

# Performance Enhancement of Federated Learning through Effective Client Management

Kang Tae Woo<sup>†</sup> · Kim Taejoon<sup>††</sup>

## ABSTRACT

In a federated learning with non-IID datasets, training delays and poor performance due to unequalness in datasets among clients are prevalent. To prevent such performance degradation, selecting appropriate clients is a crucial part of training. This paper applies a UCB algorithm based on a virtual queue for client selection in federated learning models. Unlike the traditional UCB algorithm, the proposed client selection method reflects the training progress of federated learning. It can reduce the proposed model's training time. Additionally, to further improve the performance of the model, the exploration of clients through UCB is dynamically adjusted. Once the training is stabilized in the mid-phase, the client exploration is involved to extract additional performance gain. The experimental results demonstrate the benefit of this algorithm such as a reduced training time and an improved model accuracy.

Keywords : UCB, Client Selection, Stale Client, Federated Learning, Adapted Parameter

## 효율적인 클라이언트 관리를 통한 연합학습 성능 향상

강 태 우<sup>†</sup> · 김 태 준<sup>††</sup>

## 요 약

연합학습의 non-IID 데이터셋 환경에서는 학습 딜레이 및 클라이언트 간의 데이터셋 불균형으로 인한 학습 성능 저하 등의 문제점이 있다. 이러한 성능 저하를 방지하기 위해, 적절한 클라이언트를 선택하는 것은 학습에서 중요한 부분 중 하나이다. 이 논문에서는 연합학습 모델에 가상 큐를 기반으로 한 UCB 알고리즘을 적용하여 클라이언트 선택을 진행하는 방식을 사용하고 있다. 또한 기존의 UCB 알고리즘 선택 방식과 다르게 학습 진행 상황에 따라 클라이언트 선택을 조절하는 알고리즘을 제안한다. 먼저 학습 진행 상황을 클라이언트 선택 과정에 반영하여 모델의 학습 시간을 줄일 수 있었다. 추가적으로 모델의 성능 향상을 위해 UCB의 클라이언트 탐색을 동적으로 바꾸는 방식을 제안한다. 학습 중반 이후의 학습이 안정화되면, 클라이언트의 탐색을 조절하여 모델의 성능 개선을 이루었다. 논문의 실험 결과는 해당 알고리즘을 활용하였을 때 학습시간 단축, 모델의 정확도 향상 등의 이점이 있음을 보여준다.

키워드 : UCB, 클라이언트 선택, 오래된 클라이언트 정보, 연합학습, 동적 파라미터

## 1. 서 론

딥러닝 기술은 다양한 분야에서 활발하게 적용되고 있다. 또한 통신기술이 발달에 따라 다양한 전자기기에서 데이터 생성 및 전송이 가능해졌고 이를 활용한 딥러닝의 성능향상은 가속화되었다[1]. 이전에는 데이터를 중앙 서버로 전송하는 방식으로 연구가 진행되었으나 데이터 프라이버시 문제를 해

결하고 방대한 양의 데이터를 효율적으로 활용하기 위하여 연합학습에 대한 연구가 활발히 진행되고 있다[2].

최근 연합학습 연구는 데이터 프라이버시 강화, non-IID (independent and identically distributed) 데이터셋에서 모델 성능 개선, 효율적인 클라이언트 관리와 통신 비용 절감 방법에 초점을 맞추고 있다[3]. 특히, 개인정보 보호를 위한 암호화 기술의 통합, 모델 업데이트 과정에서의 최적화, 그리고 다양한 애플리케이션과 환경에서의 적용 등이 활발히 연구되고 있다[4]. 또한 학습에 참여하는 클라이언트를 효율적으로 선택하여 정확도와 학습속도를 개선하는 연구도 진행되고 있으며 이 중에서 가장 많이 사용되는 방법은 각각의 로컬 파라미터 값을 사용하여 평균값을 구하고 그 값으로 글로벌 모델을 만드는 FedAvg (Federated Average) 기법이다[5].

※ 이 논문은 2023년도 교육부의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(RS-2023-00244014).

† 비 회 원 : 충북대학교 정보통신공학부 석사과정

†† 종 신 회 원 : 충북대학교 정보통신공학부 교수

Manuscript Received : August 30, 2024

First Revision : November 14, 2024

Accepted : December 5, 2024

\*Corresponding Author : Kim Taejoon(ktjcc@chungbuk.ac.kr)

연합학습 클라이언트 관리에 있어서는 non-IID 데이터셋에서 성능 향상, 클라이언트 선택, 그리고 오래된 그래디언트(stale gradient)를 다루는 것이 주요한 이슈들이다[6]. 다수의 클라이언트에서 데이터셋의 non-IID 특성은 모델의 정확도를 낮추고 모델 수렴성을 떨어뜨릴 수 있다[7]. 이 때문에 non-IID 데이터셋으로 이루어진 클라이언트들 중 매 학습마다 적절한 클라이언트를 선택하는 것은 연합학습에 있어서 중요한 요소이다[8]. 추가적으로, 연합학습은 학습된 로컬 모델들을 집계하여 글로벌 모델 생성하므로, 클라이언트 선택이 전체 연합학습 모델 성능에 미치는 영향은 매우 크다고 할 수 있다.

