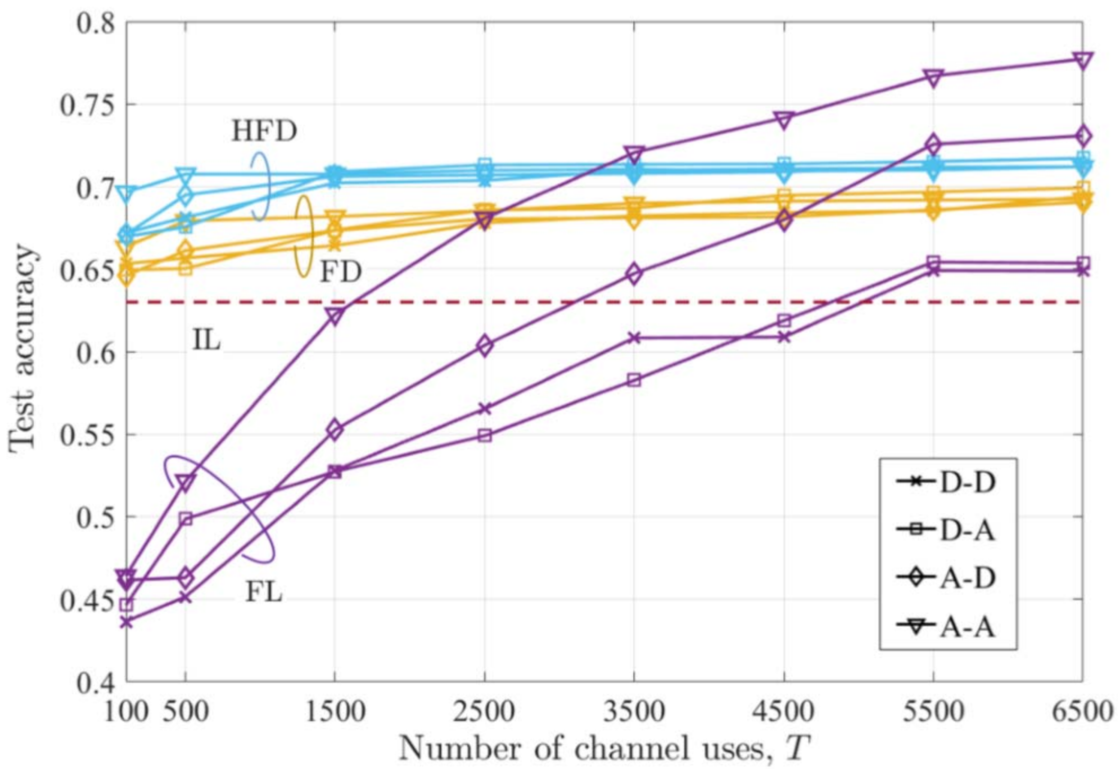


Fig. 2: Classification test accuracy for IL, FL, FD, and HFD under implementations D-D, D-A, A-D, and A-A

Wireless Cooperative Training

Performance Comparison

- The number T is 2500
- The figure confirms that FD and HFD significantly outperform FL at low values of P
- And HFD uniformly improves over FD.
- The A-A scheme shows the best performance, especially for lower values of P
- It is checked that the performance of analog transmission scheme converges when P increases
  (The figure should be plotted for larger SNR)

