

# 연합 학습의 성능 향상에 관한 연구

조영훈, 이재욱, 고한얼, 백상헌

고려대학교

{lepo033, iioioioio123, heko, shpack}@korea.ac.kr

## A Study on the Performance Improvement of Federated Learning systems

Yeonghun Cho, Jaeuk Lee, Haneul Ko, and Sangheon Pack

\*Korea University

### 요 약

최근 모바일 기기의 증가로 인해 연합 학습 (Federated Learning)에 대한 관심이 늘고 있다. 연합학습에 참여하는 단말 선택 문제는 연합학습의 성능 향상을 위한 주요한 이슈이다. 본 논문에서는 모바일 엣지 컴퓨팅 환경에서 연합학습을 위한 단말 선택 기법인 FedCS를 소개하고, 특정 상황에서 FedCS보다 더 나은 성능을 제공할 수 있는 Federated Learning with Adaptive Determination (FedAD)을 제안한다.

### I. 서 론

최근 다수의 컴퓨팅/통신능력을 가지는 단말들이 연합체를 형성하여 하나의 심층 네트워크 (Deep network)를 학습시키는 방식인 연합학습 (Federated learning)기법에 대한 관심이 증가하고 있다 [1]. 많은 양의 데이터를 클라이언트로 분산시켜 학습을 수행하기 때문에, 하나의 서버를 통해 학습하는 방식보다 적은 컴퓨팅 자원들로 빠른 학습속도를 갖는 장점이 존재한다. 이러한 장점은 제한적인 컴퓨팅 자원을 갖는 엣지 컴퓨팅 환경에서 더 극대화될 것으로 전망된다.

하지만, 연합체를 형성하는 단말들의 컴퓨팅/통신능력들이 상이하기 때문에 어떤 단말들을 통해 연합학습을 수행하느냐에 따라 성능 차이가 발생하는 문제점이 존재한다. 예를 들어, 연합학습을 위해 임의로 단말들을 선택하는 경우, 특정 단말이 다른 단말들에 비해 낮은 컴퓨팅 능력을 갖추고 있거나 무선 채널 상태가 좋지 않은 경우가 발생한다. 이때, 다른 단말들이 특정 단말을 기다려야 하므로 학습 성능이 저하되는 문제가 존재한다. 해당 문제를 해결하기 위해 [2]에서는 연합체를 구성하기 전에 단말들의 컴퓨팅/통신능력을 수집하고, 미리 정해진 학습 주기 내에 학습한 결과를 보내 줄 수 있는 단말들을 최대한 많이 선택하는 기법인 FedCS 기법을 제안하였다. 해당 기법의 경우 클라이언트들이 성능이 낮은 다른 특정 클라이언트를 기다리게 되는 경우가 적어져 학습 성능이 향상되지만, 학습 주기를 어떻게 정하느냐에 따라 여전히 성능이 영향을 받는 문제가 존재한다. 예를 들어, 180초라는 학습 주기를 정하여 그 안에 학습할 수 있는 클라이언트를 선택 할 때, 학습에 각각 5초, 30초, 180초를 소모하는 클라이언트가 선택 될 수 있는데, 이 경우 5초, 30초의 시간을 소모하는 클라이언트들은 더 많은 업데이트를 할 수 있음에도 한 주기가 끝날 때 까지 대기 상태로 있어야 한다. 이 기다림은 학습 성능의 저하를 야기한다. 따라서 본 논문에서는 FedCS와 달리 단말들의 컴퓨팅/통신능력을 고려하여 학습 주기를 적응적으로 결정하는 기법인 Federated Learning with Adaptive Determination (FedAD)을 제안하고, 실험을 통해 두 기법

의 성능을 비교 분석한다.

본 논문의 구성은 다음과 같다. 우선 본문에서 시스템 모델과 FedAD를 제안한다. 이후 FedCS와 FedAD의 성능 비교를 통해 연합학습에서 단말 선택의 중요성을 분석하고 결론을 맺는다.