연합학습은 동기식 연합학습과 비동기식 연합학습으로 나눌 수 있는데 동기식 연합학습의 경우 학습 반복이 시작되면 연합학습에 참여하는 모든 클라이언트가 서버에서 글로벌 모델을 새롭게 수신한 후 학습을 진행하기 때문에 모델 생성을 지연시키는 클라이언트들이 존재하게 되며 이를 stragglers라 한다[9]. 반면 비동기식 연합학습은 학습 중인 클라이언트가 중앙 모델을 새롭게 전달받지 않기 때문에 stragglers 현상에서 자유롭다. 이렇게 연합학습에서는 여전히 non-IID 데이터셋에 따른 성능 감소는 극복해야 할 중요한 문제이다.

이 논문에서는 이러한 non-IID 데이터셋으로 생기는 문제 해결을 위해 학습에 참여할 클라이언트를 선택하는 새로운 방식을 제안한다. 제안된 방식은 매 라운드마다  $K$ 개의 채널에 적합한 클라이언트를 선택하고 로컬학습을 진행한 뒤 글로벌 모델을 집계하는 방식으로 진행된다.

논문 [10]에서는 클라이언트 선택 시 MAB (Multi-Armed Bandit)을 기반으로 클라이언트를 선택하여 활용하고 있다. 또한 [11]에서는 클라이언트를 선택할 때 UCB (Upper Confidence Bound) 알고리즘을 사용하여 매 라운드마다 클라이언트를 무작위로 선택하지 않고 클라이언트 성능, 학습 시간 등에 따라 클라이언트의 선택횟수를 다르게 하여 효율적인 학습을 진행한다. 논문 [12], [13]에서는 특정 클라이언트들을 집중적으로 선택하는 현상을 방지하기 위해 가상 큐 값을 설정하여 가상 큐 값과 UCB 점수의 결합으로 최적의 클라이언트를 선택하는 방식을 사용한다.

본 논문에서는 학습이 진행되는 정도에 따라 클라이언트 선택에 방식을 변경하는 방식을 도입하여 모델의 안정성을 높이고, 모델의 정확도 향상 및 학습시간의 감소 등의 이점을 달성한다. 학습 초기에는 모델 학습이 안정화되지 않았기 때문에 클라이언트들의 정보가 부족하다. 이 경우에는 UCB의 탐색을 통하여 클라이언트를 선택하는 것이 효율적이지 못하다. 그러므로 모델 학습이 어느 정도 안정화될 때까지 무작위로 클라이언트를 선택하고, 모델이 어느 정도 안정화되면 UCB 기반으로 클라이언트를 선택하도록 전환하여 모델을 빠르게 안정화시킬 수 있다. 또한 모델이 수렴에 가까워지면 클라이언트에 대한 정보가 충분하므로 공정정보보다는 성능에 도움을 주는 클라이언트를 선택하는 비율을 높여 추가적인 성능

향상을 도모하게 된다.

## 2. 관련 연구

클라이언트 선택은 non-IID 데이터셋으로 구성된 클라이언트 학습 시 결정짓는 중요한 요소 중 하나이다. 기존의 연구에서는 클라이언트를 무작위로 선택하거나 단순 성능 기반으로 선택하는 방법이 사용되었다. [11]은 리소스 제약이 있는 상황에서 Multi-Armed Bandit (MAB) 접근 방식을 통해 클라이언트를 선택하는 방안을 제안하여, 클라이언트의 성능을 최적화하고 학습 속도를 개선하였다. 이 방법은 클라이언트의 리소스 변동성을 고려하여 학습 효율성을 높이는 데 기여하였다.

[12], [13]는 UCB를 통해 클라이언트의 선택 횟수를 다르게 조정하여, 클라이언트의 성능과 학습 시간을 고려한 선택 방법을 제안하였다. 이를 통해 연합학습의 성능을 향상시키고, 클라이언트의 자원 사용을 효율적으로 관리하였다. 또한 [14]는 초기 데이터 수집 단계에서 클라이언트의 성능에 대한 불확실성을 빠르게 줄이기 위해 UCB를 클라이언트 선택을 사용하고 있고 [15]에서는 UCB를 통해 클라이언트들이 탐색을 통해 얻은 보상을 기반으로 최적의 선택을 하여 클라이언트 선택에 사용하고 있다.

UCB 알고리즘은 불확실한 상황에서 최적의 선택을 하도록 돕는 기법으로, 주로 강화 학습과 Multi-Armed Bandit 문제에서 사용된다. 이 알고리즘은 각 행동의 기대 보상에 대한 신뢰 구간을 설정하여 선택지를 평가한다. 각 선택지에 대해 얻은 보상의 평균값과 함께 해당 선택지의 불확실성을 고려하는 것이다.

UCB는 기대 보상과 선택지의 불확실성을 동시에 고려하여 보다 정보에 기반한 결정을 내리도록 한다. 결과적으로, UCB 알고리즘은 장기적인 성과를 극대화하는 데 효과적이다.

연합학습에서 non-IID 데이터셋의 특성은 모델의 정확도와 수렴성에 큰 영향을 미친다. 이 논문에서는 non-IID 데이터셋으로 인해 발생하는 성능 저하 문제를 방지하기 위한 효율적인 클라이언트 선택에 대해 다루었으며, UCB 클라이언트 선택 기반으로 하고 학습 진행 상황에 따라 동적인 UCB 클라이언트 선택 방식을 제시하여 기존 UCB 클라이언트 선택 방식보다 더 나은 성능을 보일 수 있다는 것을 보여준다.

