Sepehr Goshayeshi
March 27, 2020

Comprehensive Summary 2: Energy Efficiency of Federated Learning for Internet of Things

Before discussing about the topic, it is important to define a few terms and explain some concepts. Federated learning involves using *edge devices* to do machine learning *without centralizing data* while not compromising *privacy* [1]. An *edge device* refers to any network component which connects from one's local area network to external networks, and it provides local information to an external network [2]. Edge devices can range from smartphones, IoT sensors connected to industrial equipment, or entities like hospitals. IoT refers to the interrelated web of edge devices, which transfer the data over the network. Federated learning allows you to build a model on the base server of the IoT while keeping the data at its source, and each device trains its model locally and shares it with the base server, rather than the data. The base server combines the model into a federated model without centralizing data or having direct access to one's private training data [2].

In summary, there is a model in the base server which is distributed to some of the clients, and the model is trained using the clients' own local data. It can be trained a little bit and not necessarily towards convergence [3]. Regardless, a client produces a new model that's locally trained and sends local learning model parameters to the base server [3-4]. The base server gets the locally trained model updates from all of the clients, and these updates are aggregated and averaged out. To be more precise, their gradients are averaged by the server, and the server broadcasts the local models to be updated [4]. The collaborative model becomes the initial model in proceeding rounds, which improves in succeeding cycles [3]. It is decentralized since data is not stored in one location, and it also a collaborative process of training collective data locally and building shared models [4]. Local training results are transmitted via wireless links [4]. This can affect the performance of federated learning due to time, bandwidth, and limited energy of wireless devices.

Despite all the encouraging benefits, there are still some obstacles to the feasibility of federated learning. When training on mobile devices, ensemble learning can be expensive because the entire learning process requires multiple data exchanges between the central server and the device before the entire model converges [5]. In terms of energy consumption, training on the device is very energy-intensive and reduces the battery life of the mobile device. However, this powerful assumption contradicts the original purpose of analyzing data on mobile devices, namely to keep abreast of the data collected everywhere [6]. Real-time machine learning is becoming increasingly important in time-critical applications such as autonomous vehicles, military applications, medical informatics, business analytics, and rapidly changing environments. In addition, many time-sensitive applications can benefit from real-time mobile collaborative learning. In terms of completion time, heterogeneity in the federated system can also have a significant impact on the energy efficiency of the learning process.

With the explosive growth of smart IoT devices at the Internet edge, mass data collection using sensors built into mobile devices (such as crowd recognition) has found many machine learning applications and has quickly become very popular. However, existing machine learning methods are based on centralized storage of training data. As a result, they often face many data security and confidentiality issues, including data misuse and information leakage [7]. One reason federated learning was introduced was to address issues with data confidentiality. The mobile device computes the model training locally based on the training data published by the model owner. The design allows mobile devices to collaborate on collaborative predictive models and store all training data on the device.

With FL, wireless devices can collaboratively perform training tasks by simply loading the parameters of a local training model into a base station (BS). It doesn't need to share all the training data. Implementing FL over a wireless network requires wireless devices to send federated learning results over a wireless channel [8]. Due to the limitation of wireless resources (time, bandwidth, etc.), this may affect FL performance. Also, the energy limitations wireless devices is an important consider when deploying FL. Due to such constraints in resource, it is required to optimize energy for the implementation FL. However, independent and efficient mobile devices need incentives to participate in collaborative learning. In fact, accessing mobile devices as victim volunteers is not an economically viable and sustainable option. In addition, the ensemble learning paradigm still requires a direct link between the model owner and the mobile device to send model updates [9]. In many cases, direct communication may not be available because of limited transmission range and high transmit power can prevent efficient use of energy.

The strict low latency and confidentiality requirements of expensive new applications for smart devices such as drones and smart vehicles make cloud computing unsuitable for these situations. In its place, federated machine learning has become increasingly attractive for training and direct output to the edge of the network without sending data to a central data center [10]. This has inspired a new field called ensemble learning, which trains federated machine learning models on computing, storage, energy, and mobile devices with limited bandwidth in a distributed manner. In order to maintain the confidentiality of data and solve the problem of unbalanced data points and non-IID points on different devices, a joint for ensemble models is proposed globally by calculating the weighted average of locally updated models on each selected device averaging algorithm. However, limited communication bandwidth is a major bottleneck for converging updates for local computing. Therefore, new method based on wireless computing, which quickly summarizes the global model by examining the superimposed characteristics of wireless multiple access channels [11]. This can be achieved through joint device selection and beam forming design, modeled as sparse low-level optimization tasks to support efficient algorithm development. To this end, it provides a representation of the difference between sparse and low-rank convex functions (DC) to increase sparsity and accurately define fixed rank constraints during the device selection process. A “difference of convex functions” (DC) algorithm was developed to solve such programs with global convergence. Extensive results demonstrated the algorithmic advantages and features of the method [12].

