Federated Learning in Network Operations

State of the Art Research and Future Use Cases

Contents

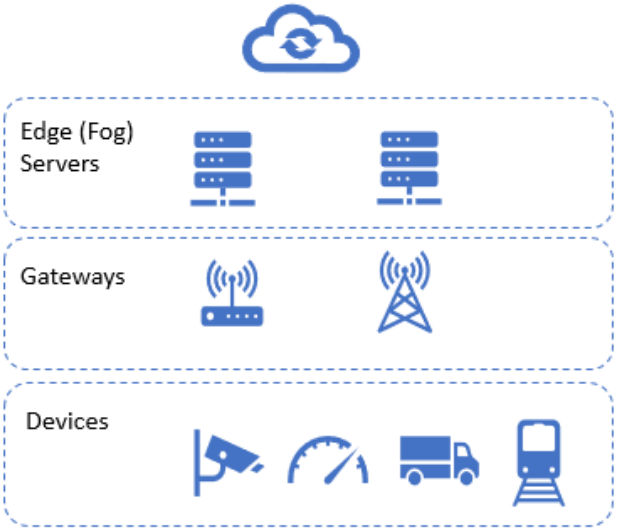
Introduction

The application of Artificial Intelligence (AI) in today's technology is becoming widespread. There have been very successful AI applications that are already household names (Siri, Alexa, Nest, Ring, etc.). However, there is a common dilemma faced by the application developers: the innovation of AI often means the sacrifice of the user data privacy. The very limited availability of high-quality training data has become one of the biggest bottlenecks for developing innovative AI applications.

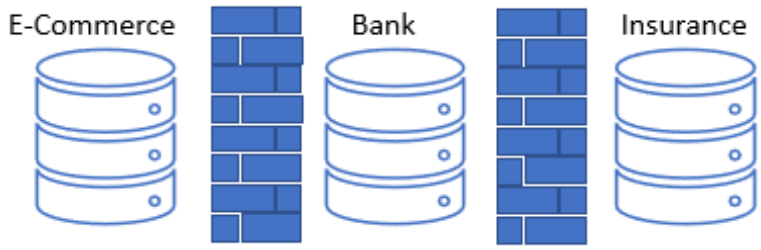
There are several common scenarios for the training data availability issue:

- Data on consumer devices
  Consumers are becoming more and more conscious about the data privacy on their smart devices. Although lots of data is valuable in training a deep learning model, many customers do not give the permission to upload that data to cloud side.

- Edge computing
  In many edge computing applications like self-driving car, unmanned aerial vehicle, industrial control, the round trip between edge and cloud cannot satisfy the stringent real-time constraints. And it is common that the edge devices have very limited power and transmission bandwidth, which makes uploading large amount of data to cloud not feasible.



- Data silos between market players
  Some industries have a large set of common customers. For example, a customer has a mortgage at a bank and car insurance at an insurance company. These industries can develop new innovative applications if the data can be shared across the companies. However, the data privacy laws (e.g. EU GDPR) do not allow uncontrolled transfer of consumer personal data to third party. In practice, there are many data silos formed containing consumer data, but they cannot be utilized effectively for new applications.



There have been some early research efforts aiming to strike a good tradeoff between data privacy and model training, such as [Shokri 2015], however they did not catch enough interest from the academia and industry. the Federated Learning (FL) was introduced in 2016 by Google to help preserving the data privacy in deep learning neural networks. It was applied in the Google Keyboard (Gboard) app to predict the upcoming words intended by users as a typing aid. Between 2016-2020, Federated Learning has been a hot research topic in academia. Multiple security enhancements were proposed to protect data privacy, and many research papers were published reporting Federated Learning in various application areas.

In this article, we discuss the application of Federated Learning in network operations. First the basics of Federated Learning is presented including its motivation, brief history, and fundamental algorithm. Next, we surveyed the recent academic research efforts related to the network operations especially in 5G networks and edge computing. In third section, we propose several use cases in network operations that can benefit from the application of the Federated Learning. Lastly, we discuss the challenges faced by industry to successfully apply Federated Learning to practical applications.

Basics of Federated Learning

The fundamental advantage of Federated Learning over traditional learning is to make more training data available, which is done by decoupling the training data from the training location. In Federated Learning, the training data is kept local at a participating device, not required to by uploaded to cloud or a central server. The traditional deep learning process is shown in Fig. 1, where the training data is first uploaded to the cloud, then training happens at the cloud, and the resulted model is downloaded to devices.

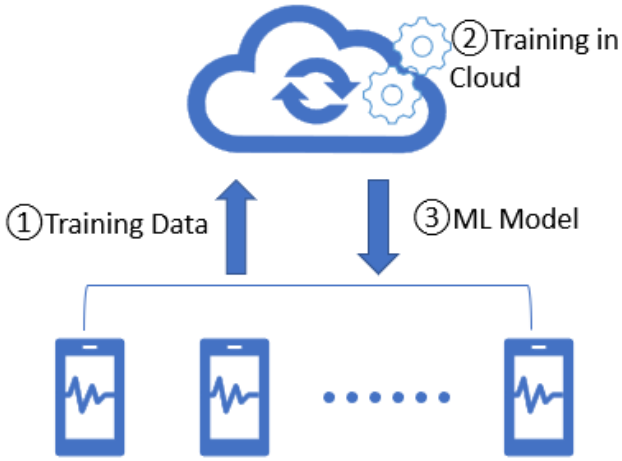


Fig. 1 Traditional Deep Learning Process

In Federated Learning shown in Fig. 2, the local training is performed at participating devices using local data. Then the local model is uploaded to the cloud where the model aggregation is performed, and then the resulted common model is downloaded to the devices. Note the local training phase can be bootstrapped from a pretrained common model to improve the system performance. Also, after the Federated Learning process is complete, the resulted model can be used by other devices that did not participate in the training stage.

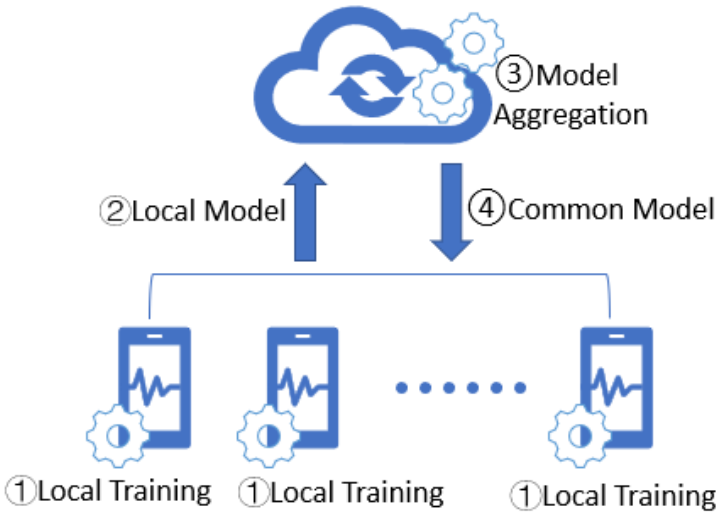


Fig. 2 Federated Learning Process

A high-performance FL system is largely dependent on the model aggregation algorithm. A simple but surprisingly working well algorithm shown in Fig. 3 was proposed by Google researchers [McMahan 2017]. In this algorithm, initially a pretrained common set of model weights (W_t) is

downloaded to the participating devices. Then there are multiple iterative rounds of training happening at participating devices and weight averaging at the cloud. At the end of each round, the new common weights are downloaded to the devices again as the starting point for the next round.

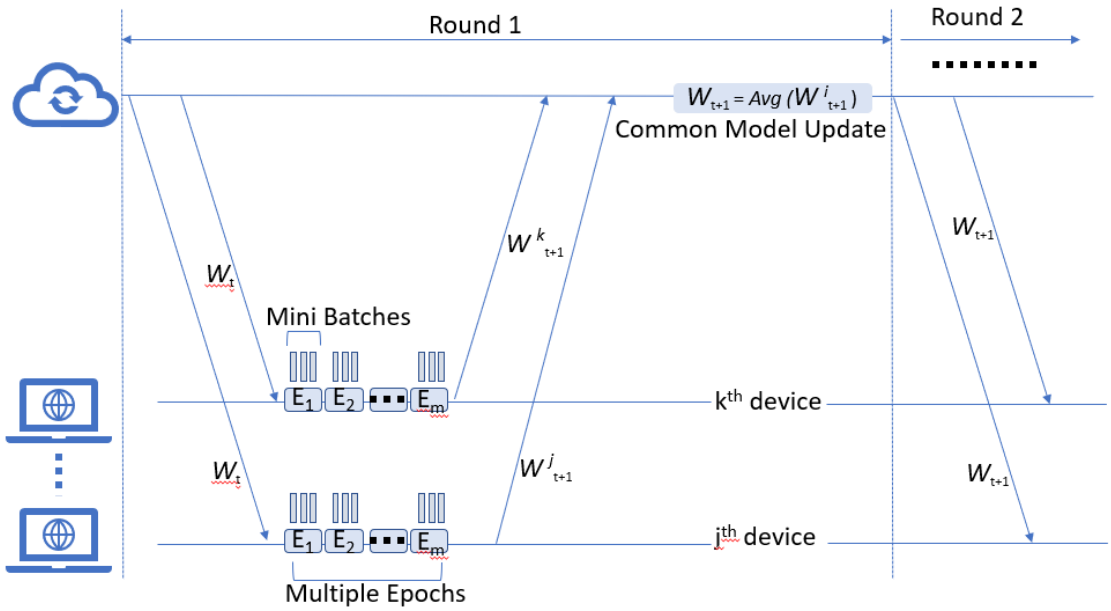


Fig. 3 Federated Averaging (FedAvg) Algorithm

A Survey of Recent FL Research Efforts in 5G and Edge Computing

Since 2016, Federated Learning has become an increasingly active research topic, covering different aspects of FL. There have been papers published in fundamental aspects of FL, such as enhancement of data security with differential privacy and secure multiparty computation (SMC) [Geyer 2017] [Truex 2019], design of production systems [Bonawitz 2019], and communication compression scheme [Felix 2019], etc.

