

# 연합학습 개방형 플랫폼의 발전과 문제점에 대한 체계적 비교 분석<sup>☆</sup>

## Advances and Issues in Federated Learning Open Platforms: A Systematic Comparison and Analysis

김진수<sup>1</sup>      양세모<sup>2</sup>      이강윤<sup>2</sup>      이광기<sup>1\*</sup>  
JinSoo Kim      SeMo Yang      KangYoon Lee      KwangKee Lee

### 요약

연합학습이 현대 인공지능 연구에 큰 패러다임을 가지고 오면서 다양한 분야의 연구에서 연합학습을 접목시키기 위한 노력을 하고 있다. 하지만 연합학습 적용을 위한 연구자들은 자신의 상황과 목적에 맞는 연합학습 프레임워크와 벤치마크 툴을 선택해야 하는 문제에 직면한다. 본 연구는 실제 연합학습을 적용하는 연구자의 상황을 고려한 연합학습 프레임워크 및 벤치마크 툴의 선택 가이드라인 제시를 목표로 한다. 특히, 본 연구에서는 3가지의 주요한 기여점이 존재한다. 첫째, 연합학습을 적용하는 연구자의 상황을 연합학습의 목표와 결합하여 일반화하고, 각 상황에 적합한 연합학습 프레임워크의 선택 가이드라인을 제안한다. 두번째, 연구자에게 연합학습 프레임워크를 각각의 특징과 성능비교를 통해 선택의 적합성을 보여준다. 마지막으로, 현존하는 연합학습 프레임워크의 한계와 실제 연합학습 운영을 위한 방안, 특히 생명주기 관리에 대한 플랫폼의 구조에 대해 제안한다.

☞ 주제어: 연합학습, 분산학습, 연합학습 프레임워크, 오픈소스

### ABSTRACT

As federated learning brings a large paradigm to modern artificial intelligence research, efforts are being made to incorporate federated learning into research in various fields. However, researchers who apply federated learning face the problem of choosing a federated learning framework and benchmark tool suitable for their situation and purpose. This study aims to present guidelines for selection of federated learning frameworks and benchmark tools considering the circumstances of researchers who apply federated learning in practice. In particular, there are three main contributions in this study. First, it generalizes the situation of the researcher applying federated learning by combining it with the goal of federated learning and proposes guidelines for selecting a federated learning framework suitable for each situation. Second, it shows the suitability of selection by comparing the characteristics and performance of each federated learning framework to the researcher. Finally, the limitations of the existing federated learning framework and a plan for real-world federated learning operation are proposed.

☞ keyword : Federated Learning, Distributed Learning, FL Framework, Open Source

## 1. 서론

연합학습은 최근 인공지능분야의 연구에서 큰 관심을 가지고 다양한 분야에서 연구되고 있다. 연합학습을 적

용하는 가장 큰 이유는 인공지능 학습을 위한 데이터가 중앙 서버에 수집되지 않고 분산된 각 클라이언트에 존재해도 학습이 가능하기 때문이다. 이러한 연합학습의 특징은 인공지능 학습 과정에서 발생할 수 있는 개인정보 보호와 통신 비용 절감이라는 문제점을 해결하는 중요 포인트가 될 수 있다.

연합학습의 관심과 함께 연합학습을 쉽게 적용 및 평가할 수 있는 오픈 소스 프레임워크와 벤치마크 툴 역시 매우 다양하고 활발하게 개발되고 있다. 하지만 여러 목적과 다양한 분야에서 통일적으로 사용되는 연합학습 솔루션은 존재하지 않는다. 연합학습 적용을 위한 연구자들은 자신의 상황과 목적에 맞는 오픈 소스 프레임워크

<sup>1</sup> Innopia Technologies, Inc, Seongnam-Si, 13217, South Korea

<sup>2</sup> Dept. of Computer Engineering, Gachon University, Seongnam, 13120, Korea

\*Corresponding author (kwangkeelee@gmail.com)

[Received 6 March 2023, Reviewed 16 March 2023(R2 2 May 2023), Accepted 3 June 2023]

☆ 이 논문은 2022년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No. 1711178441, 블록체인 융합 연합학습 기술개발)

와 벤치마크 툴을 선택해야하는 문제에 직면하게 된다.