The research [13] is specifically aimed at optimizing the performance of ensemble learning algorithms, such as training time and energy costs. However, most existing work makes an optimistic assumption that all mobile devices are unconditionally invited to participate in integrated learning. This is impractical in the real world due to the resource costs associated with training models. Without careful financial compensation, mobile devices of their own interest will hesitate to participate in joint learning. There are two limitations to creating wireless relay networks between mobile devices and enabling energy-efficient communication for sending model updates. The second set of constraints is related to the arrival time of the model update at the relay point. For routing, there were initially some restrictions that allowed each mobile device to connect to the model update and send it only to one of the other mobile devices or to the access point of the model owner. Second, there is a limitation that at least one mobile device connects to the model owner's access point and acts as one of the nodes of the last transition. Otherwise, no mobile device can forward model updates to the model owner.

The rising cost of healthcare, coupled with a growing interest in patients who do not go out of the clinic, has created an urgent need for innovative healthcare systems. This has led to creating FL solutions that help healthcare providers to monitor and assess patients' health. However, such existing work appears to lack energy efficiency. This article [14] delves into the structure of a comprehensive patient health monitoring (PPHM) system. PPHM is based on integrated cloud computing and IoT technologies. The authors propose a remote pervasive PPHM. The proposed structure combines the powerful synergy of IoT, and wireless technologies to enable efficient, high-quality remote monitoring of patient health. A case study using ECG to monitor patients with congestive heart failure in real-time to demonstrate the applicability of the proposed PPHM infrastructure. PPHM is very energy-efficient for remote patient health monitoring systems [14]. Experimental results validate this [14].

This article [15] discusses the allocation of energy-efficient transmission resources and federated learning (FL) calculations in wireless networks. In this model, each user uses limited local computing resources to use the collected data to train a local FL model and sends the trained FL model parameters to the base station (BS). The base station (BS) combines the local FL model and forwards it to all users. Since FL involves the exchange of learning models between users and BS, the computational delay and communication depend on the level of learning accuracy. On the other hand, due to the limited energy budget of wireless users, the FL process needs to consider both the local calculation energy and the transmission energy [15]. The common learning and communication problem is to consider the trade-off between FL delay and FL energy consumption, minimize FL completion time, locally calculated energy and the weighted sum of transmission energy for all users. It is framed as a kind of optimization problem; an iterative algorithm is proposed to solve such a problem. At each stage, the algorithm will produce a closed solution for time allocation, bandwidth allocation, power control, calculation frequency, and training accuracy. In the special case of minimizing the completion time, an algorithm is proposed that divides the algorithm into two halves to obtain the best solution. The numerical results show that compared with the traditional FL algorithm, the algorithm can save 25.6% of waiting time and 37.6% of energy consumption [16].

Edge devices are usually powered by batteries of limited size. Power limitation is another issue, especially when the trained FL model contains a large number of parameters and the computation and communication load is high. In addition, federal border training also faces the so-called Trump dilemma. In other words, the effectiveness of machine learning training is limited by the slowest boundary devices in communication and computing. This document [17] describes a joint edge training system consisting of an edge server and multiple peripherals. The edge server trains the shared ML model using distributed packet gradient descent (BGD) by adjusting the edge server. It is assumed that training with FL models follows certain requirements for training delay. Our goal is to load and aggregate global ML parameters as well as local updates of computational resources (ie, central frequency (CPU) frequency) FL parameters [17]. In particular, it shows the communication and computational energy consumption models as a function of local and global iterations, as well as the communication and computational load for each iteration.

