

# Flower 연합학습 프레임워크를 활용한 통신 상황에 따른 학습 성능 평가 연구

최동진, 김준우, 이웅희

한성대학교

kkyy1099@naver.com, junwooha1@gmail.com, whlee@hansung.ac.kr

## A Study on Evaluating Learning Performance depending on Communication Situation using Flower Federated Learning Framework

Dongjin Choi, Junwoo Kim, Woonghee Lee

Hansung University

### 요약

많은 주목을 받으며 빠르게 발전하고 있는 다양한 인공지능 학습 기술들은 수많은 데이터를 사용하여 학습을 수행하는데, 이를 위한 데이터들이 한곳에 모여 있는 경우보다 여러 장치들에 분산되어 저장되어있는 경우가 더욱 많다. 학습을 수행하기 위해 많은 데이터들을 한곳으로 모아야만 하는 한계를 극복하기 위해 연합학습이 제안되었으며, 연합학습이 작동하기 위해서는 서버와 참여 장치들간에 통신이 지속적으로 수행되어야 한다. 하지만 학습에 참여하는 모든 장치들의 통신 상황은 서로 다르며 지속적으로 변화하므로 이러한 통신 상황이 연합학습의 성능에 영향을 미치게 된다. 따라서 본 논문에서는 Flower 연합학습 프레임워크를 활용하여 테스트환경을 구축하였으며, 통신상황에 따른 시간 지연이 연합학습 성능에 영향을 미치는 것을 보였다.

### I. 서론

기술의 발전에 의해 장치들의 연산 능력이 향상되고 많은 데이터를 다룰 수 있게 됨으로써 인공지능(Artificial intelligence)은 빠른 속도로 발달하고 있다. 그와 동시에 스마트폰의 대중화로 더욱 많은 데이터들이 생성되게 되었으며, 이러한 데이터를 활용하여 딥러닝(Deep learning)을 포함한 다양한 학습 기술들이 많은 주목을 받고 있다. 딥러닝 기술은 데이터를 인공신경망(Artificial neural network)에 넣어 학습을 수행해야 하므로 딥러닝을 수행하기 위해선 학습에 필요한 많은 데이터가 연산이 가능한 서버로 옮겨져야 한다[1]. 따라서 이러한 한계를 극복하고 많은 사물인터넷(Internet of Things) 디바이스들의 데이터를 효과적으로 학습에 활용하기 위한 연합학습(Federated Learning) 방식이 제안되었다[4]. 연합학습이 수행되기 위해서는 학습에 참여하는 장치 간 통신이 지속적으로 유지되어야 하므로 통신상황이 연합학습 시스템의 학습 성능에 주요한 영향을 미치게 된다. 따라서 본 연구에서는 연합학습을 위한 프레임워크(Framework)인 Flower를 사용하여 테스트환경을 구축하여 통신 상태가 연합학습의 학습 성능에 미치는 영향을 분석하였다.

### II. 연합학습

기존의 딥러닝 기술들은 학습을 위해 많은 양의 데이터를 필요로 한다. 하지만 이러한 데이터들은 일반적으로 하나의 저장소에 저장된 것이 아닌 다양한 기기에 분산 저장되어 있다. 많은 기기에 분산 저장된 데이터들은 학습을 위해 서버와 공유를 한다. 데이터의 크기가 작고 기기들의 개수가 적다면 서버로 데이터를 모으는 기존의 학습 방식을 사용하여도 되나, 데이터의 크기가 크고 기기들의 개수가 무수히 많아진다면 기존의 방식은 결코 효율적이지 않다. 연합학습은 이러한 기존 방식의 한계를 보완하기 위해 제안되었다.

그림 1은 연합학습의 개념도를 보여주고 있다. 연합학습의 작동 방식은 다음과 같다. 먼저, 중앙서버에서 제작한 학습 모델을 참여 장치들에게 배포한다. 장치들은 각 장치들의 로컬 데이터를 활용해 배포 받은 모델을 학습시킨 후 학습된 모델을 중앙서버로 다시 전송한다. 중앙서버는 학습된 모델들을 통합하고, 통합된 모델을 다시 장치들에게 배포한다[2]. 이러한 모델의 학습 과정은 만족할만한 성능의 네트워크가 구성될 때까지 계속해서 반복된다.

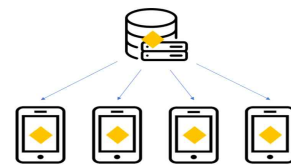


그림 1. 연합학습 개념도.

### III. 통신환경에 따른 연합학습의 학습 성능 분석

#### 1. 실험 환경

Flower는 연합학습을 위한 프레임워크로서, 10,000개 이상의 클라이언트를 무리 없이 돌릴 수 있는 확장성을 가지고 있다. 또한 Flower는 서버뿐만 아니라 모바일 장치와 같은 다양한 장치를 지원하며 기존의 인공지능 학습 프레임워크인 파이 토치(PyTorch)와 텐서플로(TensorFlow)를 연계해서 사용할 수 있다[3]. 따라서 본 연구진은 Flower 프레임워크를 활용하여 실험 환경을 구축하였으며 실험에 사용된 PC의 성능은 표 1과 같다. 연합학습 성능에 대한 통신 상황의 영향성을 보기 위함으로 고성능의 학습을 필요로 하지 않아, 본 실험에서는 GPU를 활용하지 않았다.

구성요소	사양
Processor	AMD Ryzen 5 3600 @3.6GH
Memory	16GB DDR4 2133MHz
OS	Ubuntu 20.04.3
Flower Version	V0.18.0
Pytorch Version	V1.10.0 + cu102
Anaconda Version	V4.10.1

표 1. 실험에 사용된 PC의 사양.