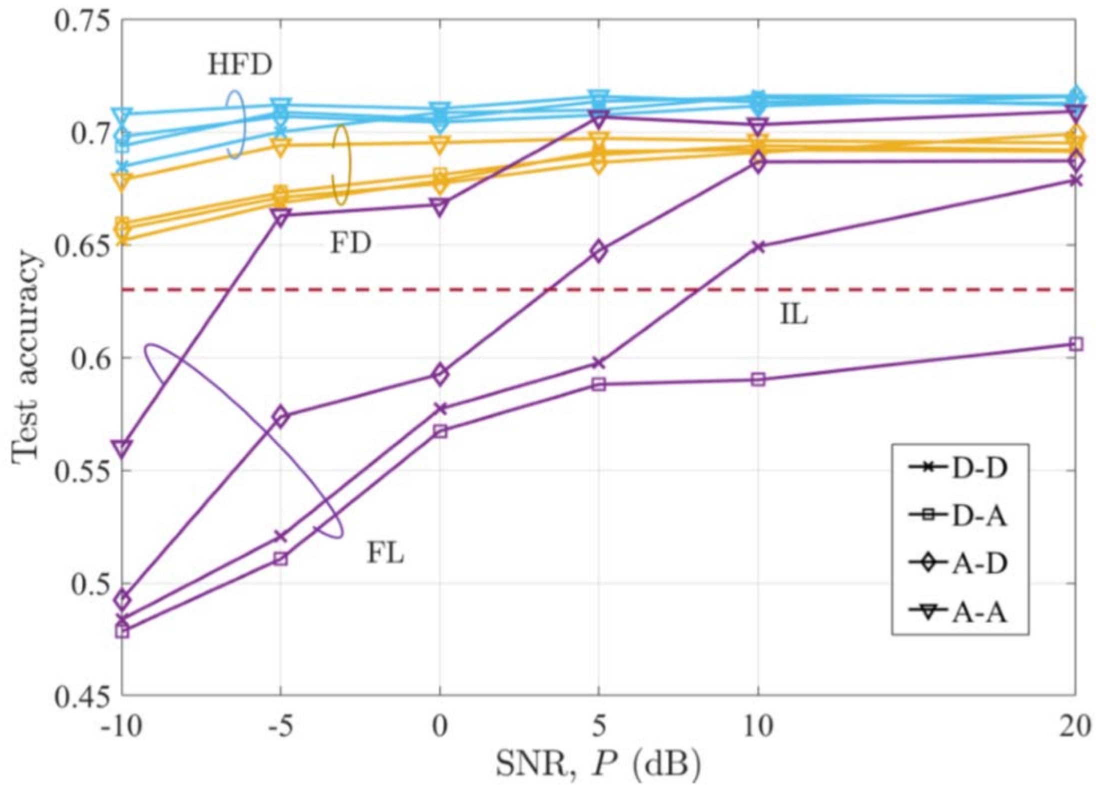


Fig. 2: Classification test accuracy for IL, FL, FD, and HFD under implementations D-D, D-A, A-D, and A-A

Conclusion

Development of FD/HFD to support FL under limited communication resources

- Propose the HFD training protocol
- Investigate the wireless implementations of FD/HFD