### II. 본론

#### 2.1 시스템 모델

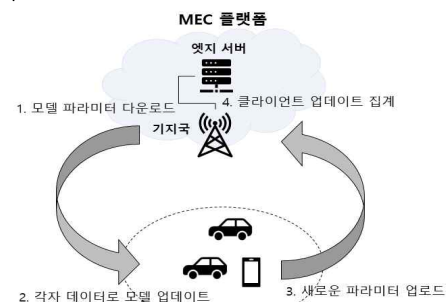


그림 1. 시스템 모델

그림 1은 본 논문의 시스템 모델을 나타낸다. 시스템 모델은 파라미터 서버 역할을 하는 엣지 서버와 연합체를 형성하여 학습을 수행하는 단말들이 존재한다. 연합학습을 수행하는 경우 우선 엣지 서버가 학습에 참여할 단말들을 선택하고 선택된 단말들에 학습할 심층 네트워크를 전송한다. 해당 심층 네트워크를 받은 단말들은 학습을 시킨 후에 학습된 결과물을 엣지 서버에 미리 정해진 시간 내에 다시 전송한다. 해당 결과물을 받은 엣지 서버는 학습된 결과를 종합하여 심층 네트워크를 업데이트하고, 업데이트된 심층 네트워크를 다시 학습시키기 위해 단말들을 선택한다. 상기의 과정을 반복적으로 수행한다.

상기의 동작 과정을 수행하기 위해 단말들은 각기 다른 컴퓨팅 능력을 갖추고 있으며, 하나의 기지국을 통해 무선 통신을 수행한다. 단

말의 컴퓨팅 능력과 통신 처리량은 정규분포를 따른다고 가정한다. 또한, 각 클라이언트에 대한 패킷 손실률은 무시할 수 있다고 가정한다. 각 클라이언트의 처리량은 시간에 따라 달라질 수 있다고 가정한다.

## 2.2 FedAD (Federated Learning with Adaptive Determination)

FedCS는 학습 주기를 지정하여 그 시간마다 업데이트를 하지만 본 논문에서 제안하는 FedAD는 FedCS와 같은 방식으로 클라이언트를 수집하고 지정된 클라이언트 수를 만족하면 바로 업데이트를 실행한다. 본 논문의 프로토콜을 구체적으로 기술하면 1. 서버는 랜덤 혹은 공개 데이터로 사전 학습을 통해 글로벌 모델을 초기화한다. 2. 리소스 요청을 통해 클라이언트들의 자원 정보를 파악하고 3. 최적의 클라이언트를 선택한다. 클라이언트를 선택하는 수식은 아래와 같다.

$$\begin{aligned} \min T_{rosnd} \\ s.t. |S| \geq S_{req} \\ T_{rosnd} = T_s^d + \theta_{|S|} + T_{agg} + T_{cs} \end{aligned}$$

매 round의 포함되는 모든 과정에 소모되는 시간인  $T_{rosnd}$ 를 최소화하는 것을 목표로 한다. 업데이트 과정이 필요로 하는 최소한의 클라이언트 수  $S_{req}$ 를 지정한다.  $T_{rosnd}$ 는 글로벌 모델을 배포하는 시간인  $T_s^d$ 와 클라이언트들에게서 넘겨받은 모델을 집계하는 시간인  $T_{agg}$ 와 클라이언트를 선택하는 과정에서 소모되는 시간  $T_{cs}$ , 그리고  $\theta_{|S|}$ 의 합으로 이루어진다.  $\theta_{|S|}$ 는 업데이트와 업로드 시간의 합으로, 클라이언트 수가 늘어남에 따라 성능이 증가 하지만 클라이언트들의 업로드와 업데이트 시간이 증가하는 관계를 갖는다. 4. 클라이언트가 선택된 후 해당 클라이언트들에 글로벌 모델의 매개변수를 배포한다. 5. 요구된 최소 클라이언트의 수를 만족하였을 경우 클라이언트는 글로벌 모델을 업데이트하고 새로운 매개변수를 업로드 한다. 6. 서버는 업데이트된 매개변수를 평균화하고 글로벌 모델을 평균 모델로 대체한다. 이 과정들을 초기화를 제외한 모든 단계가 원하는 정확도를 달성할 때까지 반복한다.