In this article, we mainly focus on the application of FL in 5G network and edge computing. Fig. 4 lists some of the FL research topics found in recent literature, which is by no means comprehensive. We selected several representative research papers that we think are of interest to network operations community.

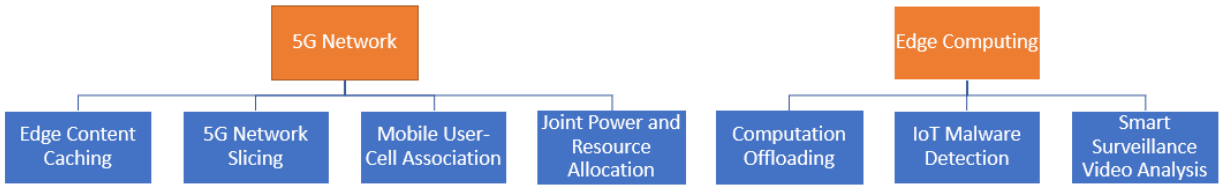


Fig. 4 Federated Learning Research Topics in 5G and Edge Computing

5G Networks

Content Caching

Multimedia content (especially video files) caching is a very effective approach to reduce the load on mobile backhaul network and cloud data centers. However, the content popularity depends on user preferences and varies over time, which poses a challenge to the caching policy design. There has been intensive research aiming to design proactive caching algorithms, such as the works by [Bastug 2014] and [Qiao 2016]. These approaches however require the server to collect users' context information e.g. file request history, location, velocity, and mobility patterns, which will become a data privacy concern.

Federated Learning is a natural choice to preserve the user data privacy when learning for caching decisions. FL is applied to content caching at mobile edge by [Yu 2018], [Wang 2019], and [Qi 2020]. The system model of this approach is shown in Fig. 5. The federated learning is formed between user devices and base station, where user devices performs training on local data and base station performs model aggregation and makes caching decisions based on the common model. When cache miss happens, the original multimedia file is downloaded from cloud.

Fig. 5 Mobile Edge Caching with Federated Learning

The cache efficiency (ratio of cache hits to the number of requests on the cache) of FL based algorithm is quite good. Fig. 6 shows the performance of the FPCC algorithm presented in [Yu 2018] compared with other algorithms, using MovieLens 1M dataset [Harper 2015]. Here the

"Oracle" line represents an algorithm that has the perfect knowledge about the future demands, so it provides the best possible cache efficiency. The FL-based FPCC algorithm gives a performance very close to the "Oracle" algorithm.

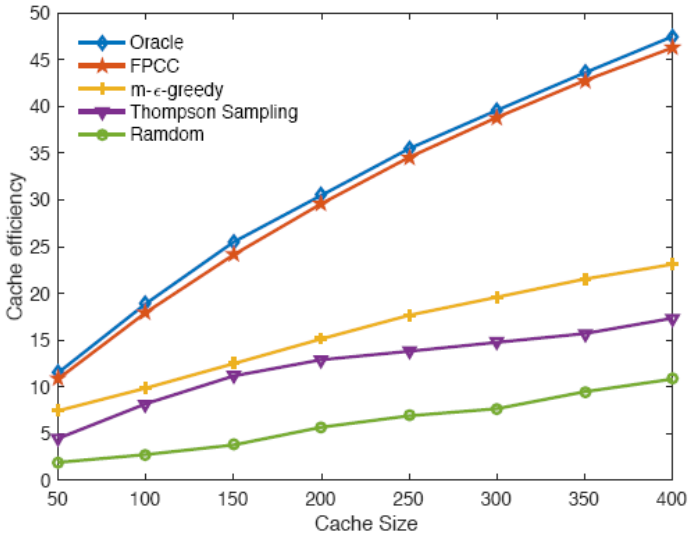


Fig. 6 Mobile Edge Cache Efficiency Comparison [Yu 2018]

5G Network Slicing

Network slicing is an enabling technology in 5G networks to provide end-to-end resource isolation and increased statistical multiplexing [Ordonez-Lucena 2017]. Each network slice represents an independent virtualized network and allows operators to provide customized functions to meet QoS requirements of different use cases cost-effectively. To make network slicing practical for network operators, an important issue is how to allocate network resources efficiently to individual slices. The goal of such allocation policy is to maximize the network operator profit with the constraints of SLAs of each slice.

There has been a fair amount of research effort devoted to this topic, such as [Sciancalepore 2017] and [Huynh 2019]. The approach taken by [Sciancalepore 2017] first performs user traffic forecast with Holt-Winters method to cope with the seasonal effect and the SLAW mobility model [Lee 2012] to handle the effect of user mobility. Forecasted traffic patterns is then used to predict demands to the slices and thus maximize the system resource utilization via admission control. In [Huynh 2019], instead of relying on time series forecast, a Q-learning based Deep Dueling technique is used to converge to the optimal policy after a finite number of iterations between the slicing manager and the network.

[Fantacci 2020] formulates the network slicing problem as such, "how many VNFs to place in different network slices to maximize the service provider's revenue"? First, Federated Learning was used to forecast the user demand to network slices (Fig. 7). Each slice runs federated learning with its respective model aggregator in the network slice controller. The final models of all slices are used in the demand prediction module for the forecast. Compared with [Sciancalepore 2017], using FL here reduces the privacy concerns of collecting user traffic statistics and mobility. In addition to FL, prospect theory (*https://en.wikipedia.org/wiki/Prospect_theory*) was used to model user decision making process. Once demand forecast and user behavior models are

available, a heuristic algorithm based on matching theory was designed to place the VNFs in network slices. The effect of forecast with FL module is evaluated with numerical study, with the results shown in Fig. 8. The profit of service provider has a difference of around 20% with and without FL Module.

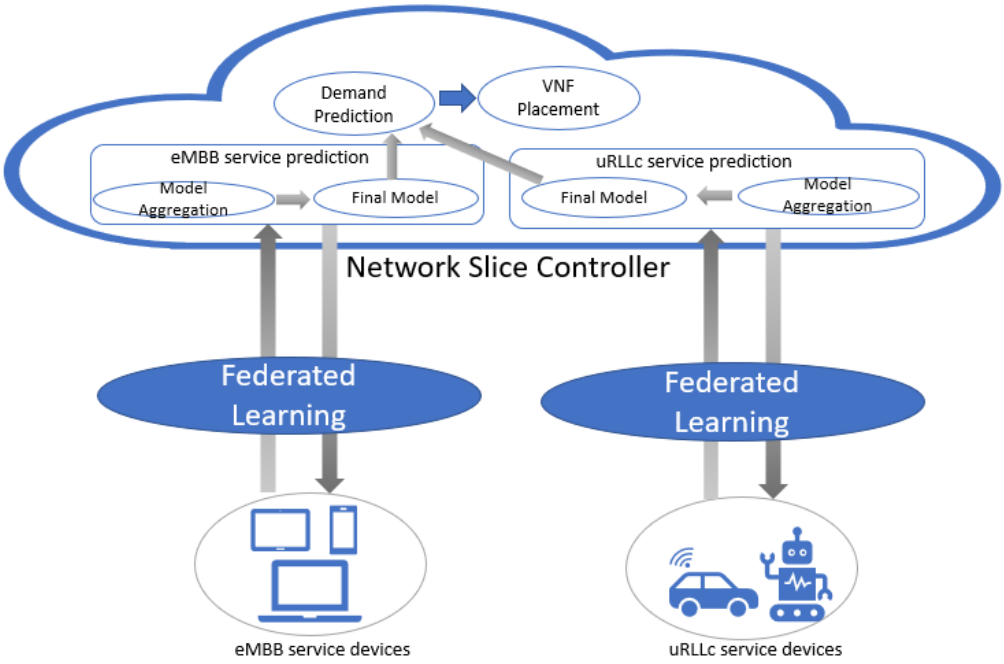


Fig. 7 Federated Learning in 5G network slicing [Fantacci 2020Sciancalepore 2017]