Questions → wlsgus3396@kaist.ac.kr

References

[1] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, “Towards an intelligent edge: Wireless communication meets machine learning,” [Online]. Available: https://arxiv.org/abs/1809.00343, 2018.
[2] M. I. Jordan, J. D. Lee, and Y. Yang, “Communication-efficient distributed statistical inference,” Journal of the American Statistical Association, vol. DOI: 10.1080/01621459.2018.1429274, Feb. 2018.
[3] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, “Communication-efficient learning of deep networks from decentralized data,” in International conference on Artificial Intelligence and Statistics (AISTATS), (Fort Lauderdale, Florida), Apr. 2017.
[4] J. Konecn y, H. B. McMahan, F. X. Yu, P. Richtarik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," in Proc. of the NIPS Workshop on Private Multi-Party Machine Learning, Dec. 2016.
[5] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, “Sparse binary compression: Towards distributed deep learning with minimal communication,” in Proc. IEEE Int. Joint Conf. Neural Netw. (IJCNN), to be published.
[6] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, “Communication-efficient learning of deep networks from decentralized data,” in Proc. Int. Conf. on AISTATS, Fort Lauderdale, Florida, Apr. 2017.
[7] 7] G. Zhu, Y. Wang, and K. Huang, “Broadband analog aggregation for low-latency federated edge learning (extended version),” Dec. 2018, arXiv:1812.11494. [Online]. Available: https://arxiv.org/abs/1812.114
[8] J. Chen, X. Pan, R. Monga, S. Bengio, and R. Jozefowicz, “Revisiting distributed synchronous sgd,” [Online]. Available: https://arxiv.org/abs/1604.00981, 2017.
[9] R.Tandon,Q.Lei,A.G.Dimakis,andN.Karampatziakis,“Gradientcoding:Avoidingstragglersindistributedlearning,”inInternational Conference on Machine Learning (ICML), (Sydney, Australia), Aug. 2017.
[10] M. Kamp, L. Adilova, J. Sicking, F. H¨uger, P. Schlicht, T. Wirtz, and S. Wrobel, “Efficient decentralized deep learning by dynamic model averaging,” [Online]. Available: https://arxiv.org/abs/1807.03210, 2018.
[11] T. Chen, G. B. Giannakis, T. Sun, and W. Yin, “Lag: Lazily aggregated gradient for communication-efficient distributed learning,” in Conference on Neural Information Processing Systems (NIPS), (Montreal, CANADA), Dec. 2018.
[12] A. F. Aji and K. Heafield, “Sparse communication for distributed gradient descent,” in Conference on Empirical Methods in Natural Language Processing (EMNLP), (Copenhagen, Denmark), Sep. 2017.
[13] Y. Lin, S. Han, H. Mao, Y. Wang, and W. J. Dally, “Deep gradient compression: Reducing the communication bandwidth for distributed training,” in International conference on learning representation (ICLR), (Vancouver, Canada), May 2018.
[14] E. Jeong, S. Oh, H. Kim, J. Park, M. Bennis, and S. Kim, “Communication-efficient on-device machine learning: federated distillation and augmentation under non-IID private data,” in Proc. NIPS, 2018.
[15] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” in Proc. NIPS, 2014.

References

[16] Y. Lin, S. Han, H. Mao, Y. Wang, and W. J. Dally, “Deep gradient compression: Reducing the communication bandwidth for distributed training,” arXiv:1712.01887v2 [cs.CV], Feb. 2018.
[17] H. Wang, S. Sievert, S. Liu, Z. Charles, D. Papailiopoulos, and S. Wright, “ATOMO: Communication-efficient learning via atomic sparsification,” arXiv:1806.04090v2 [stat.ML], Jun. 2018.
[18] M. M. Amiri and D. G¨und¨uz, “Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air,” ArXiv e-prints, Jan. 2019.
[19] M. M. Amiri and D. G¨und¨uz, “Federated learning over wireless fading channels ,” ArXiv e-prints, July 2019. [20]→ [23] in Deniz 1
[21] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, “Sparse binary compression: Towards distributed deep learning with minimal communication,” arXiv:1805.08768v1 [cs.LG], May 2018.
[22] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, “Energy efficient federated learning over wireless communication networks,” arXiv preprint arXiv:1911.02417, 2019.
[23] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, “A joint learning and communications framework for federated learning over wireless networks,” arXiv preprint arXiv: 1909.07972, 20019.
[24] K. Yang, T. Jiang, Y. Shi, and Z. Ding, “Federated learning via overthe-air computation,” arXiv:1812
[25] B. Nazer and M. Gastpar, “Computation over multiple-access channels,” *IEEE Trans. Inf. Theory*, vol. 53, pp. 3498-3516, Oct. 2007.
[26] X. Cao, G. Zhu, J. Xu, and K. Huang, “Optimal power control for over-the-air computation in fading channels,” ArXiv e-prints, June 2019.
[27] W. Liu and X. Zang, “Over-the-air computation systems: optimization, analysis and scaling laws,” ArXiv e-prints, Sep. 2019.