본 연구에서는 연합학습 적용을 위한 연구자의 상황에 따라 적합한 연합학습 오픈 소스 프레임워크와 벤치마크 툴을 선택할 수 있는 가이드라인 제시하며 기여점은 다음과 같다.

첫번째, 연합학습을 적용하는 연구자의 상황을 연합학습의 목표와 결합하여 일반화하고, 각 상황에 적합한 연합학습 프레임워크의 선택 가이드라인을 제안한다

두번째, 연구자에게 연합학습 프레임워크를 각각의 특징과 성능비교를 통해 선택의 적합성을 보여준다.

세번째, 현존하는 연합학습 프레임워크의 한계와 실세계 연합학습 운영을 생명주기 관리 플랫폼의 구체적인 구성을 제안한다.

본 논문의 구성은 다음과 같다. 2.관련 연구에서는 연합학습의 이론적 배경에 대해 설명하고 현재 연합학습의 연구 동향 논문에 대해 서술한다. 그리고 현존하는 벤치마크 툴과 오픈 소스 프레임워크, 데이터에 대해 설명한다. 3.연합학습 프레임워크 선택 가이드라인에서는 연합학습 적용을 위한 연구자의 상황을 4가지로 분류하고 각 상황에 맞는 벤치마크 툴과 오픈 소스 프레임워크를 제안한다. 4.프레임워크 한계 및 실세계 적용 방안에서는 기존 오픈 소스 프레임워크의 한계점을 서술하고 실세계 적용과 지속적인 연합학습 플랫폼의 운용 방안에 대하여 제시한다. 5.결론에서는 본 연구의 요약과 함께 추가적인 연구의 방향성과 계획에 대해 서술한다.

## 2. 관련연구

### 2.1 연합학습 개요

연합학습은 로컬 데이터 샘플을 보유하는 다수의 분산 에지 장치 또는 서버들이 원시 데이터를 교환/공유하지 않고 기계학습 문제를 해결하기 위해 협력하는 기술로, 각 로컬 노드(클라이언트/디바이스)는 생산한/보유한 원시 데이터를 로컬 모델 학습에만 사용함으로써, 데이터 생산자/제공자의 프라이버시를 보호하고, 데이터 소유/활용의 파편화 문제를 해결할 수 있다.

연합학습은 모든 로컬 데이터 세트가 하나의 서버에 수집/공유 단계인 전통적인 중앙집중식 기계학습 방식 혹은 로컬 데이터 샘플이 동일하게 분포 (identically distributed) 된다고 가정하는 전통적인 분산 접근방식과는 대비되며, 데이터의 통계적 이질성 문제와 디바이스의 시스템적 이질성 극복이 필수적으로 요구된다.

데이터 통계적 이질성 : 다수의 다양한 사용자/디바이스, 동적 환경 및 시공간으로부터 수집된 데이터는 독립동일분포(iid: independent identically distributed) 조건을 만족하지 못하고 비 균일/불균형의 특성을 지님

디바이스 시스템적 이질성 : 연합학습에 참여/기여하는 디바이스의 성능과 기능 및 네트워크 환경이 다양하고, 디바이스의 추가, 변동이 지속적으로 발생

연합학습은 (그림 1)과 같이 데이터 소유, 관리, 활용의 파편화 문제를 해결하기 위한 사일로-교차 연합학습과 디바이스/서비스 사용자 데이터를 활용하기 위한 디바이스-교차 연합학습으로 특징과 이슈를 구분할 수 있다.

대규모 데이터 활용은 인공지능 기술 발전의 기본 조건이지만 프라이버시 보호, 보안 등의 이유로 데이터의 공유 및 활용이 제한적인 분야가 존재하며, 데이터의 직접적인 공유 없이 분산 환경에서 기계학습을 수행할 수 있는 연합학습 기술을 통해 인공지능 기술 활용의 확장을 기대할 수 있다.