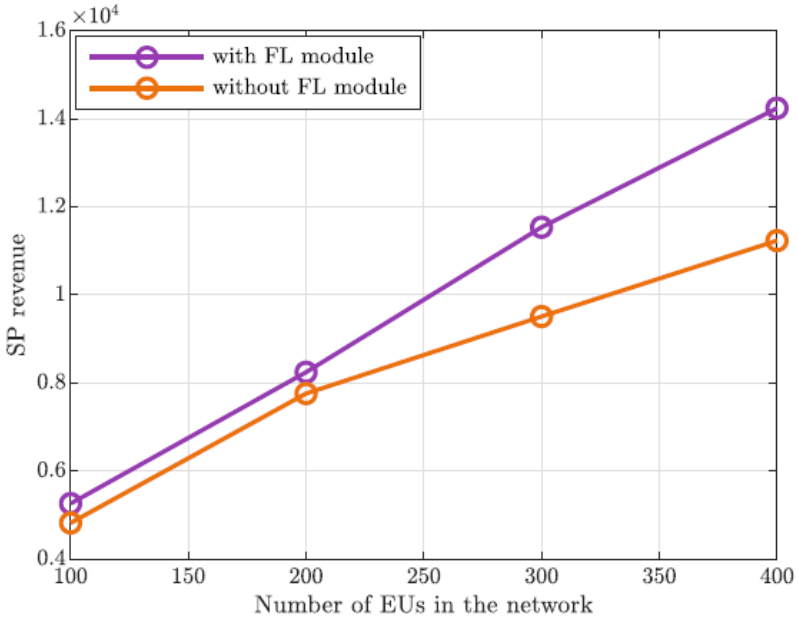


Fig. 8 Network Service Provider Profit with and without FL Module [Fantacci 2020Sciancalepore 2017]

Although the works of [Yu 2018] and [Fantacci 2020] target different domains, they have high similarity in terms of how Federated Learning is applied. Both use FL to utilize the user demand data to achieve more efficient system design without sacrifice user privacy.

Mobile User-Cell Association

In 5G networks, the use of mmWave frequencies can add capacity for Enhanced Mobile Broadband (eMBB) services and reduce the transmission delay for Low-Latency Communication (uRLLC) services. However, mmWave communication has a higher near-field path loss and channel uncertainty compared to existing wireless systems using sub-6 GHz, μW frequency bands. For 5G networks to efficiently service both eMBB and uRLLC scenarios, it is necessary to support the deployment scenario in which mmWave co-exists with sub-6GHz frequency [Semiari 2019]. Fig. 9 gives an example scheme to realize separate RRU functions but shared Layer 2 and Layer 3 functions for the mmWave and sub-6 GHz radio access technologies.

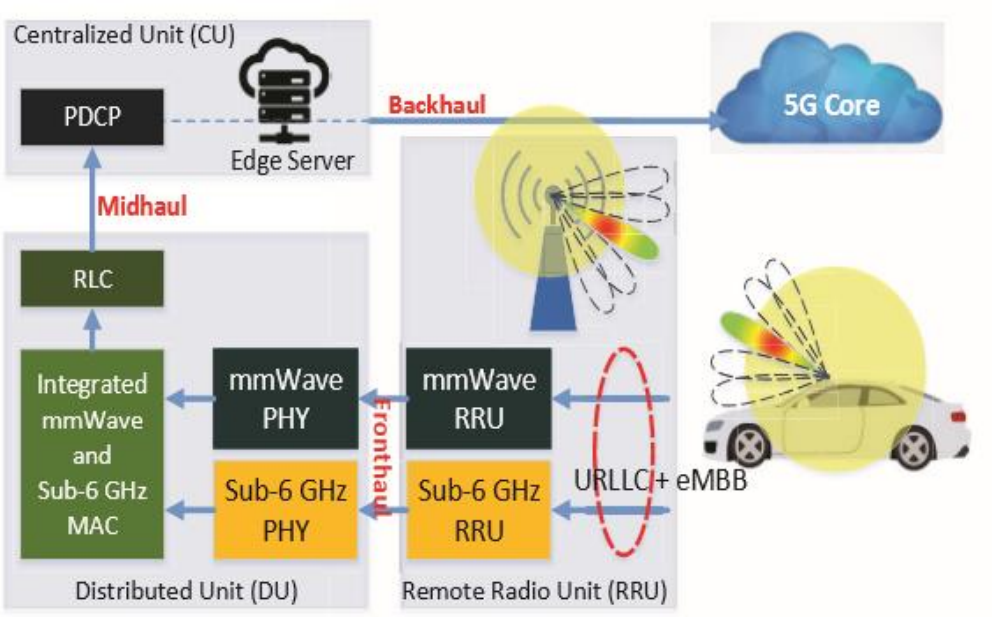


Fig. 9 Example hybrid mmWave and sub-6 GHz system [Semiari 2019]

In such hybrid systems, the mobile user should be able to associate with two types of base stations on both the uplink and downlink within a network of ultra-dense small cells. An important problem faced by network operator is how to minimize the average network delay under any arbitrary spatial distribution of the mobile users as well as the optimal cell partition of base stations. The optimal solution to this problem helps network to achieve efficient use of spectrum & energy, reduced interference, and maximized capacity, thus has a high impact on user experience and operator CapEx.

In [Elshaer 2016], the authors developed a general analytical model to characterize the uplink and downlink cell association and showed advantages of decoupled uplink and downlink cell association strategies. A distributed algorithm based on the Echo State Networks (ESNs) machine learning framework is proposed in [Chen 2017], in which the base stations could choose their

optimal resource allocation strategies given information on the network and mobile terminals. [Waheidi 2019] proposed a distributed multiclass cell association algorithm based on game theory to associate devices with different service characteristics.

The above solutions assume base stations having the capability to share desired user data between each other, such as user movement pattern and traffic load. The federated learning is used in [Chen 2020] to develop an algorithm for user-cell association to enhance mobile user experience, specifically minimizing breaks in presence (BIP) within VR applications. It is observed that the user movement pattern data is dispersed across multiple BSs, and it's not practical for the base stations collectively maintain a synchronized copy of collected user data due to the transmission cost and processing delays.

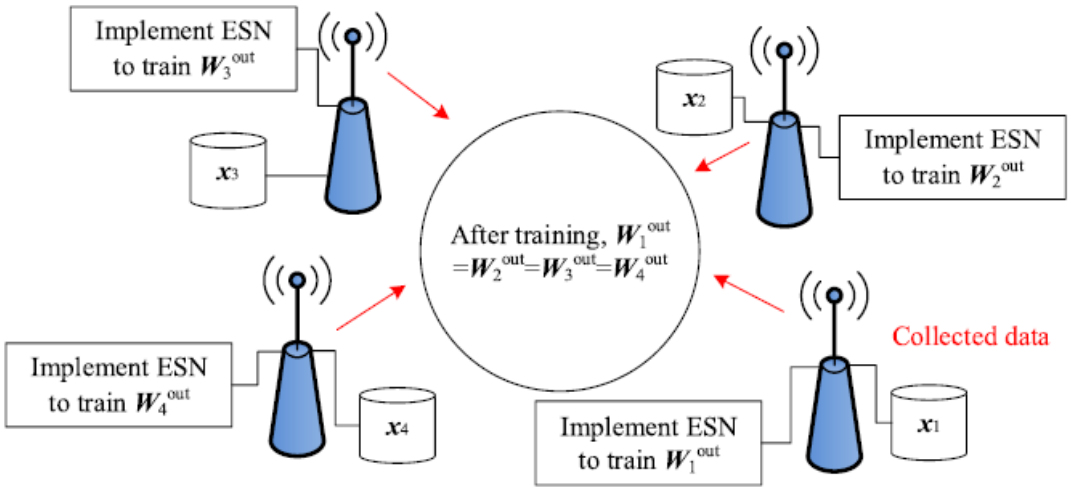


Fig. 10 Federated Learning to train ESN model [Chen 2020]

A federated learning algorithm was introduced to train a machine learning model (ESN) for user mobility forecast. The idea is shown in Fig. 10, where BSs train local models using the data collected by themselves, then BSs can cooperatively build a learning model by sharing their trained models to each other. Finally, the user-cell association is determined by the forecast of user mobility given by the collectively trained model.

The numerical result shows that the FL-based algorithm can achieve up to 16% reduction in the number of BIPs compared to baseline algorithm.

Edge Computing

The trend of IoT-cloud convergence is picking up speed with most of the IoT platforms supported with cloud computing. However, the QoS requirements of the applications

demanding low-latency, real-time operation and seamless wireless coverage cannot be met with existing cloud computing platforms.

Edge computing extends the cloud computing functions to the edge of the network. Edge nodes are deployed close to the end-users and geographically distributed, therefore they can better support the real-time, mobile applications/services. However, edge nodes usually have stringent limitations on their available computing, communications, storage and energy resources.

Today's IoT applications' requirement of processing massive amount of collected data pose a great challenge to the existing edge computing frameworks. In addition, many IoT devices collect and store privacy-sensitive information of their users, e.g. health, home automation, shopping history, calendar, etc. It's difficult to utilize those data without violating privacy protection regulations. Therefore, there is a clear need to investigate the usage of federated learning in edge computing to process massive amount of IoT data in a secured manner. In this section, we review several recent research efforts of federated learning in the area of edge computing.