## 3. System Model

### 3.1 $K$ -selection 연합학습

[16]에서 중앙 서버의 전역 모델을 업데이트하는 기본 모델은 다음과 같이 정의된다.

$$\omega_{j+1} = \omega_j - \frac{\eta}{K} \sum_{i=1}^K g(\omega_{j,i}, \epsilon_{j,i}) \quad (1)$$

여기서  $\omega_j$ 는  $j$ 번째 라운드에서의 중앙모델을 뜻하며  $g(\omega_{j,i}, \epsilon_{j,i})$ 은  $j$ 번째 라운드에서  $i$ 번째 클라이언트를 통해 얻어진 그래디언트를 뜻한다.  $\omega_{j,i}$ 는  $j$ 번째 라운드에서  $i$ 번째 클라이언트를 통해 얻어진 지역 모델을 뜻한다.  $\epsilon_{j,i}$ 는  $j$ 번째 라운드에서 얻어진 클라이언트  $i$ 의 그래디언트를 갱신하기 위해 활용된 데이터 샘플의 집합을 의미한다.  $\eta$ 는 모델의 학습률을 뜻한다. 본 논문에서는 이 방식을 변형하여  $P$ 개의 클라이언트 중 매 반복마다  $K$ 개의 클라이언트를 선택하고, 이 클라이언트들을 활용하여 전역 모델을 업데이트한다.

$$\omega_{j+1} = \omega_j - \eta \sum_{i=1}^K g(\omega_{j,i}, \epsilon_{j,i}) p_{j,i} \quad (2)$$

이 논문에서는 [16]의 모델 업데이트 방식의 일부를 사용하였다. (1) 방식의 기본 모델로 업데이트를 진행하면 그래디언트의 오래된 정도와 non-IID 데이터에 따라 비합리적인 업데이트 방식으로 진행되기 때문에 가중치  $p_{j,i}$ 를 사용한다.  $p_{j,i}$ 는  $j$ 번째 라운드에서 클라이언트  $i$ 의 가중치이고  $\sum_i p_{j,i} = 1$ 이다. 지역 그래디언트와 추정된 중앙 그래디언트 사이의 유사한 정도를 수치화 시켜 계산한다. 또한 학습률  $\eta_j$ 를 학습 반복마다 그래디언트의 오래된 정도를 기반으로 조절한다.

Fig. 1에서는 클라이언트의 전체 개수  $P$ 가 5고 채널의 개수가 2인 경우를 가정하고 업데이트 방식을 설명한다. 학습이 시작되면 모든 클라이언트에게 글로벌 모델을 전송하고, 각각의 클라이언트는 로컬 학습을 시작하게 된다. 매 반복에 각 채널마다 클라이언트를 선택하여 총  $K$ 개의 클라이언트의 모델들을 활용하여 새로운 글로벌 모델을 만들게 된다. 이 과정이 마무리되면 학습에 참여한  $K$ 개의 클라이언트에게 새로운 글로벌 모델 정보를 전달한다. 이 그에서 노란색 화살표는 로컬학습이 완료된 시점을 뜻하고 파란색 화살표는 로컬학습이 진행중임을 의미한다. 먼저 라운드마다 학습에 참여시킬 클라이언트를 선택하고 선택된  $K$ 개의 클라이언트의 로컬학습이 전부 완료되면 학습이 완료된 그래디언트를 통해 글로벌 그래디언트를 새롭게 업데이트한 후 학습에 참여했던 클라이언트에 전달한다.

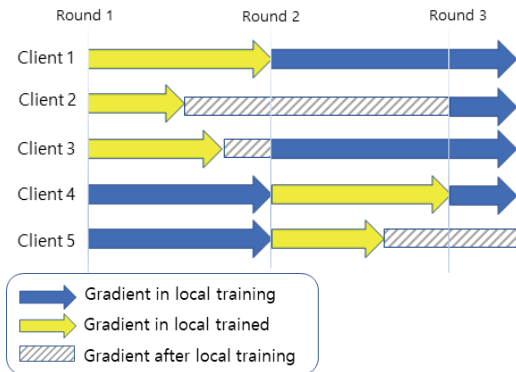


Fig. 1.  $K$ -Selection with  $K=2, P=5$ .

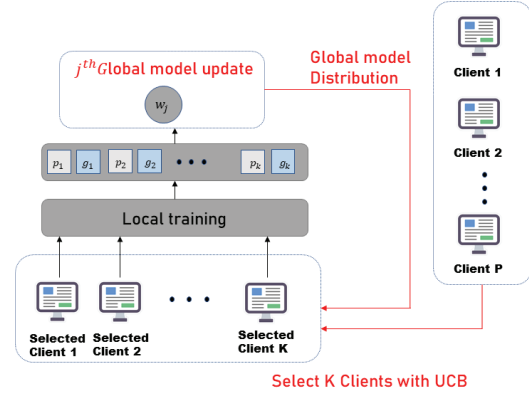


Fig. 2.  $K$ -Selection with detailed procedures.

Fig. 2에서는 연합학습에서 UCB 알고리즘을 사용하여 클라이언트를 선택하고 매 반복마다 이들이 학습한 로컬 그래디언트들을 집계하여 새로운 전역 모델을 만드는 과정을 보여준다. 먼저 총  $P$ 개의 클라이언트중  $K$ 개의 클라이언트를 선택하고, 선택된 클라이언트들의 로컬학습을 통하여 만들어진 그래디언트값을 추정된 중앙 그래디언트값  $g$ 와의 유사도에 따라 가중치를 부여하여 계산한 뒤 새로운 전역 모델을 만들고 학습에 참여했던 클라이언트들에게 전역 모델을 분배한다.

