中国移动
China Mobile

FCFN
FEDERATED COMPUTING FORCE NETWORK

Federated Computing
Force Network (FCFN)

Enhancing Federated Learning Architecture through Parameter Optimization and Neural Architecture Search in Computing Force Network

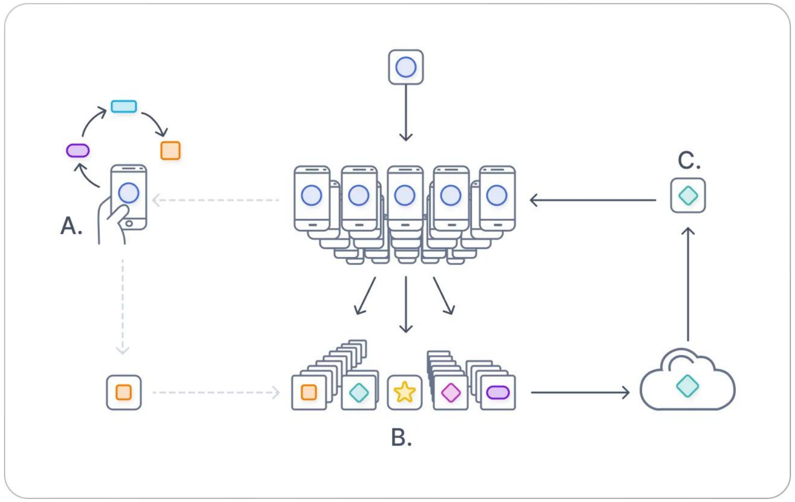
APONS

China Mobile Research Institute, Beijing, China

China Mobile Technology (USA) Inc, Milpitas, CA, U.S.A.

0. Background

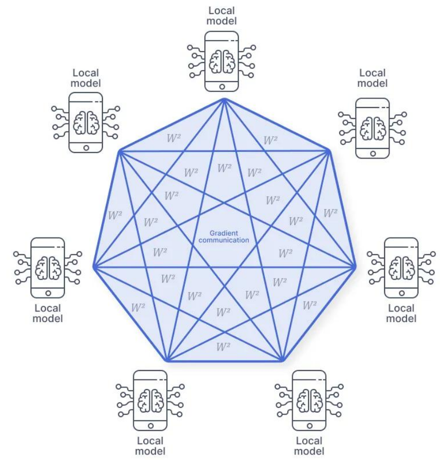
Computing Force Network



Computing Force Network: with the flourish of new business scenarios such as hybrid cloud, multi-cloud, AI, big data, edge computing, building a new information infrastructure based on multiple key technologies and converging cloud and network, will better support global digital transformation. This new infrastructure will not only relates to cloud, it is getting more and more connected with network, and at the same time, we also need to consider how to converge multiple technologies like AI, Blockchain, big data, security to provide this integrated service. China Mobile is working on building Computing Force Network infrastructure to meet the business requirements from digital and economy development, Computing Force Network is a new information infrastructure that takes network as the foundation and computing as the core, deeply converge Artificial intelligence, Block chain, Cloud, Data, Network, Edge computing, Terminal computing, Security, to provide all-in-one services. This CFN infrastructure will manage ubiquitous distributed computing force by a unified platform, integrate multiple technologies like AI, blockchain and etc, to provide intelligent services to consumers, to support digital transformation and new business like IoT, IoV, Metaverse and etc.

0. Background

Federated learning



Federated learning is a decentralized approach to training machine learning models. It doesn’t require an exchange of data from client devices to global servers. Instead, the raw data on edge devices is used to train the model locally, increasing data privacy. The final model is formed in a shared manner by aggregating the local updates.

0. Background

Federated Computing Force Network (FCFN)

Combining "Federated Learning" with "Computing Force Network" can lead to the concept of a "Federated Computing Force Network" (FCFN). In this context, FCFN would represent a distributed infrastructure that harnesses federated learning techniques to train machine learning models across a network of interconnected computing resources.

In practical terms, within the FCFN framework, each node in the network (which could be edge devices, cloud servers, or other computing entities) would participate in training machine learning models using local data. Federated learning algorithms would orchestrate this process, allowing each node to learn from its own data without sharing it centrally. These local models would then be aggregated to create a global model that benefits from the collective knowledge of all participating nodes.