IoT Malware Detection

The number of IoT devices is quickly growing due to the emerging new IoT applications in home and industrial use cases. However, it's not uncommon that the IoT devices have inherent security vulnerabilities due to several reasons: strict development timeframe, lack of standard security solution, and limited over-the-air update channel, just name a few. Some malwares are specially designed to target at IoT devices, such as Mirai botnet [Kolias 2017], to launch DDoS attacks.

Recent research in [Hafeez 2016] and [Miettinen 2017] was aimed to develop mechanisms to defend against these attacks. [Hafeez 2016] proposed a two-tier structure, in which a lightweight gateway calls a cloud service to detect malicious traffic. [Miettinen 2017] utilized traffic traces captured during IoT device onboarding process, to identify device manufacturer and model. This information is then used to set up access control for these IoT devices.

The challenge to the above approaches is that new IoT devices are released very frequently, therefore it's hard to train a precise traffic profile covering all behaviors of various devices. Also, some IoT devices do not generate a lot of traffic to the server, and these techniques fall short in detecting attacks with sporadic network activity. [Nguyen 2019] proposed an autonomous self-learning distributed system called DÏoT for detecting compromised IoT devices. Anomaly detection models are learned using a federated learning approach through the cooperation between security gateways and the cloud. The system architecture of DÏoT is shown in Fig. 11, where the security gateways use locally collected data to train local models, while IoT Security Service on the cloud aggregates the uploaded local models into a global model.

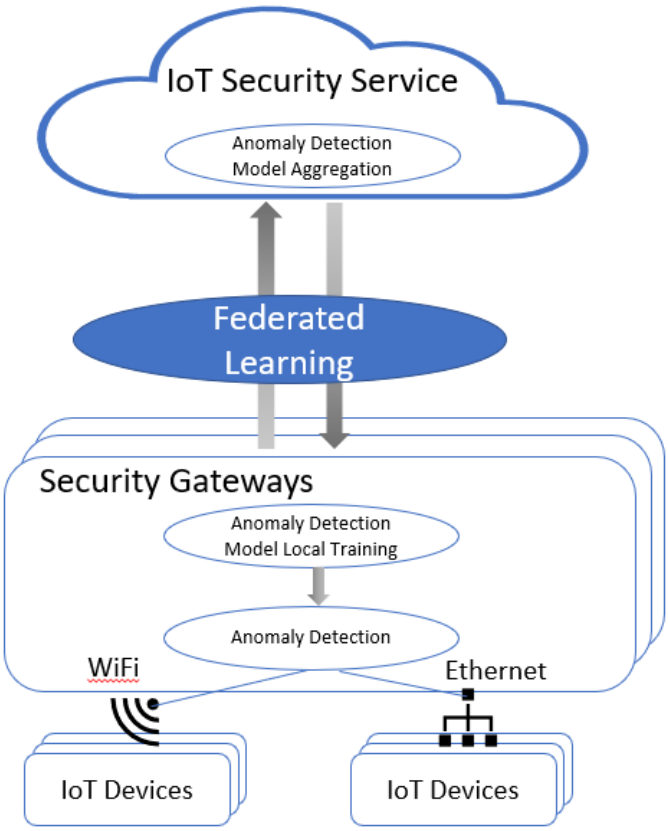


Fig. 11 DÏoT [Nguyen 2019] Architecture

Experimental study of DÏoT tested with more than 30 off-the-shelf IoT devices, showing a detection in 257ms and effective 95.6% true positive rate, with zero false alarms.

Smart Surveillance Video Analysis

Smart surveillance video analysis is an important application in smart city solutions. It's enabled by the integration of real-time high-definition video capturing, reliable high-speed wireless network, online streaming analytics, and machine learning algorithms. It allows near real-time emergency response to traffic conditions, accidents, and high-risk infrastructures.

Despite the benefits of smart surveillance video analysis systems, some challenges need to be addressed. First, the vast data volume collected from surveillance cameras will put high pressure on the network and computing infrastructure. Secondly, some video content may be privacy-sensitive, and there is a high security risk in case attackers manage to break in the central server.

A distributed video analytics architecture based on federated learning was proposed in [Sada 2019], the structure of which is shown in Fig. 12. In this architecture, the federated learning is achieved between the cooperation between the individual cities and cloud. The local learning is

performed in local data centers within participating cities and the global model aggregation module is located at the cloud end.

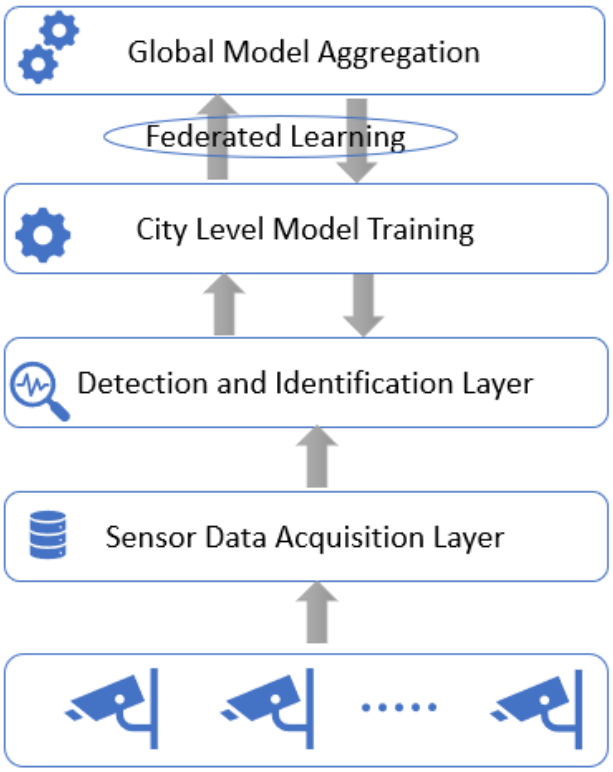


Fig. 12 Distributed Video Analytics Architecture [Sada 2019]

Computation Offloading to Edge and Fog Nodes

To cope with the issue of edge nodes only possessing limited computing, storage and energy resources, the computation-intensive tasks can be offloaded from IoT devices to edge servers. Since edge nodes' communication and computing capacities are limited, many MUs offloading to the same edge node at the same time will lead to resource contention. This scheduling problem is a hard optimization problem and has been studied by the research community. There have been various ways to formulate it, as a mixed integer programming (MIP) [Chen 2016], as a game theory problem [Chen 2018], or as a reinforcement learning problem [Dinh 2018].

The authors in [Ren 2019] proposed a federated learning-based computation offloading scheme. In this scheme, multiple deep reinforcement learning (DRL) agents deployed on edge nodes to indicate the decisions of the IoT devices. Federated learning is used in a hierarchical manner, both 1) between edge nodes and the cloud and 2) between the IoT devices and the edge nodes. The architecture of this scheme is shown in Fig. 13.

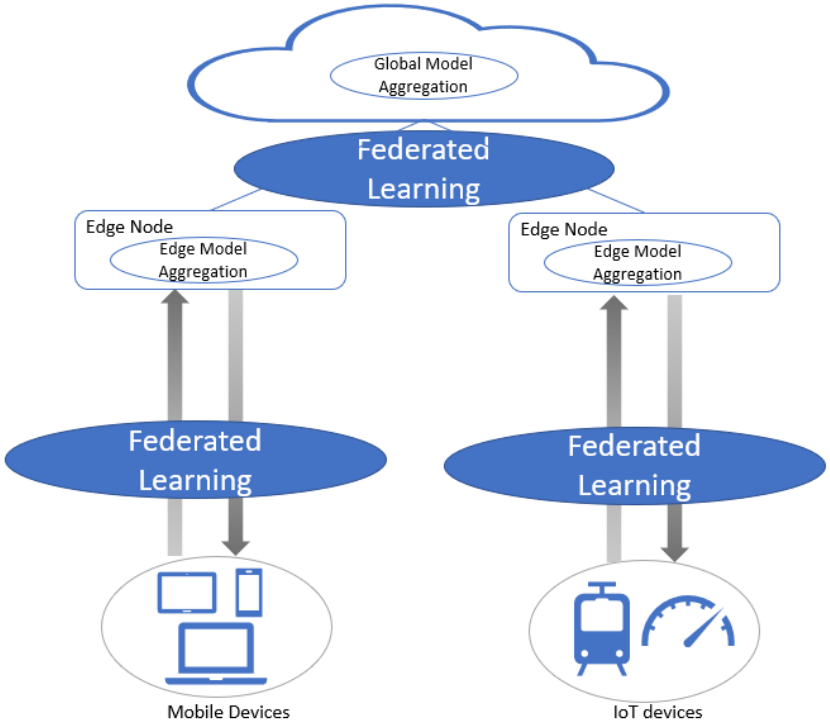


Fig. 13. Computation Offloading at Edge with Federated Learning [Ren 2019]

In contrast to the offloading from IoT devices to edge server, the offloading from edge server to fog node is studied in [Qian 2019]. In this scheme, edge nodes perform lightweight learning using local data and the fog nodes perform federated average to update the common model. The numerical results show that the model accuracy is not sacrificed much by reducing the training data at fog nodes by using federated learning.