### 3.2 UCB 클라이언트 선택

해당 논문에서는 클라이언트 선택시 UCB 방식을 활용하는 방식을 인용하고 있다. UCB의 주요 개념은 탐색(Exploration)과 활용(Exploitation) 사이의 균형을 맞추는 것이다. 여기서 탐색이란, 아직 충분히 시도해보지 않은 옵션을 더 시도해보려는 것이고, 활용은 현재까지 얻은 보상이 높은 옵션을 반복적으로 선택하려고 하는 것을 의미한다.

이 논문에서는 [12],[13]의 클라이언트 방식을 인용하였다. 먼저 클라이언트 선택시 클라이언트가 선택되지 않은 경우 해당 클라이언트의 가상 큐길이를 만들고 그 길이와 ucb 점수값의 최댓값을 가지는 클라이언트를 우선적으로 선택하여 학습을 진행하게 된다. 해당 식은 아래와 같다.

$$S_k(t) = \operatorname{argmax}_k (Q_k(t) + \beta \hat{\mu}_k(t)) \quad (3)$$

위 식에서 점수값  $\hat{\mu}_k(t)$ 는 매 라운드마다 바뀌며 아래와 같은 식으로 정의된다.

$$\hat{\mu}_k(t+1) = \hat{\mu}_k(t) + \sqrt{\frac{3 \ln t}{2d_k(t)}} \quad (4)$$

위 식과 같이 클라이언트 선택시 UCB 알고리즘을 사용하고 UCB 점수값이 높은 클라이언트를 우선 선택하므로, 평균 성과가 좋은 클라이언트이면서도 가상 큐를 활용하여 선택 횟수가 적었던 클라이언트에 대해 우선 탐색을 수행하게 된다. 이

Table 1. Notations

Notations	Description
$\eta$	Learning rate
$\hat{\mu}_k$	Client $k$ 's UCB score
$\bar{g}$	Global gradient
$D$	Total number of data samples
$P$	Total number of clients
$j$	Iteration index
$\omega$	Model Parameter
$g$	Local gradient
$K$	Number of participating clients per round
$d_k$	Number participated rounds for client $k$
$\tau$	Client's Staleness
$\beta$	Parameter of UCB's score

논문에서는 이러한 가상 큐 기반의 UCB 클라이언트 선택을 기반으로 학습 진행 상황에 따른 동적인 클라이언트 선택 방식을 제안하여 기존의 방식보다 더 높은 성능을 보여준다.

Table 1에서는 본 논문에서 사용될 용어들을 정리했다.

#### 4. 성능 향상을 위한 클라이언트 선택 알고리즘

##### 4.1 클라이언트 선택 방식에 따른 성능 향상 알고리즘

이 논문에서는 클라이언트 선택 방식에 따른 학습시간의 감소, non-IID 데이터셋의 영향 완화와 최종적으로는 학습 정확도 향상을 목표로 3가지 알고리즘을 제안한다. 먼저 무작위로 선택되는  $K$ 개의 클라이언트를 사용한 업데이트 방식에서 가상큐 기반 UCB 클라이언트를 선택방식으로 변경한다. 또한 기존 가상 큐 기반의 UCB 클라이언트 선택 방식을 동적으로 변경시켜 학습 상황에 따른 성능 악화 방지를 위한 알고리즘을 제안한다.

##### 4.2 $K$ -Selection 기법

먼저  $K$ -Selection은 [16]를 변형하여 매 라운드 선택되는 클라이언트를 가상 큐 기반의 UCB 클라이언트 선택방식으로  $K$ 개의 클라이언트를 선택하여 전역 모델 업데이트를 수행하는 방식을 제안한다.  $K$ -Selection은 아래와 같은 식으로 정리된다.

먼저 매 라운드마다 선택되지 않은 클라이언트  $k$ 에 대하여 가상 큐 길이를 총 클라이언트 수  $P$ 를 총 데이터수  $D_k$ 로 나눈 값만큼 증가시킨다. 매 반복마다 클라이언트  $k$ 의 UCB 점수  $\hat{\mu}_k$

를  $\hat{\mu}_k(t+1) = \hat{\mu}_k(t) + \sqrt{\frac{3\ln t}{2d_k(t)}}$ 를 통하여 업데이트 한다. UCB 점수와 가상 큐 길이에 따라 클라이언트를 선택한다.  $d_k(t)$ 는 라운드  $t$ 까지 클라이언트  $k$ 가 선택된 횟수이다. 그 후 선택된 클라이언트의 로컬 그래디언트 값들을 사용하여 새로운 글로벌