FCFN could provide several advantages, including enhanced data privacy, reduced communication overhead, and the ability to leverage distributed computing resources for machine learning tasks. This approach aligns with the vision of the Computing Force Network by integrating federated learning as a core component, enabling intelligent services while preserving data privacy and leveraging the collective computing power of the network.

1. Introduction

APONS:
Enhancing Federated Learning Architecture through
Parameter Optimization and
Neural Architecture Search in Computing Force Network

The Federated Edge Learning technique can successfully assist the edge deployment of 6G networks. Using a lot of user data, it can train machine learning models by communicating with many edge clients. However, in 6G-enabled mobile edge computing networks, heterogeneity and resource limitations among distributed edge clients can lower the effectiveness of Federated Edge Learning training. This study suggests a novel Federated Learning framework to expedite the training process. This paper proposes a new model based on the relationship between training loss, resource consumption, and heterogeneity. Then, to reduce latency effects brought on by client heterogeneity and resource limitations, we suggest using a search technique called **APONS** to generate local models of edge clients and optimal imprecision of band allocation. As a result, modifying the percentage of frequency bands and the local model of the edge client's inaccuracy can significantly increase training efficiency. The simulation outcomes demonstrate our algorithm's benefits in increasing training effectiveness.

2.Relate Work

Research on advanced networks highlights AI integration in Mobile Ultra Broadband and Super IoT, focusing on applications across industrial, medical, and vehicular sectors. Google introduced federated learning for enhanced privacy in distributed settings, with subsequent studies optimizing data transmission and minimizing latency to manage learning loss within resource constraints. Enhancements in federated systems have prioritized efficient data and resource allocation, ensuring robust model training in distributed IoT networks. Further advances include the use of gradient-based methods in network architecture searches within federated learning frameworks, incorporating strategies like diverse searches and differential privacy to boost model accuracy and efficiency. These developments are pivotal in creating resource-efficient, privacy-focused distributed network systems.

3. Proposed Architecture

Figure 1 shows a mobile edge computing network built by a base station and multiple mobile clients. Edge servers also provide computing and storage capabilities and antennas, whereas each mobile client has an antenna and a local dataset. Furthermore, the Base Station and all mobile clients train a standard model using the APONS architecture's federated training algorithm.

Figure 1: APONS training model on mobile edge computing enabled by 6G

We consider an federated learning is a setting with a set of clients n of size N, each holding a dataset client 1, client 2,..., client n. Each client k has training and validation data used to solve a supervised learning task.

3. Proposed Architecture

APONS exploits parameter sharing to perform informed updating of non-architecture parameters during the learning process. It consists of two stages: the search stage and verifies stage (See Figure 2).

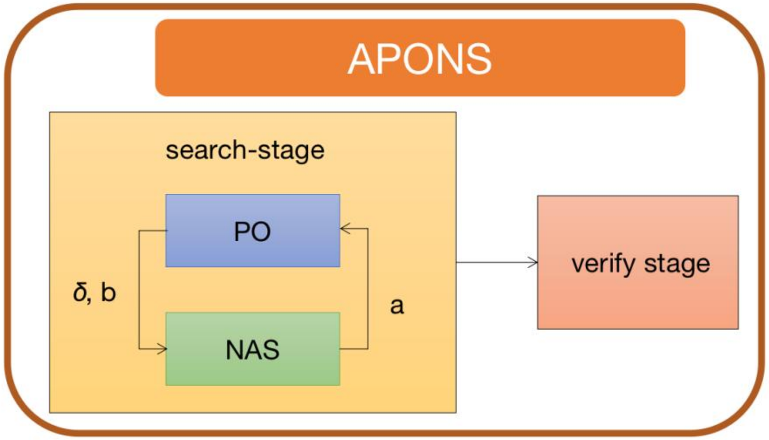


Figure 2. Two stages of APONS

(1) Search Stage
The aim is to identify high-performance cell-based architectures for building models used in the validation stage. The search stage includes a PO (Parameter Optimization) and a NAS (Neural Architecture Search) stage.

(2) Verify Stage
The unit architecture found during the search phase can use to build an evaluation model. When training this model, we use PO to identify well-optimized non-architectural parameters to optimize global validation loss. Our search space contains only non-architectural parameters. We adjusted the learning rate, weight decomposition, speed, and path loss. We chose the first three parameters because they directly affect the optimizer's behavior and, thus, the learning progress. Path omission was determined to avoid over-fitting, and it was demonstrated that APONS could select an appropriate configuration.