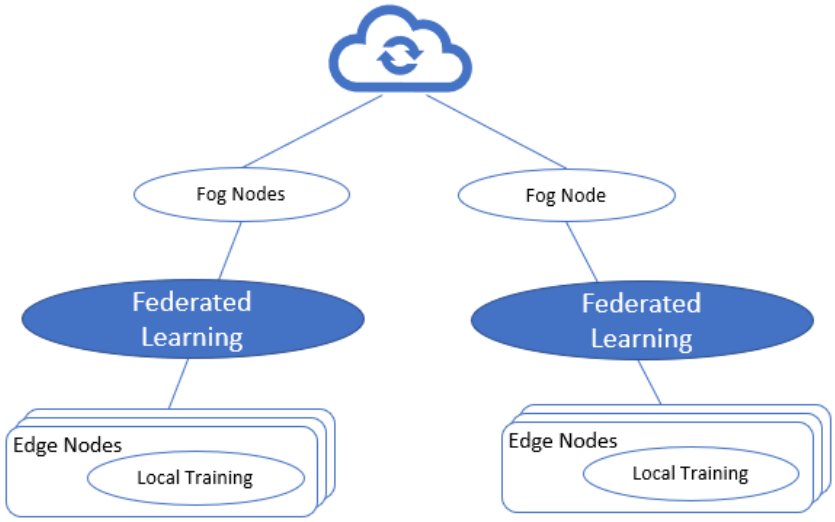


Fig. 14 Computation Offloading at Fog Nodes with Federated Learning [Qian 2019]

Use Cases of Federated Learning in Network Operations

In the next few years, the network complexity will increase dramatically due to the upcoming 5G deployment. 5G networks will most likely co-exist with 2G/3G/4G networks for some time, which will bring a significant challenge for the service providers. Network automation is considered by many mobile network operators (MNO) as the only way to handle this new level of network complexity, and the goal is to achieve simplified network deployment, OPEx optimization and improved user experience and service agility. According to a survey of 76 mobile operators worldwide by Analysis Mason (shown in Fig. 15), more than 50% of MNOs globally have little or no automation in their networks in 2018. By comparison, by 2025 about 60% MNOs expect to have automated 60% or more of their network operations.

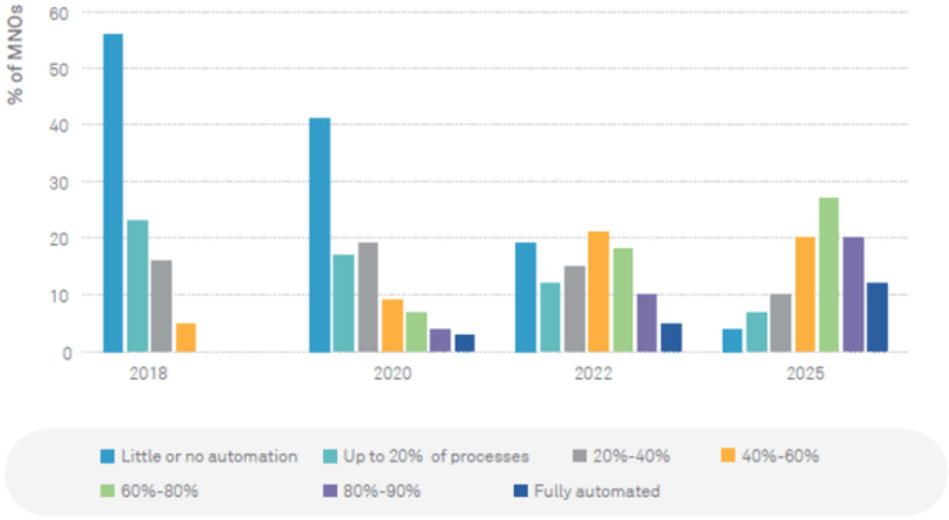


Fig. 15 Forecasted Percentage of MNOs with Network Automation (Source: Analysis Mason)

There are promising use cases envisioned by the industry on applying artificial intelligence to network automation [GSMA 2019]. It's a general understanding of the network automation should be data driven. However, the collection of user privacy-sensitive data and transmission of huge amount of data makes it very challenging to achieve meaningful AI innovations. In this section, we will summarize several potential use cases that Federated Learning can contribute to the goal of data-powered network automation.

Network Planning

The effective network planning starts from an accurate forecast of the busy area, busy hours and the increasing volume of the traffic. This can help network operators to discover the potential insufficient coverage or backhaul capacity, and to reach the best ROI within budget. Without using machine learning, accurate forecast is very hard to achieve since the traffic pattern will increasingly exhibit uneven distribution both in temporal and spatial domain.

One good example is the hotspot prediction for 5G networks. Existing LTE macro cells tend to reach the traffic capacity limit especially in dense urban areas. Some areas of the network see much higher traffic volume, thus higher network utilization, than other areas of the same network. One solution to this problem is to deploy small cells in these hotspots to offload the high volume of traffic from the macro cells to cope with the traffic bustiness. However, the costs for site acquisition, installation, configuration, backhaul, power, and maintenance are not trivial for deploying small cells. Thus, it is very important to have an accurate prediction of network hotspots to justify the costs incurred by small cell deployments.

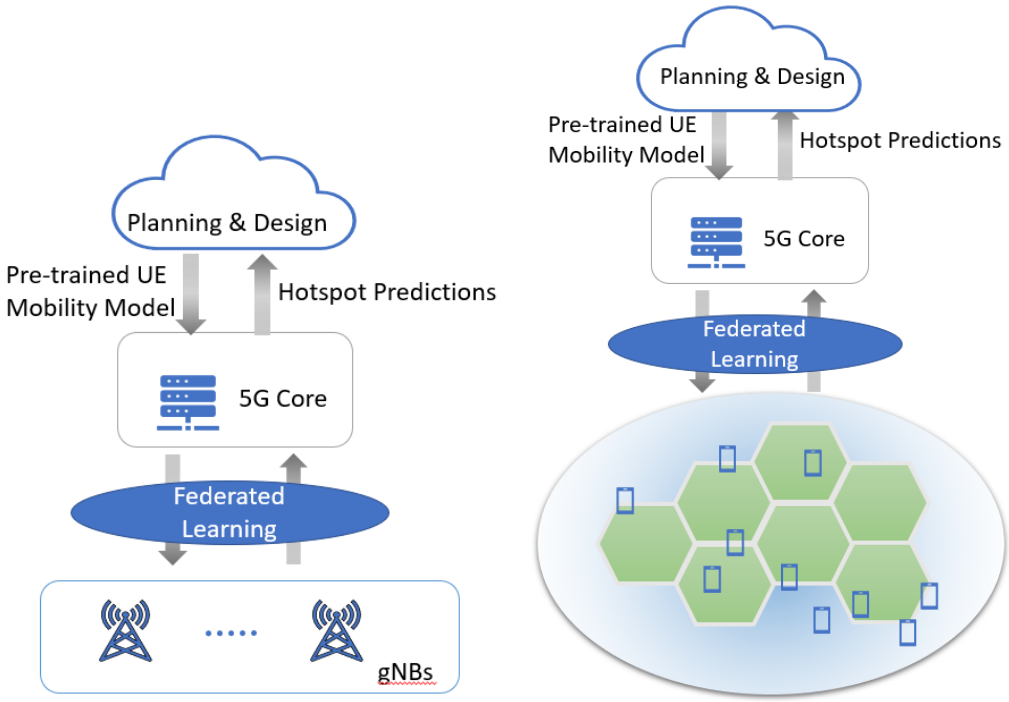
3GPP LTE supports methods of mobile terminal localization, e.g. Minimization of Drive Tests (MDT) and uplink or downlink Time Difference Of Arrival (TDOA). Using these methods to collect user position should get permission from the mobile subscriber because of the sensitive nature of the location data. [Zhang 2018] and [Liu 2020] got around this privacy data limitation by using aggregate data from the base stations instead of the individual mobile user location. [Zhang 2018] used Holt-Winters algorithm while [Liu 2020] used long short term memory (LSTM) RNN for the time series forecast to obtain the number of users in each cell and complete user density assessment.

In an effort to protect user privacy, [Ewe 2016] proposed a mechanism utilizing Pseudo Pico Cells (PPC), which cannot carry actual data traffic but can trigger the mobile UE to send out handover requests. The location of these pico-cells is known at any given time, thus combining with the handover requests, the geo-density of the mobile UEs can be indirectly inferred. The handover request message contains no privacy related information that would allow for identifying the mobile user. The disadvantage of this approach is it requires specialized equipment deployed in the network and the radio interference caused by a PPC can result in higher packet loss rate for the actual downlink traffic.

The historical locations of mobile users are privacy-sensitive information. Federated Learning can be applied in two ways for hotspot prediction: 1) between 5G core and gNBs and 2) between 5G core and mobile UEs, shown in Fig. 16. In first approach (Fig. 16a), the aggregate information on gNBs are used to train a forecast model on the number of UEs in each cell. Many gNBs can participate in the training together with the 5G core (especially AMF function) to improve the accuracy of the model. The advantage of using FL in this scenario is to avoid the transmission of the large amount of data from gNBs to 5G core via the backhaul.

In the second approach (Fig. 16b), the real-time geolocation of the mobile UEs can be used in the model training on the UEs locally, while the location data is kept private on the UEs. The advantage of FL is fully utilized in this scenario, and the forecast could be much more accurate than using the aggregate data on gNBs. However, the disadvantage is there is a burden to get mobile users to participate in the learning process. A good incentive mechanism is needed to get enough users on board.