##### Algorithm 1: $K$ -selection

Input

$D_k$ : size of dataset for client  $k$ ,  $\omega_j$ : round  $j$ th model  
 $\tau$ : staleness  $\tau_{\min}$ : minimum staleness,  $j$ : iteration  
 $g(\omega)$ : local gradient,  $d_k$ : selected count for client  $k$   
 $Q_k$ : client  $k$ 's length of virtual-Q,  $S_k(t)$ : selected client  
 $\eta_j$ :  $j$ th iteration's learning rate

for  $j = 0$  do

$S_k(t) = \operatorname{argmax}_k(Q_k(t) + \beta\hat{\mu}_k(t))$  // Client selection

for  $k = 1$  to  $P$

If client  $k$  is not selected

$$\Delta Q = \max\left(\frac{P}{D_k}, 0\right)$$

else:

$$\Delta Q = \frac{P}{D_k} - 1$$

$Q_k += \Delta Q$

$$\hat{\mu}_k(t+1) = \hat{\mu}_k(t) + \sqrt{\frac{3\ln t}{2d_k(t)}}$$

$k = k + 1$

$$\bar{g}(\omega_j) = g(\omega_j) + \alpha\bar{g}(\omega_{j-1})$$

$$\eta_j = \eta_0 \frac{1}{\tau_{\min} + 1}$$

$$\omega_{j+1} = \omega_j - \frac{\eta}{K} \sum_{i=1}^K g(\omega_{j,i}, \epsilon_{j,i})$$

$j = j + 1$

end for

별 모델을 만들고 학습률  $\eta$ 을 클라이언트의 staleness에 따라 갱신한다. staleness는 로컬 학습을 마치고 클라이언트가 선택 되서 글로벌 모델 업데이트에 참여하기 전까지의 지연된 정도를 의미한다. 마지막으로 만들어진 글로벌 그래디언트를 학습에 참여한 클라이언트들에게 전송해준다.

##### 4.3 학습 초기 성능을 위한 Hybrid-Random-UCB 설정

논문 [17]에서는 학습 초기의 클라이언트에 대한 정보가 없는 경우의 학습 성능 감소에 대한 대안으로 효율적인 심층 신경망 활용 방법을 제안하고 있다. 또한 논문 [18]에서는 학습 초기에 개별 클라이언트의 특성을 고려하여 모델을 개별적으로 학습하여 학습 초기 성능 향상을 제안하고 있다. 이처럼 non-IID 데이터셋으로 이루어진 클라이언트들을 학습시킬 때 학습 초기 클라이언트에 대한 정보가 부족하다. 그러므로 손실값이 안정화되기 이전에 클라이언트를 인위적으로 선택할 경우 잘못된 클라이언트 선택이 발생할 수 있고, 이는 학습 성능의 악화를 유발할 수 있다.

알고리즘 2에서는 이를 방지하기 위해 학습진행 상황에 따라 클라이언트를 선택하는 방식을 바꿔준다. 학습 초기에는 클라이언트를 무작위로 선택하고, 학습이 안정화 됨에 따라 UCB 클라이언트 선택 방식으로 바꿔준다. 이는 초기 탐색에 따른 학습 시간 증가와 이에 따른 성능 열화를 경감시키는 데 도움을 준다. 해당 방식은 아래와 같은 식으로 정리된다.

Algorithm 2: Hybrid Random-UCB Selection

---

Input  
 $\hat{\mu}_k(t) = 0$ ,  $R(k)$ : randomly selected client  
 $\alpha_{Ref} = \frac{e^K}{\mu \ln N}$   
 $K$ : numbers of client,  
 $N$ : numbers of data samples  
 $L_t$ : loss in round  $t$

---

for  $j = 0$  do  
 if  $L_t \geq \alpha_{Ref}$   
 $S_k(t) = R(k)$   
 else  $L_t < \alpha_{Ref}$   
 $S_k(t) = \operatorname{argmax}_k(Q_k(t) + \beta \hat{\mu}_k(t))$   
 $j = j + 1$   
end for

---

$L_t$ 라는 임계값을 통해 손실값이  $L_t$ 보다 크면  $S_k(t) = R(k)$ 와 같이 클라이언트를 무작위로 선택하게 된다. 이후 손실값이  $L_t$ 보다 작게되면 모델이 안정화 되었다고 가정하고  $S_k(t) = \operatorname{argmax}_k(Q_k(t) + \beta \hat{\mu}_k(t))$ 와 같이 가상큐와 UCB를 기반으로 한 클라이언트 선택 방식으로 바꾸면서 학습을 진행시킨다.

#### 4.4 UCB 가중치 조절을 위한 Adatped- $\beta$ 설정

제안하는 클라이언트 선택 방식에선 가상 큐 값과 UCB 알고리즘을 사용하여 특정 클라이언트가 과하게 선택되어 편향된 결과값을 도출하게 되는 것을 방지한다. 학습이 진행되고 학습 중반 이후에 모델이 수렴에 가까워지면 과적 합이 일어날 확률이 적고 클라이언트의 공정성을 위한 가상큐의 가치 보다는 성능이 높은 클라이언트의 탐색 및 선택의 비중을 높여주어 학습 성능의 향상을 이끌 수 있다. 알고리즘 3에서는 클라이언트의 성능에 따라 클라이언트의 선택하게 하는 파라미터 값을 동적으로 변경시키며 학습을 진행하는 방식을 제안

 Algorithm 3: Adapted- $\beta$  Selection

---

Input  
 $t$ : round index,  
 $\Delta L$ : proportional variation of loss  
 $L_t$ : value of loss at round  $t$   
 $\beta$ : parameter that determine UCB weighting  
 $F_{flat}$ : degree of parameter adjustment

---


$$\Delta L = \frac{L_t - L_{t-d}}{L_{t-d}}$$

for  $j = 0$  do  
 if  $\Delta L < F_{flat}$   

$$\beta(t) = \beta(t-1)(1 + \frac{P}{D})$$
  

$$S_k(t) = \operatorname{argmax}_k(Q_k(t) + \beta \hat{\mu}_k(t))$$
  
 $j = j + 1$   
end for

---

하고 있다. 손실값의 변화량  $\Delta L$ 이 설정한 임계값  $F_{flat}$ 보다 작게되면 학습이 수렴에 가까워진 상황으로 간주하고 UCB 점수에 따른 파라미터 값  $\beta$ 를 학습이 종료될 때까지 학습에 쓰이는 전체 클라이언트 수에 비례하여  $\beta(t) = \beta(t-1)(1 + \frac{P}{D})$ 와 같이 점진적으로 증가시키는 방법으로 진행하였고 아래와 같은 식으로 정리된다.