기존에는 중앙 서버에 모든 데이터를 수집 후 학습하는 과정이 일반적이고 프라이버시 침해 위험이 존재하므로, 이를 개선하기 위해 각 디바이스에서 로컬 모델을 학습하고 이를 동기화하는 연합학습 기술 필요성 대두되었으며, 연합학습을 통해 여러 참여자들 및 이해 당사자 간에 데이터를 공유하지 않고도 공통의 강력한 기계학습 모델을 구축할 수 있으므로 데이터 개인정보 보호, 데이터 보안, 데이터 액세스 권한 및 이기종 데이터에 대한 액세스와 같은 중요한 문제를 해결할 수 있어서 연구개발이 가장 활발히 진행되고 있는 인공지능 분야 중 하나로 자리매김하고 있다.

연합학습 기술의 활성화는 다변화/파편화 되어있는 데이터 소유/관리/활용 주체 간의 협력을 활성화하여 산업 내, 산업 간 융합을 촉진하고, 데이터 프라이버시 문제로 인하여 학습에 활용되지 못하고 있는 고부가가치 데이터를 활용함으로써 신규 사업/산업 창출을 기대할 수 있다.

### 2.2 연합학습 이론적 배경

연합학습이란 구글에서 논문[1]으로 처음 제안된 개념으로써 연합학습은 데이터를 중앙화된 서버로 수집하여 학습을 진행하는 기존 인공지능 학습 방식과는 다르게 분산된 클라이언트 로컬 데이터와 모델의 파라미터를 이용하여 서버에서 모델을 생성하는 분산화된 학습 방식이다.

연합학습은 분산된 클라이언트의 학습인 로컬 업데이트와 중앙 서버에서 각 클라이언트의 파라미터를 통합하는 글로벌 업데이트로 두 가지로 구성된다.

연합학습에서 기본 로컬 업데이트는 다음과 같은 방식으로 진행된다.

$$D = \{D^1, D^2, D^3, \dots, D^i\} \quad (1)$$

$$W_{(t+1)}^i = W_t - \alpha \nabla L(D^i, W_t) \quad (2)$$

수식(1)의  $D$ 는 각 클라이언트가 가지고 있는 로컬 데이터 셋을 나타낸다. 수식(2)의  $W$ 는 학습 모델의 파라미터를 뜻하며,  $t$ 와  $i$ 는 각각 라운드와 클라이언트이다.  $\alpha$ 는 학습률,  $L$ 은 손실 함수 표현한다.

연합학습에서 기본 글로벌 모델 업데이트는 다음과 같은 방식으로 진행된다.

$$W_{(t+1)} = \sum_i^k P_i W_{(t+1)}^i \quad (3)$$

수식(3)의  $P_i$ 는 글로벌 업데이트에서 클라이언트의 로컬 모델의 중요도에 대한 표현이다. 대표적인 예시로는 연합학습의 기본이 되는 FedSGD와 FedAvg의 경우  $P_i$ 는  $\frac{k_i}{k}$ 가 될 수 있다.

결국 최종적인 연합학습의 목표는 수식(4)와 같이 목적 함수(Object Function)를 최적화하는 것이다.


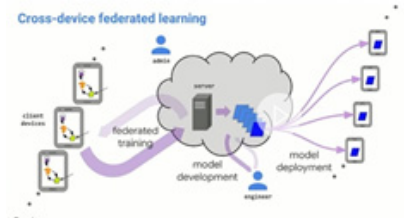
$$\min F(W), \text{ where } F(W) = w E[L(W, D)] + \lambda * g(W) \quad (4)$$

$E[L(W, D)]$ 는 로컬 클라이언트의 모델과 데이터셋의 업데이트 기대 값을 나타내며,  $g(W)$ 는 규제(Regularization)의 강도를 조절하는 하이퍼 파라미터와 규제항(term)을 의미한다. 대표적으로는 FedProx[2]가 있으며 FedSGD와 FedAvg의 경우 규제항이 따로 존재하지 않으며, 다양한 알고리즘에서  $g(W)$ 를 추가하여 성능 향상을 이루고 있다.

### 2.3 연합학습 연구 동향

연합학습에 대한 관심이 높아진 만큼 수 많은 연구들이 등장하고 있다. Bingyan Liu(2020)[3]의 논문에서 연합학습 파이프라인의 핵심은 집계 알고리즘이며 이에 따라 집계 최적화, 이기종 연합학습, 안전한 연합학습, 공정한 연합학습으로 분류하고 이러한 범주 안에서 최신 연합학습 사례 방법과 기능에 대해 제시한다.