In both scenarios of Fig. 16, the 5G core instead of the planning app on the O&M platform works as the model aggregator for the FL, this helps to reduce the latency of each training round of FL learning between the aggregator and the individual participant.



(a) FL between 5G core and base stations (b) FL between 5G core and UEs

Fig. 16 Hotspot Prediction with Federated Learning

The hotspot predication result returned to the Planning and Design App contains the forecasted number of UEs in each cell in different timeframes. This information can be utilized in various ways:

- Combining with other features (Building/structure ownership, lease cost, power availability, LoS blocking, etc.), use another machine learning algorithm to select optimal geolocation of the new small cell gNB RRUs.
- Design the backhaul network (topology changes, new link sizing, routing) to connect the new cells to 5G core.
- Scale up the microservices in 5G core VNFs to support the extra traffic load.

Network Performance Optimization

The goal of network performance optimization is to perform dynamic adjustments based on the traffic forecast and channel environment. The network functions need to be flexible enough to cope with the evolving situation and to meet the experience expectation from the end users.

In the hotspot prediction example given in the Network Planning section, good offline planning alone cannot solve the problem completely, because the hotspots may also shift between different areas during different time frames of the day. Obviously, the fixed small cell deployment solution is not cost effective, and most of the time a high percentage of the network resources is wasted, while some cells keep congested. An online optimization system can redirect the UEs to

the neighboring gNBs of the hotspot cell to offload the traffic, as shown in Fig. 17. This algorithm requires much higher speed than the planning one, such that the network congestion can be alleviated within minutes or even seconds. In Fig. 17, UEs in orange are redirected to the neighboring gNBs from a hotspot cell. The algorithms proposed in "Mobile User-Cell Association" section could be considered for this problem as well.

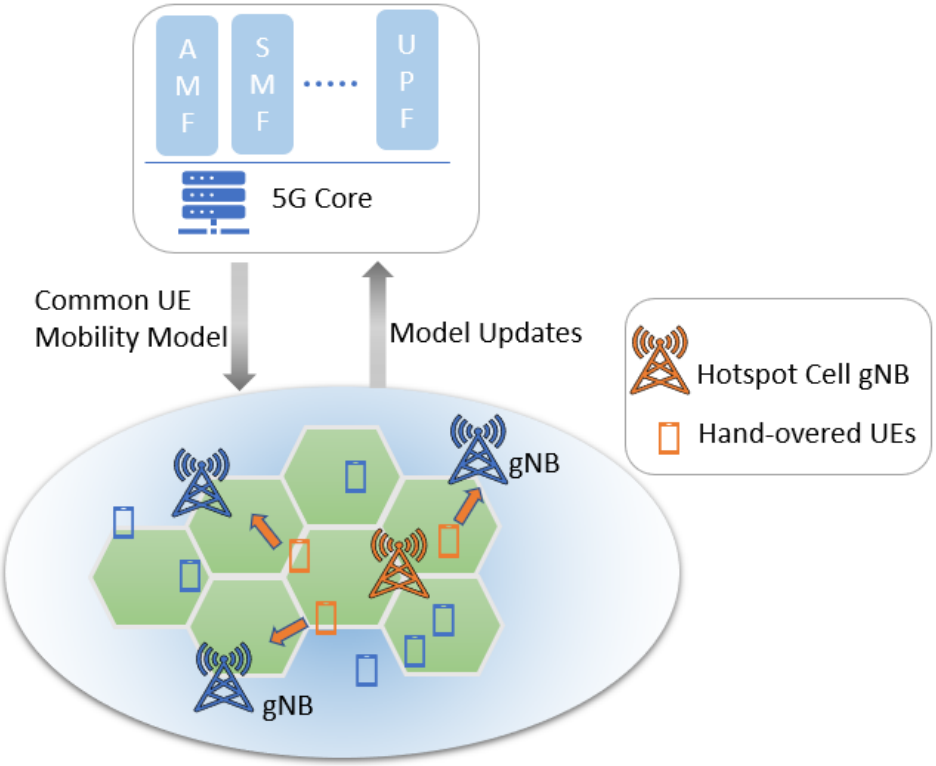


Fig. 17 Hotspot Cell Optimization with Federated Learning

After hotspot cells identified with FL, another possible way of increasing the network throughput without small cell deployment is forming virtual small cells (VSCs). In VSC approach, selected UEs are designated as cell heads to collect the intra-cell traffic and relay the aggregated traffic to the macro-cell base station [Behnad 2017]. Additionally, dynamic beamforming technique can be employed to enable a dedicated fronthaul link for VSC, thus minimizing the transmit power under throughput requirements and power constraints [Liu 2020].

Network Maintenance

AI algorithms can be used to distract the most important ones from the enormous warning messages and assist the operation personnel in finding out the root cause of the network issues, e.g. using the warning message pattern that happened before and is the most similar. They can

also be used to support dynamic thresholds, which will be more reasonable than preconfigured ones.

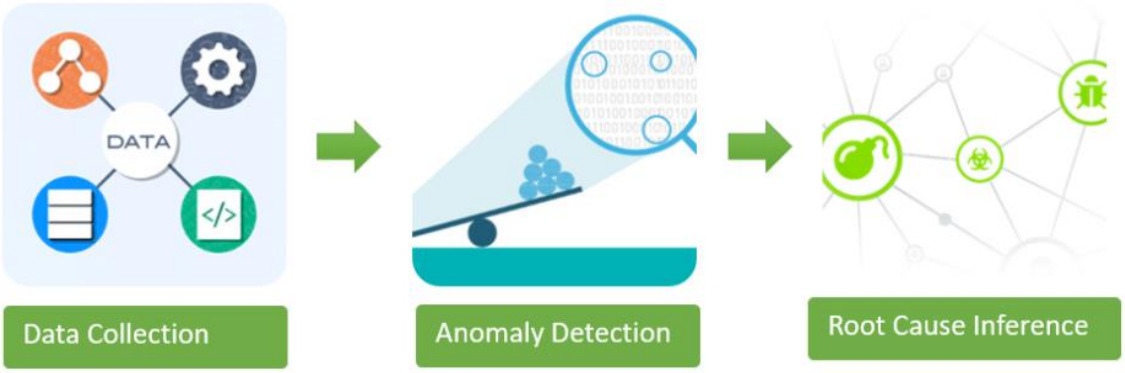


Fig. 18 General Steps of Root Cause Analysis

One of the most needed functionalities in network maintenance is Root Cause Analysis (RCA). The general steps of RCA are shown in Fig. 18, where both the anomaly detection and root cause inference phases both need to use machine learning algorithms to effectively locate the root cause. ML algorithms can be used to distract the real network anomalies from the huge amount of warning messages, and then assist the human in finding out the root cause of the network issues.

The federated learning can be used across different operators to share the knowledge of RCA, as shown in Fig. 19a. It's more straightforward for anomaly detection, for which a large set of deep learning models have been developed. For root cause inference, it's more involved since normally deep learning models cannot explain why a specific anomaly is the root cause.

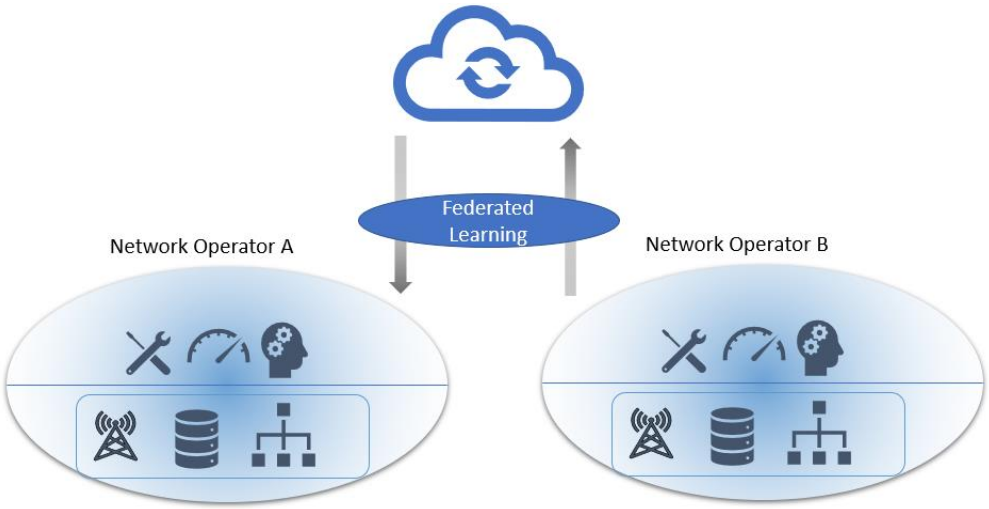


Fig. 19a Federated Learning across the boundary between network operators

In a longer term, it's possible that the network operators and vendors form an alliance to share the operational intelligence between them via Federated Learning, as shown in Fig. 19b. The members in the alliance join together for the mutual benefit of obtaining a set of more intelligent

models, which may not be limited to anomaly detection purposes. In this system, there is no central coordinator as in the case of classic Federated Learning. Each edge in Fig. 19b represents the neighbors adjacent to it are participating FL, with green edge denote an operator-vendor relationship and orange edge denote an operator-operator relationship.