이 방법은 학습 중반 이후 손실값의 변화량이 평단화 되고 학습이 수렴에 가까워지면,  $\beta$  값을 점진적으로 증가시켜 가상 큐 길이에 따른 클라이언트 공정성보다는 높은 성능의 클라이언트를 선택하는 비중을 높게 하여 학습 성능을 높인다.

## 5. 실험 결과 및 분석

### 5.1 실험 환경

클라이언트 선택방식에 따른 성능 변화를 측정하기 위하여 pysyft로 구축된 연합학습 환경에서 실험을 진행하였다. 데이터셋은 EMNIST 데이터셋[17]을 사용하였으며 0-9 까지의 라벨을 가지는 데이터 70,000개로 이루어진 데이터셋에서 non-IID 데이터셋을 활용하기 위해 한 클라이언트당 2000 - 4000개 사이의 데이터 수와 1 - 10 사이로 랜덤하게 지정되는 클래스 수로 실험을 진행하였다. Iteration은 1000회 진행하였고 학습률은 0.005로 진행하였으며  $K$ 값은 3, 전체 클라이언트 갯수  $F$ 는 100을 설정하였다. 이 실험에서는 간단한 2개의  $5 \times 5$  합성곱 신경망(CNN)을 활용한다. 첫번째는 32개의 채널, 두 번째는 64개의 채널을 가지고 ReLU 활성화 함수와  $2 \times 2$  최대 풀링을 활용하며 이후 결합층에서 활성화 함수를 사용하는 소프트맥스(softmax) 출력 계층을 가진다.

### 5.2 $K$ -selection 클라이언트 선택방식 적용

해당 논문에서는 클라이언트를 UCB 점수 및 가상 큐를 활용하여 선택함으로써 클라이언트의 성능 및 데이터 수에 따라 선택횟수를 적절히 조절하여 학습 성능을 증가시켰다. Fig. 3에서는 10개의 클라이언트를 무작위로 선택하였을 경우와  $K$ -selection 및 알고리즘 2,3을 적용시켰을 경우의 클라이언트

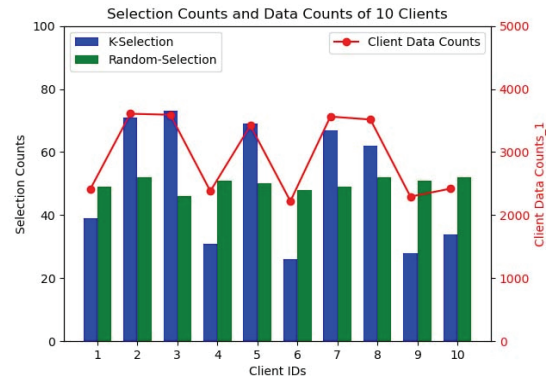


Fig. 3. Selection counts with data counts of 10 clients.

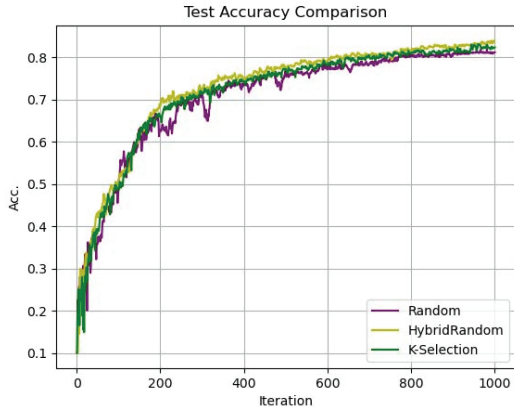


Fig. 4. Test accuracy comparison among Random, Hybrid Random-UCB and  $K$ -selection.

가 선택되는 비율을 제시하였다. 클라이언트 별 데이터수와 클라이언트 선택 횟수의 관련성을 나타내기 위해 1에 가까울수록 두 변수간에 강한 상관관계가 있음을 의미하는 피어슨 상관관계수(Pearson correlation coefficient)를 구하였다. 상관관계 값은 0.986으로 랜덤으로 선택하는 경우 -0.573와 큰 차이를 보였다.

### 5.3 Hybrid Random-UCB Selection의 적용

Fig. 4에서는 알고리즘 2에 제안된 Hybrid Random-UCB Selection과 무작위 클라이언트 선택,  $K$ -selection을 비교하였다. Non-IID 데이터셋에서 학습 초기 클라이언트 선택을 위한 학습정보가 거의 없기 때문에 잘못된 클라이언트 선택으로 불필요하게 학습시간이 길어지고 컴퓨팅 자원이 낭비될 수 있다.  $\alpha_{Ref} = \frac{e^K}{\mu \ln N}$  에서  $\mu$ 값은 0.5로 설정하였다. 그래프를 보면 알고리즘 2이 기존 가상 큐와 UCB 알고리즘만을 사용한  $K$ -Selection 선택방식보다 더 나은 테스트 정확도 성능을 달성하는 것을 관찰할 수 있다.