Appendix: Wireless Cooperative Training

- Uplink Digital Transmission (FL, FD/HFD)
  - Consider for simplicity an equal resource allocation to devices
  - The number of bits that can be transmitted from each device k at the i-th global iteration is given using Shannon's capacity

$$B_{U,k,i} = \frac{T_U}{K} \log_2 \left(1 + \left|h_i^k\right|^2 KP_U\right)$$

  - Each device k compresses the corresponding information to be sent to the AP to no more than $B_{U,k,i}$ bits
  - Devices are aware of the rate and hence of the channel power
  - AP has full channel state information

Appendix: Wireless Cooperative Training

- Uplink Digital Transmission (FL)
  - Each device k aims to send $\Delta \mathbf{w}_i^k$ at the i-th global iteration
  - Adopts spares binary compression with error accumulation as

$$\mathbf{v}_i^k = \mathrm{sparse}_{q_i^k}\left(\Delta \mathbf{w}_i^k + \Delta_i^k\right)$$

where the accumulated quantization error is updated as

$$\Delta_i^{k+1} = \Delta \mathbf{w}_i^k + \Delta_i^k - Q_b\left(\mathbf{v}_i^k\right)$$

  - Then it sends

$$B_{U,k,i}^{FL} = b + \log_2 \binom{W}{q_i^k}$$

bits to send the value $Q_b(\mu)$ and the indices of the non-zero elements of $\mathbf{v}_i^k$, where q_i^k is chosen as the largest integer satisfying $B_{U,k,i}^{FL} \le B_{U,k,i}$

$\mathrm{sparse}_q(\mathbf{u})$

- All elements except the largest q elements and smallest q elements of $\mathbf{u}$ are set to zero
- μ^+ : mean of remaining positive elements
  μ^- : mean of remaining negative elements
- If $\mu^+ > |\mu^-|$, the negative elements are set to zero and the positive elements are set to μ^+
- If $|\mu^-| > \mu^+$, the positive elements are set to zero and the negative elements are set to μ^-

$Q_b(\mathbf{u})$

- Quantizes each non-zero element of $\mathbf{u}$ using a uniform quantizer with b bits per each non-zero element

Appendix: Wireless Cooperative Training

- Uplink Digital Transmission (FD/HFD)
  - Each device k aims to send logit vectors $\mathbf{s}_{i,t}^k$ at the i-th global iteration for all labels $t = 1, \ldots, L$
  - Adopts sparsification and quantization as

$$\mathbf{q}_{i,t}^k = Q_b(\mathrm{thresh}_{q_i^k}(\mathbf{s}_{i,t}^k)) \quad t = 1, \ldots, L$$

  - Then it sends

$$B_{U,k,i}^{FD} = L(bq_i^k + \log_2 \binom{L}{q_i^k})$$

bits to send the non-zero values and the indices of the non-zero elements of $\mathbf{q}_{i,t}^k$ where q_i^k is chosen as the largest integer satisfying $B_{U,k,i}^{FD} \leq B_{U,k,i}$

$\mathrm{thresh}_q(\mathbf{u})$

- Sets all elements of the input vector $\mathbf{u}$ to zero except the q elements with the largest absolute values

Appendix: Wireless Cooperative Training

- Downlink Digital Transmission (FL, FD/HFD)
  - The number of bits that can be transmitted from AP to devices at the i-th global iteration is given using Shannon's capacity

$$B_{D,i} = \min_{k} \left(T_D \log_2 \left(1 + \left| g_i^k \right|^2 P_D \right) \right)$$

  - Satisfying $B_{D,i}^{FL} \leq B_{D,i}$ and $B_{D,i}^{FD} \leq B_{D,i}$, AP compresses and quantizes the corresponding information