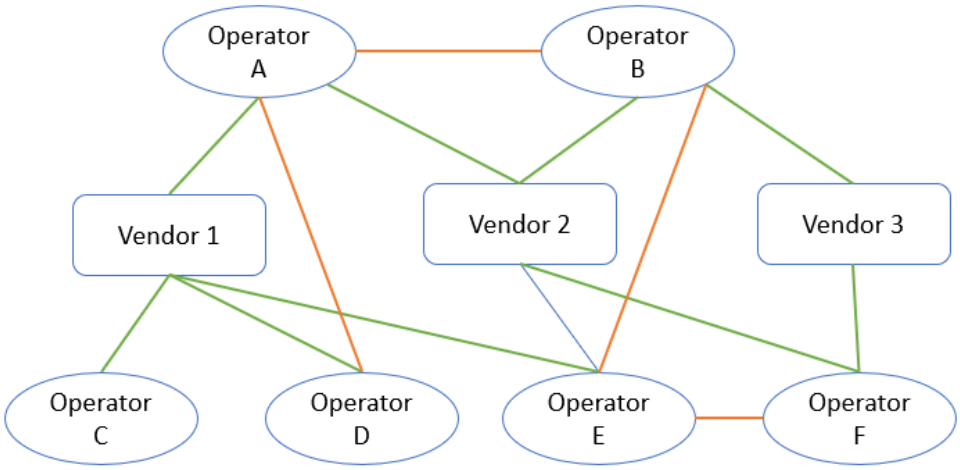


Fig. 19b Federated Learning between network operators and vendors

This peer-to-peer mode of model sharing poses new challenges to the standard Federated Learning framework, which will be discussed in the next section.

Service Provisioning

Streamlined network slice setup is imperative for operators to compete in 5G era. The customers would expect the network slice agreement signed almost real-time when the service is needed. Therefore, the operators need algorithms to predict whether the required QoS can be achieved under the current network environment and the estimated slice traffic.

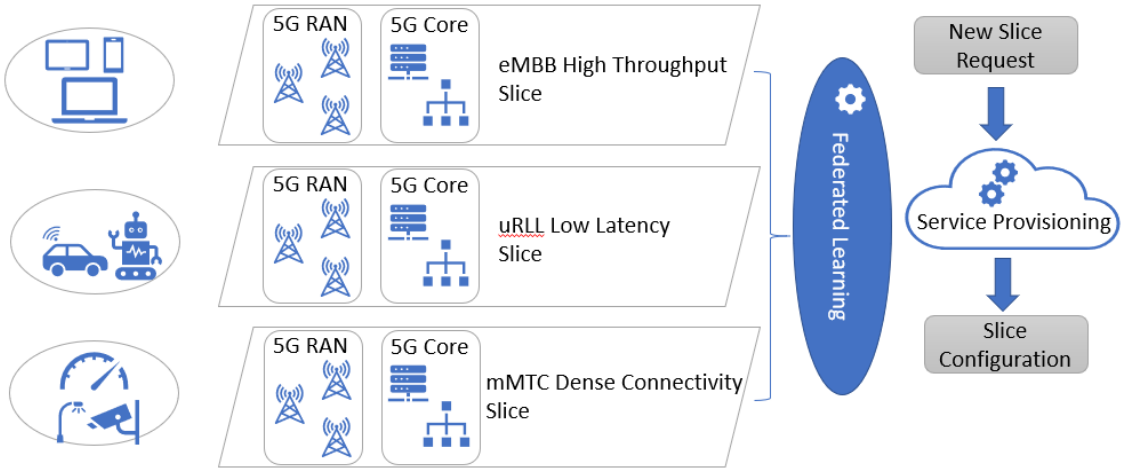


Fig. 20 Federated Learning for slice provisioning

As shown in Fig. 20, the federated learning can be used to train a deep learning model that maps the slice configurations to user quality of experience (QoE). When new slice requests arrive, the slice provisioning function can use the model to generate appropriate slice configurations.

Power Savings

Energy consumption constitutes between 20 – 40% of the OPEX of a network operator according to GSMA. A typical 5G base station consumes by average 68% more power than a 4G base station [Huawei 2019], as shown in Fig. 21. This mainly due to the need for more frequencies/antennas at macro cells and denser deployment of small cells. Edge cloud facilities with local processing and new IoT services increase the overall network power usage.

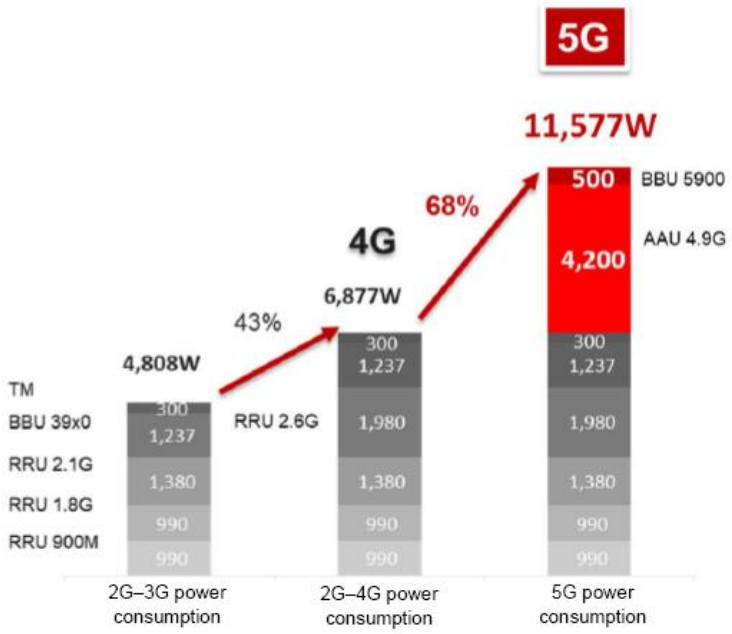


Fig. 21 Typical 5G cell site maximum power consumption [Huawei 2019]

A myriad of approaches are being investigated by the industry to reduce the power consumption of 5G network, including the intelligent collaboration at module level, site level, network level, and service level. One effective option of achieving efficient energy savings is to perform intelligent shutdown of frequency bands or entire cell site. This can be deemed as the reverse operation of hotspot prediction in the "Network Performance Optimization" section above. The goal now is to identify the idle sites instead of the hotspot sites. The way Federated Learning is used here is thus similar, it can be performed between the gNBs and the 5G core to perform the traffic load prediction of each individual site. If the intelligent shutdown can be performed in a smaller granularity (e.g. 30 minutes), a better balance can be reached between the power consumption and the offered quality of service.

Challenges of Federated Learning

Infrastructure

Several Federated Learning frameworks have been available, including TensorFlow Federated [TFF 2019], FATE [WeBank 2019], and PySyft [OpenMined 2020]. These frameworks abstract away machine learning and secure multiparty communication in programmatic APIs and languages. This abstraction allows data scientists and software developers to focus more on building the machine learning model and application without being burdened with low level details. However, these frameworks are still at their early stage, and there is no reported deployment at a scale of service provider network yet. For example, the Tensorflow Federated is still more of a simulation environment, and there is no device run time published that can be deployed to mobile devices for distributed learning.

Algorithm

Although the idea of privacy preservation of Federated Learning is very general, but strictly speaking FL currently can only be used for deep learning models. Although the deep learning models are widely used nowadays, there are many scenarios in network operations domain where deep learning is not suitable. For example, to perform root cause analysis, it is required for the ML model to explain why a particular issue is the root cause. Thus, deep learning models are not commonly seen in the close-loop remedy use cases. Also, deep learning models generally takes longer time to train even with the assistance of GPUs, therefore have limited applicability in scenarios require fast, incremental model updates at the network edge.

For scenarios that deep learning is indeed a good fit, it's imperative to carefully design the model aggregation algorithm in FL. The classic FederatedAveraging algorithm [McMahan 2017] is based on Stochastic Gradient Descent (SGD) algorithm, which does not guarantee convergence to global minimum for non-convex loss functions. Whether the FederatedAveraging algorithm gives good results highly depends on the structure of the deep learning model in question. Therefore, the challenge is a deep learning model that worked well with centralized training may not work as expected when transformed to federated training. The naive parameter averaging strategy used by FederatedAveraging algorithm that works surprisingly well in image classification may not cut it in another application.

Absence of Central Server

In classic federated learning, a central server orchestrates the training process and receives the model updates from all participants. When a trusted party, like a large company distributing a popular App, coordinates with its own users, generally the trusted party can play the role of central server. However, a reliable central server may not always be desirable in more collaborative learning scenarios. Furthermore, the server may even become a bottleneck or single point of failure when the number of clients increases. The scenario of peer-to-peer or fully decentralized learning poses great challenge to Federated Learning. There is no longer a global common model as in classic federated learning, but all local models need to converge to the desired global consensus by averaging one participant's local model parameters with its direct peers.

One possible solution is to utilize blockchain as a distributed ledger shared among training participants. However, data on the blockchains is publicly available by default, this would discourage users from participating in the federated learning. Therefore, the existing privacy-preserving techniques need to be modified to support decentralized federated learning. Many algorithmic questions still remain open in terms of the usability of decentralized algorithms for real-world deep learning.