### 5.4 UCB 가중치 조절을 위한 Adapted- $\beta$ 적용

Fig. 5에서는 알고리즘 3에 제안된 Adapted- $\beta$ 와 무작위 클라이언트 선택,  $K$ -selection을 비교하였다. 실험에서  $F_{fit} = 0.1$ ,  $d = 10$ 로 설정하였다. 그래프를 보면 알고리즘 3이 기존 UCB 알고리즘만 사용한 선택 방식보다 더 나은 테스트 정확도를 달성하는 것을 관찰할 수 있다. 이는 학습 중반 이후 수렴에 가까워질 때, 높은 성능을 보인 클라이언트를 선택하는 비율을 높여 전체 학습 성능에 도움을 줄 수 있다는 것을 보여준다.

### 5.5 Hybrid Random-UCB Selection with Adapted- $\beta$ 적용

Fig. 6에서는 무작위 선택, 알고리즘 2, 3를 같이 사용하였을 경우와  $K$ -selection을 비교하였다. 알고리즘 2, 3를 같이 적용하였을 때 더 높은 성능을 달성하였다. 이는 학습 진행 상황에 따라 클라이언트 선택 방식을 조절함으로써 학습 정확도

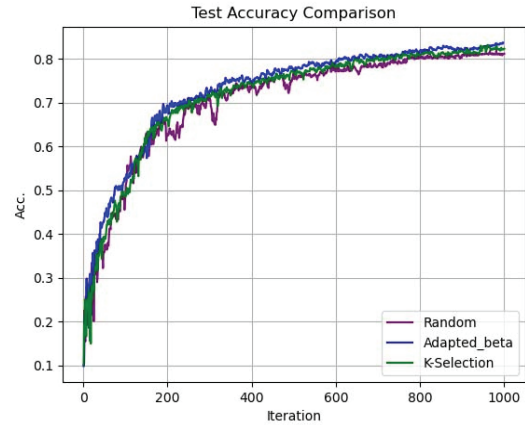


Fig. 5. Test accuracy comparison among Random, Adapted- $\beta$  and  $K$ -selection.

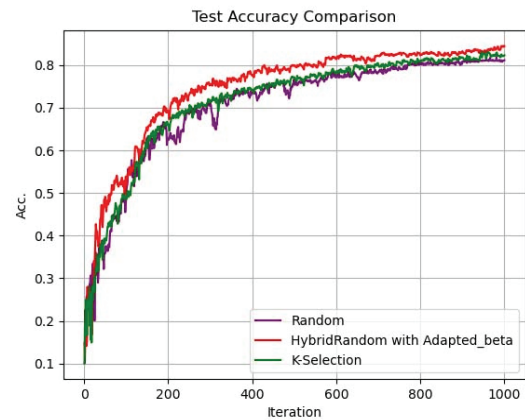


Fig. 6. Test accuracy comparison between Hybrid Random-UCB combined with adapted- $\beta$  and  $K$ -selection.

에 영향을 줄 수 있다는 것을 보여주고 있으며 이는 제안된 알고리즘들이 학습 성능 향상에 도움을 줄 수 있다는 것을 재확인시켜준다.

최종적으로 모든 알고리즘을 결합하여 결과값을 측정하였을 때, 또한  $K$ -selection을 통한 클라이언트 선택 방법에 비해 0.1 정도 높은 test accuracy 값을 보였다. 이를 통해 적절한 클라이언트 선택을 통해 학습 정확도 향상의 효과를 보일 수 있다는 것을 알 수 있다.

### 5.6 Iteration에 따른 학습시간 비교

Table 2에서는 iteration에 따른 학습시간을 비교하였다. 표를 보면 같은 iteration 값에 따라 학습시간이 차이나는 것을 볼 수 있다. 이를 통해 제안된 알고리즘들이 학습 시간 감소의 효과를 보일 수 있다는 것을 알 수 있다.

마지막으로 Table 3에서는 각 알고리즘들의 초기  $\beta$  값에 따른 마지막 10개 iterations의 test accuracy 값의 평균을 나타내었다. 전반적으로  $\beta$  값이 0.5 일때 가장 좋은 성능을 보였다.

Table 2. Comparison of Training Time by Iteration.

Iteration	300	500	700	1000
Random	269s	349s	561s	842s
K-Selection	260s	335s	550s	831s
Hybrid Random-UCB	248s	313s	526s	816s
Adapted- $\beta$	254s	331s	521s	809s
Hybrid with Adapted- $\beta$	252s	309s	507s	798s

Table 3. Last 10 test accuracy comparison with varying  $\beta$ .

Test Accuracy	$\beta = 0.25$	$\beta = 0.5$	$\beta = 1$	$\beta = 2$
K-Selection	0.8221	0.8287	0.8254	0.8233
Hybrid Random-UCB	0.8230	0.8338	0.8319	0.8260
Adapted- $\beta$	0.8241	0.8321	0.8279	0.8247
Hybrid with Adapted- $\beta$	0.8275	0.8393	0.8332	0.8279

## 6. Conclusion

이 논문에서는 비동기 연합학습 환경에서 개선된 클라이언트 선택을 통해 학습 성능을 높이는 것을 목표로 새로운 알고리즘을 제안하고 있다. 이를 통하여 학습시간 감소, 정확도 향상을 달성할 수 있었다. 제안된 방식에서는 학습 진행 상황에 따라 클라이언트 선택 방식을 동적으로 변화시켜 성능 향상을 달성하였으며, 모델이 수렴에 가까워지는 학습 중후반부터는 클라이언트 탐색의 영향성을 크게 하여 추가적인 정확도 향상 및 학습 시간 감소를 달성하였다.