## III. 성능 평가

### 3.1 학습 환경

FedCS와 FedAD의 성능 평가환경은 다음과 같다. 우선 연합학습을 통해 손글씨 이미지를 인식 하는 학습 환경을 가정하였다. FedAD를 구현하기 위해 keras의 softmax 함수를 사용하는 심층 네트워크를 생성하였고, 데이터 셋은 original NIST dataset을 사용하였다. 또한, 학습에 필요한 하이퍼 파라미터인 epoch, batch size 그리고 optimizer의 learning rate는 각각 5, 50 그리고 0.25로 정의 하였다. 또한, 단말들의 컴퓨팅 능력과 통신능력은 평균 5에서 4000 사이의 랜덤한 값을, 분산이 1인 정규분포를 따른다고 가정하였고, 단말은 1000개가 존재한다고 가정하였다. 비교 기법인 FedCS의 경우 학습 주기를 180초로 설정하였으며, 제안 기법인 FedAD의 경우 업데이트에 필요한 최소 클라이언트 수인  $S_{req}$ 를 10, 20, 30, 40, 50으로 설정하였다. 그에 따라 명칭을 FedAD10, FedAD20 등으로 표시하였다. 비교 기법인 FedCS에서 매 round에서 평균 33개의 클라이언트를 선택하였기 때문에, 180초 이내에 업데이트를 완료할 수 있는 클라이언트는 33개라고 가정 후 FedAD40과 FedAD50은 각각 요구된 클라이언트 수에서 부족한 수를 채우기 위해 180초 이후로 업데이트가 가능한 클라이언트들을 속도 별로 정렬 후 부족한 수만큼 추가로 선택 하였다.

### 3.2 결과

Scheme	Round	Elapsed time (sec)	Accuracy
FedCS	186	33480	0.81
FedAD10	94	8317	0.81
FedAD20	154	19559	0.81
FedAD30	229	38928	0.8
FedAD40	195	41238	0.8
FedAD50	221	55711	0.8

표 1. 실험 결과

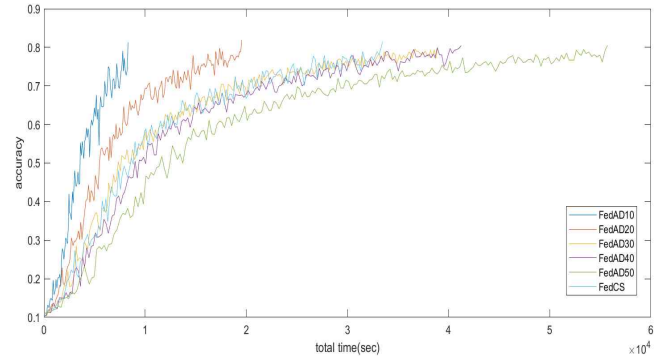


그림 2. 실험 결과

표 1과 그림 2는 실험에 따른 결과를 나타낸다. 실험 결과를 통해 확인 해 보면 0.8의 정확성을 얻기 위해 FedCS의 경우 186 round가 필요했으며, 총 33480초의 시간이 소요 되었다. 반면 FedAD의 기법을 따라 10개의 클라이언트를 선택한 FedAD10은 다른 기법들 중에 가장 좋은 성능을 보였다. 이는 한 round 안에서 선택되는 클라이언트 수를 증가시켜 성능이 좋지 않은 클라이언트가 포함되게 하는 것 보다 성능이 좋은 클라이언트를 한정적으로 선택하여 적은 수의 클라이언트로 업데이트와 업로드를 진행 하는 것이 성능 향상에 도움이 된다는 것을 보여준다.

이후 성능은 FedAD20 > FedCS > FedAD30 > FedAD40 > FedAD50의 순서로 진행 된다. FedCS와 FedAD30은 거의 비슷한 성능을 보여주었다.

## IV. 결론

본 논문은 FedCS의 클라이언트 선택 과정에서 문제가 발생하는 특정 상황에서 효율적인 프로토콜을 제시하는 것을 목표로 하였다. 본 논문의 실험 결과에 따르면 FL의 과정에서 한 round 안에 많은 클라이언트를 선택 하는 방식 보다 최소한의 클라이언트 수를 만족하면 업데이트를 진행하는 방식을 통해 성능이 좋은 클라이언트를 한정적으로 선택할 때 학습의 성능이 높아진다는 것을 알 수 있다.

## ACKNOWLEDGMENT

본 연구는 2019년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(과제번호: 2019R1C1C1004352)

## 참 고 문 헌

- [1] 권오성, 빅데이터 프라이버시 문제 해결할 인공지능 기술 오나, [Online] Available: [http://www.hani.co.kr/arti/science/science\\_general/887030.html](http://www.hani.co.kr/arti/science/science_general/887030.html)
- [2] Takayuki Nishio, Ryo Yonetani, Client Selection for Federated Learning with Heterogeneous Resources in Mobile Edge, Oct. 2018.