## 2. 실험 과정

본 실험에서는 만족하는 성능에 도달하기 위한 학습 시간을 측정하였다. 1개의 서버와 3개의 참여 장치가 수행하는 연합학습을 모사하기 위해 1개의 서버 터미널과 3개의 클라이언트 터미널로 구성된 총 4개의 터미널을 활용하여 연합학습을 수행하였다. 수행 학습으로는 flower git[5] 코드의 quickstart\_pytorch 코드를 활용하여 CIFAR10 데이터셋을 분류하는 CNN(Convolutional Neural Network) 네트워크를 학습시키는 연산을 수행하였다. 그림 2는 CNN을 구성하는 코드를 보여준다.

```
class Net(nn.Module):
    def __init__(self) -> None:
        super(Net, self).__init__()
        self.conv1 = nn.Conv2d(3, 6, 5)
        self.pool = nn.MaxPool2d(2, 2)
        self.conv2 = nn.Conv2d(6, 16, 5)
        self.fc1 = nn.Linear(16 * 5 * 5, 120)
        self.fc2 = nn.Linear(120, 84)
        self.fc3 = nn.Linear(84, 10)

    def forward(self, x: torch.Tensor) -> torch.Tensor:
        x = self.pool(F.relu(self.conv1(x)))
        x = self.pool(F.relu(self.conv2(x)))
        x = x.view(-1, 16 * 5 * 5)
        x = F.relu(self.fc1(x))
        x = F.relu(self.fc2(x))
        x = self.fc3(x)
        return x
```

그림 2. CNN 구성 코드.

client.py의 코드를 수정하여, 0에서 1까지의 확률에 따라 3에서 5초의 지연이 발생하는 코드를 추가함으로써 통신 상황에 따른 지연을 모사하였다. 그림 3은 위에서 설명한 코드의 수정 영역을 보여준다. 통신 환경이 시간에 따라 계속해서 변경된다는 것을 고려하여 지연확률을 나타내는 threshold는 0에서 1까지 0.2씩 변경하였고 그에 따른 통신 지연을 주기 위해 time.sleep() 값을 3에서 5까지 변경하며 실험을 수행하였다.

```
def fit(self, parameters, config):
    current_time = datetime.now().time()
    print(current_time, ' / Fit')

    self.set_parameters(parameters)
    train(net, trainloader, epochs=1)

    threshold = 0.6
    if random.random() < threshold:
        print('Delayed')
        time.sleep(2)
    return self.get_parameters(), len(trainloader), {}
```

그림 3. 통신 상황에 따른 지연 발생 코드.

## 3. 실험 결과

본 실험의 결과 그래프는 그림 4와 같다. 통신 지연 시간을 4초로 설정한 실험을 나타내는 delay 4가 굴곡이 심한 것을 알 수 있는데, 이는 실험에 사용한 PC의 연산 능력(computing power)으로 인해 학습 시간이 기본적으로 오래 걸림으로써 발생하는 학습 시간의 편차에 의한 결과이다. 하지만 모든 통신 지연에서 지연 확률이 높아질수록 전체 학습 시간이 증가함을 알 수 있다. 본 결과를 통해 통신 상황이 연합학습의 성능에 큰 영향을 미친다는 것을 알 수 있으며, 연합학습에서 효율적인 통신을 수행하는 것이 매우 중요하다는 것을 알 수 있다.

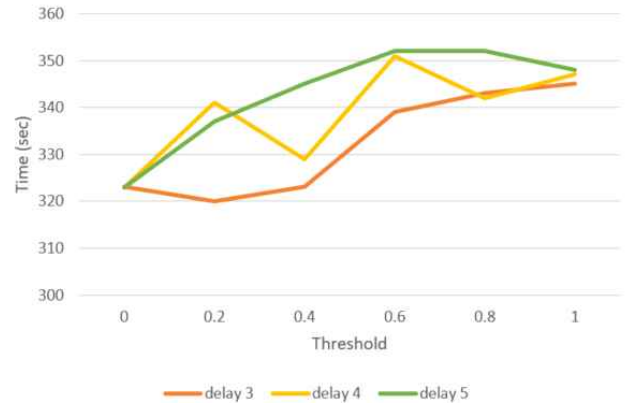


그림 4. 지연확률에 따른 학습 소요 시간.

## IV. 결론

본 논문에서는 Flower 프레임워크를 통해 통신환경이 연합학습에 어떠한 영향을 미치는가를 분석하였다. 4개의 터미널을 사용해 3개의 참여 장치와 1개를 서버를 모사한 테스트환경을 구축하였고 지연 확률에 따라 참여 장치들의 통신 지연이 발생하는 상황을 모사하여 실험을 수행하였다. 그리고 실험 결과를 통해 통신환경이 연합학습에 주요한 영향을 미친다는 것을 보였다.

## ACKNOWLEDGMENT

이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. 2021R1G1A1092939).

## 참 고 문 헌

- [1] 안성만. "딥러닝의 모형과 응용사례." 지능정보연구 22.2 (2016): 127-142.
- [2] Li, Qibin, Zeyi Wen, and Bingsheng He. "Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection." (2019).
- [3] Flower: A Friendly Federated Learning Framework (<https://flower.dev/>)
- [4] McMahan, Brendan, et al. "Communication-efficient learning of deep networks from decentralized data." Artificial intelligence and statistics. PMLR, 2017.
- [5] Flower - A Friendly Federated Learning Framework, (<https://github.com/adap/flower>)