## References

- [1] K. Wei et al., "Federated Learning with Differential Privacy: Algorithms and Performance Analysis," *IEEE Transactions on Information Forensics and Security*, Vol.15, No.6, pp.3454-3469, 2020.
- [2] M. Ammad-Ud-Din et al., "Federated Collaborative Filtering for Privacy-Preserving Personalized Recommendation System," *arXiv preprint arXiv:1901.09888*, 2019.
- [3] F. Sattler, S. Wiedemann, K. R Müller, and W. Samek, "Robust and Communication-Efficient Federated Learning From Non- i.i.d. Data," *IEEE Transactions on Neural Networks and Learning Systems*, Vol.31, No.9, pp.3400-3413, 2019.
- [4] X. Shen et al., "Verifiable Privacy-Preserving Federated Learning Under Multiple Encrypted Keys," *IEEE Internet of Things Journal*, Vol.11, No.3, pp.3430-2445, 2024.
- [5] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp.1223-1234, 2016.
- [6] H. Wang, Z. Kaplan, D. Niu, and B. Li, "Optimizing Federated Learning on Non-IID Data with Reinforcement Learning," *Proceedings of IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, 2020.
- [7] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated Learning with Non-IID Data," *Proceedings of the NeurIPS Workshop on Federated Learning*, 2018.
- [8] W. Xia, T. Q. S. Quek, K. Guo, W. Wen, H. H. Yang, and H. Zhu, "Multi-Armed Bandit-Based Client Scheduling for Federated Learning," *IEEE Transactions on Wireless Communications*, Vol.19, No.11, pp.7108-7123, 2020.
- [9] T. T. Vu, D. T. Ngo, H. Q. Ngo, M. N. Dao, N. H. Tran, and R. H. Middleton, "Straggler Effect Mitigation for Federated Learning in Cell-Free Massive MIMO," *Proceedings of ICC 2021 - IEEE International Conference on Communications*, 2021.
- [10] Y. Chen, Y. Ning, M. Slawski, and H. Rangwala, "Asynchronous Online Federated Learning for Edge Devices with Non-IID Data," *Proceedings of IEEE International Conference on Big Data (Big Data)*, 2020.
- [11] N. Yoshida, T. Nishio, M. Morikura, and K. Yamamoto, "MAB-Based Client Selection for Federated Learning with Uncertain Resources in Mobile Networks," *Proceedings of IEEE Globecom Workshops (GC Wkshps)*, 2020.
- [12] H. Zhu, Y. Zhou, H. Qian, Y. Shi, X. Chen, and Y. Yang, "Online Client Selection for Asynchronous Federated Learning with Fairness Consideration," *IEEE Transactions on Wireless Communications*, Vol.22, No.4, pp.1234-1245, 2023.
- [13] H. Zhu, J. Kuang, M. Yang, and H. Qian, "Client Selection with Staleness Compensation in Asynchronous Federated Learning," *IEEE Transactions on Vehicular Technology*, 2023.
- [14] J. Tong, Z. Chen, L. Fu, J. Zhang, and Z. Han, "From Learning to Analytics: Improving Model Efficacy with Goal-Directed Client Selection," *IEEE Transactions on Mobile Computing*, 2024.
- [15] C. Shi, W. Xiong, C. Shen, and J. Yang, "Reward Teaching for Federated Multiarmed Bandits," *IEEE Transactions on Signal Processing*, 2023.
- [16] Z. Zhou, Y. Li, X. Ren, and S. Yang, "Towards Efficient and Stable K-Asynchronous Federated Learning with

Unbounded Stale Gradients on Non-IID Data,” *IEEE Transactions on Parallel and Distributed Systems*, Vol.33, No.12, pp.3291-3305, 2022.

- [17] C. Wang, Q. Wu, Q. Ma, and X. Chen “A Buffered Semi-Asynchronous Mechanism with MAB for Efficient Federated Learning,” *Proceedings of the International Conference on High Performance Big Data and Intelligent Systems (HDIS)*, 2022.
- [18] A. Fallah, A. Mokhtari, and A. Ozdaglar, “Personalized Federated Learning with Theoretical Guarantees: A Model-Agnostic Meta-Learning Approach,” *Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS)*, 2020.
- [19] G. Cohen, S. Afshar, J. Tapson, and A. V. Schaik, “EMNIST: Extending MNIST to Handwritten Letters,” *Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN)*, 2017.



강 태 우

<https://orcid.org/0009-0006-8061-8741>

e-mail : twoo96@chungbuk.ac.kr

2023년 2월 충북대학교

정보통신공학부(학사)

2023년 3월~현 재 충북대학교

정보통신공학부 석사과정

관심분야: 연합학습, 머신러닝



김 태 준

<https://orcid.org/0000-0001-6326-2559>

e-mail : ktjcc@chungbuk.ac.kr

2003년 연세대학교 전자공학과 공학사

2011년 한국과학기술원 전기,전자공학과  
공학박사

2011~2013년 한국전자통신연구원

선임연구원

2013년~현 재 충북대학교 전자정보대학 정보통신공학부 교수

관심분야: 연합학습, 머신러닝, 네트워크 최적화