The document further elaborates upon two transport protocols on edge devices, which help uploading local model parameters to the edge server: non-orthogonal multiple access (NOMA) and time division multiple access (TDMA). According to these two transmission protocols, the proposed energy minimization problem is non-convex and is often difficult to solve. The convex surfaces and an effective algorithm to obtain the optimal solution. The numerical results show an interesting trade-off between energy consumption, learning speed, and learning accuracy [18]. Compared to other test methods that do not perform this joint optimization, the proposed joint design of communication and computing has shown significant performance improvements. It has also been shown that choosing the correct number of global and local iterations can effectively balance the trade-offs between communication calculations and further improve the energy efficiency of the system.

References

[1] “Federated learning: machine learning on decentralized data (Google I/O'19),” *Youtube*, https://youtu.be/89BGjQYA0uE, 2019.

[2] F. Andrade, “What is an edge device, and why is it so crucial to IIoT?,” *Netilion Blog*, https://netilion.endress.com/blog/what-is-edge-device-iiot/, 2019

[3] “TensorFlow Federated (TFF): Machine Learning on Decentralized Data (TF Dev Summit ‘19),” *Youtube*, https://youtu.be/1YbPmkChcbo, 2019.

[4] Z. Yang, M. Chen, W. Saad, C. H. Hong, and M. S. Bahaei, “Energy efficient federated learning over wireless communication networks,” A*rxiv,* https://arxiv.org/pdf/1911.02417.pdf, 2019.

[5] Q. Yang et al., “Federated machine learning: Concept and applications,” ACM Transactions on Intelligent Systems and Technology (TIST), vol. 10, no. 2, pp. 12:1–12:15, 2019.

[6] J. X. Xiaopeng Mo, "Energy-Efficient Federated Edge Learning with Joint Communication and Computation Design," *https://arxiv.org/abs/2003.00199,* 2020.

[7] Sumudu Samarakoon, "Distributed Federated Learning for Ultra-Reliable Low-Latency Vehicular Communications," *IEEE TRANSACTIONS ON COMMUNICATIONS,* vol. 68, no. 2, pp. 1146-1159, 2020.

[8] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, “Mobile unmanned aerial vehicles (UAVs) for energy-efficient Internet of Things communications,” IEEE Trans. Wireless Commun., vol. 16, no. 11, pp. 7574–7589, Nov. 2017

[9] Z. Hou, H. Chen, Y. Li, and B. Vucetic, “Incentive mechanism design for wireless energy harvesting-based internet of things,” IEEE Internet of Things Journal, vol. 5, no. 4, pp. 2620–2632, 2018.

[10] P. Popovski et al., “Wireless access for ultra-reliable low-latency communication: Principles and building blocks,” IEEE Netw., vol. 32, no. 2, pp. 16–23, Mar./Apr. 2018

[11] K. Bonawitz et al., “Towards federated learning at scale: System design,” 2019, arXiv:1902.01046. [Online]. Available: https://arxiv.org/abs/1902. 01046

[12] C. Lu and Y.-F. Liu, “An efficient global algorithm for single-group multicast beam forming,” IEEE Trans. Signal Process., vol. 65, no. 14, pp. 3761–3774, Jul. 2017

[13] T. J. Y. S. Z. D. Kai Yang, "Federated Learning via Over-the-Air Computation," *IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS,,* vol. 19, no. 3, pp. 2022-2035, 2020.

[14] M. M. H. Jemal H. Abawajy, "Federated Internet of Things and Cloud Computing Pervasive Patient Health Monitoring System," *Impact of Next-Generation Mobile Technologies on IoT: Cloud Convergenc.*

[15] Y. Mao, J. Zhang, and K. B. Letaief, “Dynamic computation offloading for mobile-edge computing with energy harvesting devices,” IEEE J. Sel. Areas Commun., vol. 34, no. 12, pp. 3590–3605, Dec. 2016.

[16] M. C. W. S. Zhaohui Yang, "Energy Efficient Federated Learning Over Wireless Communication Networks," *https://arxiv.org/,* 2019.

[17] X. Cao, F. Wang, J. Xu, R. Zhang, and S. Cui, ”Joint computation and communication cooperation for energy-efficient mobile edge computing,” IEEE Internet Things J., vol. 6, no. 3, pp. 4188-4200, Jun. 2019.

[18] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, “Energy-optimal mobile cloud computing under stochastic wireless channel,” IEEE Trans. Wireless Commun., vol. 12, no. 9, pp. 4569-4581, Sep. 2013.