Incentive

The success of Federated Learning depends on a sufficient number of participants to jointly utilize their data and computing power for model training and sharing. Due to the increased awareness in our society for data privacy, many users are not willing to take the risk to expose their private data without any rewards. For a FL-based system to work in production scale, an effective incentive mechanism needs to be in place. There could be large difference between individual participants in terms of the data size, data quality, computing power, channel quality, etc. Although Federated Learning does not necessarily assume an independent identical distribution (IID) data among participants, the error bound of local SGD in the non-IID scenario will deteriorate [Yu 2019]. It's quite challenging to design an incentive system that is fair to all participants and permits good training robustness and performance. There have been research efforts in this area [Nishio 2019] [Jiao 2020], but real experience from production system is not widely available yet.

Security

There have been data security enhancements to the original Federated Learning algorithm, such as differential privacy [Geyer 2017] and secure multiparty computation (SMC) [Truex 2019]. However, how to effectively protect the system against adversarial clients or compromised server is still an open issue. Malicious clients can inspect all messages received from the server in the rounds they participate in, which include the model iterates and possibly the final model. A malicious adversary controlling the server could simulate many fake client devices or could select previously compromised devices to participate the training. These potential security issues motivate the development of more refined trust models for federated learning frameworks.

Conclusion

Federated Learning is a promising ML paradigm aimed to solve the training data availability issue that is hindering the innovation of AI applications. In this article, we reviewed the basic principles of FL, surveyed the current research efforts in academia in 5G as well as edge computing areas, then proposed several potential use cases in network operations that can benefit from the application of the Federated Learning. Federated Learning is still at its early stage of real-world application, there are still plenty of challenges faced by industry to successfully apply Federated Learning to network operations at the scale of telecommunication service providers.

References

- [McMahan 2017] **Communication-Efficient Learning of Deep Networks from Decentralized Data,** Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, 2017
- [Shokri 2015] **Privacy-Preserving Deep Learning,** Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, 2015
- [Bonawitz 2019] **Towards Federated Learning at Scale: System Design,** 2nd SysML Conference, 2019
- [Geyer 2017] **Differentially Private Federated Learning: A Client Level Perspective**, 31st Conference on Neural Information Processing Systems (NIPS 2017)
- [Truex 2019] **A Hybrid Approach to Privacy-Preserving Federated Learning,** 12th ACM Workshop on Artificial Intelligence and Security, November 2019
- [Felix 2019] **Robust and Communication-Efficient Federated Learning From Non-i.i.d. Data,** IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, November 2019
- [Bastug 2014] **Living on the edge: The role of proactive caching in 5G wireless networks**, IEEE Communications Magazine (Volume: 52, Issue: 8, Aug. 2014)
- [Qiao 2016] **Proactive Caching for Mobile Video Streaming in Millimeter Wave 5G Networks**, IEEE Transactions on Wireless Communications (Volume: 15, Issue: 10, Oct. 2016)
- [Yu 2018] **Federated Learning Based Proactive Content Caching in Edge Computing**, 2018 IEEE Global Communications Conference (GLOBECOM)
- [Wang 2019] **In-Edge AI: Intelligentizing Mobile Edge Computing, Caching and Communication by Federated Learning**, IEEE Network (Volume: 33, Issue: 5, Sept.-Oct. 2019)
- [Qi 2020] **Popularity Prediction with Federated Learning for Proactive Caching at Wireless Edge**, 2020 IEEE Wireless Communications and Networking Conference (WCNC)
- [Harper 2015] **The Movielens Datasets: History and Context**, ACM Transactions on Interactive Intelligent Systems, vol. 5, no. 4, Dec. 2015.
- [Ordonez-Lucena 2017] **Network Slicing for 5G with SDN/NFV: Concepts, Architectures, and Challenges**, IEEE Communications Magazine (Volume: 55, Issue: 5, May 2017)
- [Sciancalepore 2017] **Mobile Traffic Forecasting for Maximizing 5G Network Slicing Resource Utilization**, 2017 IEEE INFOCOM Conference
- [Huynh 2019] **Real-Time Network Slicing with Uncertain Demand: A Deep Learning Approach**, IEEE Journal on Selected Areas in Communications (Volume: 37, Issue: 6, June 2019)

- [Lee 2012], **SLAW: Self-Similar Least-Action Human Walk**, IEEE/ACM Transactions on Networking, (Volume: 20, Issue 2, Apr. 2012)
- [Fantacci 2020] **When Network Slicing Meets Prospect Theory: A Service Provider Revenue Maximization Framework**, IEEE Transactions on Vehicular Technology (Volume: 69, Issue: 3, March 2020)
- [Kolias 2017] **DDoS in the IoT: Mirai and other botnets**, IEEE Computer Magazine, (Volume: 50, Issue: 7, March 2017)
- [Hafeez 2016] **SecureBox: Toward safer and smarter IoT networks**, 2016 ACM Workshop Cloud Assisted Networking
- [Miettinen 2017], **IoT sentinel: Automated device-type identification for security enforcement in IoT**, 2017 IEEE 37th Int. Conf. Distributed Comput. Systems (ICDCS)
- [Nguyen 2019] **DÏoT: A Federated Self-learning Anomaly Detection System for IoT,** 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)
- [Semiari 2019] **Integrated Millimeter Wave and Sub-6 GHz Wireless Networks: A Roadmap for Joint Mobile Broadband and Ultra-Reliable Low-Latency Communications**, IEEE Wireless Communications (Volume: 26, Issue: 2, April 2019)
- [Elshaer 2016] **Downlink and Uplink Cell Association with Traditional Macrocells and Millimeter Wave Small Cells**, IEEE Transactions on Wireless Communications (Volume: 15, Issue: 9, September 2016)
- [Chen 2017] **Echo State Networks for Self-Organizing Resource Allocation in LTE-U with Uplink-Downlink Decoupling**, IEEE Transactions on Wireless Communications (Volume: 16, Issue: 1, Jan. 2017)
- [Waheidi 2019] **User Driven Multiclass Cell Association in 5G HetNets for Mobile & IoT Devices**, IEEE Access, June 2019
- [Chen 2020] **Federated Echo State Learning for Minimizing Breaks in Presence in Wireless Virtual Reality Networks**, IEEE Transactions on Wireless Communications (Volume: 19, Issue: 1, Jan. 2020)
- [Sada 2019] **A Distributed Video Analytics Architecture Based on Edge-Computing and Federated Learning,** IEEE Intl Conf on Dependable, Autonomic and Secure Computing 2019
- [Chen 2016] **Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing**, IEEE/ACM Transactions on Networking (Volume: 24, Issue: 5 , October 2016)
- [Chen 2018] **Task offloading for mobile edge computing in software defined ultra-dense network**, IEEE Journal on Selected Areas in Communications (Volume: 36, Issue: 3, March 2018)
- [Dinh 2018] **Learning for Computation Offloading in Mobile Edge Computing**, IEEE Transactions on Communications (Volume: 66, Issue: 12, Dec. 2018)

- [Ren 2019] **Federated Learning-Based Computation Offloading Optimization in Edge Computing-Supported Internet of Things**, IEEE Access (Volume: 7, June 2019)
- [Qian 2019] **Distributed Active Learning Strategies on Edge Computing**, 2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)
- [Sanabria-Russo 2019] **IoT Data Analytics as a Network Edge Service**, IEEE INFOCOM 2019
- [GSMA 2019] https://www.gsma.com/futurenetworks/
- [Ewe 2016] **Mobile User Hotspot Detection in LTE Networks by Moving Pseudo Pico Cells,** in European Wireless 2016 (22nd European Wireless Conference)
- [Zhang 2018] **Hotspot Localization and Prediction in Wireless Cellular Networks via Spatial Traffic Fitting**, 2018 IEEE/IFIP Network Operations and Management Symposium
- [Behnad 2017] **Virtual Small Cells Formation in 5G Networks**, IEEE COMMUNICATIONS LETTERS (Volume: 21, Issue: 3, Mar. 2017)
- [Liu 2020] **Deep Learning Based Hotspot Prediction and Beam Management for Adaptive Virtual Small Cell in 5G Networks**, IEEE Transactions on Emerging Topics in Computational Intelligence (Volume: 4, Issue: 1, Feb. 2020)
- [Huawei 2019] **5G Power Whitepaper**, https://carrier.huawei.com/en/spotlight/5g
- [TFF 2019] **TensorFlow Federated**, https://www.tensorflow.org/federated
- [WeBank 2019] **FATE**, https://fate.fedai.org/
- [OpenMined 2020] **PySyft**, https://github.com/OpenMined/PySyft
- [Yu 2019] **On the Linear Speedup Analysis of Communication Efficient Momentum SGD for Distributed Non-Convex Optimization**, 2019 International Conference on Machine Learning
- [Nishio 2019] **Client Selection for Federated Learning with Heterogeneous Resources in Mobile Edge**, IEEE International Conference on Communications, May 2019
- [Jiao 2020] **Toward an Automated Auction Framework for Wireless Federated Learning Services Market**, IEEE Transactions on Mobile Computing, May 2020