Appendix: Wireless Cooperative Training

- Uplink Analog Transmission (FL, FD/HFD)
  - All the devices transmit their information simultaneously in an uncoded manner to the AP
  - Different types of power control at each devices have been studied in the literature, namely full-power transmission, channel inversion [18],[19], and optimized power control [26], [27]
  - In this paper, full-power transmission is considered for simplicity
  - Each device have knowledge of the phase of the channel to the AP, and the AP has full channel state information
  - In analog transmission of a vector, only the values of number of channel uses can be sent (usually much less than the number of network model coefficients)
  - → The gradient update should be sparsfied and compressed into a smaller dimension
  - → The PS recovers the sum of gradient updates by applying AMP (approximate message passing)
  - → It is assumed that the gradient updates have similar sparsity pattern among the devices under the i.i.d. data distribution

Appendix: Wireless Cooperative Training

- Uplink Analog transmission (FL)
  - Each device k aims to send $\Delta \mathbf{w}_i^k$ at the i-th global iteration
  - In order to enable dimensionality reduction, a pseudo-random matrix $\mathbf{A}_U \in \mathbb{R}^{2T_U \times W}$ with i.i.d. entries $\mathcal{N}(0, 1/2T_U)$ is generated and shared
  - Each device k computes and $\mathbf{v}_i^k = \mathrm{thresh}_q \left(\Delta \mathbf{w}_i^k + \Delta_i^k\right)$ for sparsficaton
  - To transmit dimension reduced vector $\hat{\mathbf{v}}_i^k = \mathbf{A}_U \mathbf{v}_i^k$, transmit $\mathbf{x}_i^k \in \mathbb{C}^{T_U \times 1}$,

$$\mathbf{x}_i^k(m) = \hat{\mathbf{v}}_i^k(2m-1) + j\hat{\mathbf{v}}_i^k(2m), m = 1, \ldots, T_U$$

  - Each device k transmits $\gamma_i^k e^{-j\angle h_i^k} \mathbf{x}_i^k \in \mathbb{C}^{T_U \times 1}$, $\gamma_i^k = \sqrt{P_U T_U} / \|\mathbf{x}_i^k\|_2$ for full power transmission
  - The PS scales the received signal by $\nu_i = \dfrac{\sum_{k'=1}^{K} \gamma_i^{k'} \left|h_i^{k'}\right|}{\frac{1}{2} + \sum_{k'=1}^{K} \left(\gamma_i^{k'} \left|h_i^{k'}\right|\right)^2}$

    for minimum mean square error estimate of the sum $\mathbf{A}_U \sum_{k=1}^{K} \mathbf{v}_i^k$
  - The PS applies AMP algorithm to recover $\sum_{k=1}^{K} \mathbf{v}_i^k$

Appendix: Wireless Cooperative Training

- Uplink Analog transmission (FD)
  - Each device k aims to send $\mathbf{s}_{i,t}^{k}$ at the i-th global iteration $t = 1, \ldots, L$
  - Apply repetition coding since L^2 is usually lower than $2T_U$
  - Each device applies repetition coding with the source integer bandwidth expansion factor $\rho = \lfloor 2T_U / L^2 \rfloor \geq 1$
  - And compute

$$\mathbf{v}_i^k = \mathbf{R}_\rho \mathbf{s}_i^k \in \mathbb{R}^{\rho L^2 \times 1}$$

$$\mathbf{R}_\rho = \mathbf{1}_\rho \otimes \mathbf{I}_{L^2}$$

$$\mathbf{1}_\rho = (1, \ldots, 1)^T$$

$$\mathbf{s}_i^k = \left[(\mathbf{s}_{i,1}^k)^T, \ldots, (\mathbf{s}_{i,L}^k)^T \right]^T$$

And transmit as the same way with case of FL

AP multiplies $\mathbf{R}_\rho^T / \rho$ to estimate $\sum_{k=1}^{K} \mathbf{v}_i^k$

Appendix: Wireless Cooperative Training

- Downlink Analog Transmission (FL, FD/HFD)
  - For the downlink broadcast communication from AP to devices,
    - The AP transmits with full power in a same manner of each device at the uplink
    - Each device applies a scaling factor and the AMP algorithm in order to estimate the vector transmitted by the AP, in a similar manner of AP at the uplink