연합학습을 실제 산업 및 연구에 적용하기 위해서는 연합학습에서 발생하는 여러 문제들을 고려해야 한다. Chen Zhang(2021)[4]은 데이터 파티셔닝, 프라이버시 매커니즘, 기계 학습 모델, 통신 비용, 통계적 및 구조적 이질성에 대한 5가지 측면을 소개하여 연합학습의 실제 적용 시나리오에 대해 제시하였다. 또한, Ahmed Saidani

	<p>사일로-교차 연합학습 (Cross-Silo)</p> <ul style="list-style-type: none"> <li>- 서로 다른 기관(의료 혹은 금융) 혹은 지리적으로 분산되어 있는 데이터센터들이 각자의 사일로 데이터를 학습 (2 - 100 clients)</li> <li>- 데이터/통계적 이질성, 디바이스/시스템적 이질성 문제 小</li> <li>- 모든 클라이언트가 항상 가용</li> </ul>
	<p>디바이스-교차 연합학습 (Cross-Device)</p> <ul style="list-style-type: none"> <li>- 사용자의 개인 디바이스 (휴대폰, IoT)가 개인 데이터를 학습 (Massive of clients)</li> <li>- 데이터/통계적 이질성, 디바이스/시스템적 이질성 문제 大</li> <li>- 일정 시간에 일부 클라이언트만 가용하고, straggler effect 대응 필요</li> </ul>

(그림 1) 사일로-교차 연합학습과 디바이스-교차 연합학습 비교

(Figure 1) Comparison of Cross-Silo Federated Learning and Cross-Device Federated Learning

(2023)[5]는 현재 다양한 연합학습 프레임워크가 존재하지만, 연합학습을 운영할 수 있는 기능이 부족하고 프로덕션에서 사용할 수 없다는 것을 지적하며 여러 연합학습 프레임워크에 대한 특성 분석 및 기능 비교를 수행하였다.

다양한 연합학습 프레임워크를 쉽게 배포하고 테스트할 수 있는 Xiaoyuan Liu(2022)[6]의 UniFed는 연합학습 연구자가 특정 응용 프로그램에 대한 연합학습 프레임워크를 선택하는데 도움이 되는 정보와 연합학습 툴킷을 개발하여 정량적 평가의 워크플로우를 제공한다.

연합학습 연구 동향 논문에서 다양한 연합학습 프레임워크의 기능 설명과 성능 및 유용성에 대한 비교 연구는 많지만, 연합학습을 사용하는 연구자의 상황을 고려하여 연합학습 프레임워크에 대한 선택적 가이드라인 정보를 제공하지 않는다. 이에 우리는 연합학습을 적용하는 연구자의 상황을 일반화 시키고 각 상황에서 공통적으로 발생 할 수 있는 질문을 통해 상황에 따라 연합학습 프레임워크를 선택할 수 있는 가이드를 제시한다.

## 2.4 연합학습 프레임워크 및 벤치마크

시뮬레이션 환경에서의 연합학습은 데이터와 모델에 따른 연합학습 성능비교에 중점을 둔다. FedScale[7], FedBench[8], Federated scope[9], EasyFL[10], TFF[11], FedTree[12], FLUTE[13] 등 다양한 시뮬레이션 환경의 프레임워크는 성능 비교 실험을 위해 오픈 데이터와 모델을 제공하고, 기존의 연구와 자신의 연합학습 알고리즘 시뮬레이션이 쉽게 가능하도록 도움을 준다.

실세계 적용이 가능한 연합학습 프레임워크는 시뮬레이션 환경의 프레임워크와 다르게 다양한 사용 사례를 다룰 수 있다. FATE[14], FedML[15], Flower[16], Paddle FL[17]와 같은 프레임워크는 실제 클라이언트 환경에 데이터와 모델을 배포할 수 있고 연합학습 라운드를 수행할 수 있다. Flower에서는 연합학습 환경을 모바일에서도 지원하여 실제 안드로이드/iOS에서 연합학습 적용이 가능한 환경을 제공하며, FedML의 경우 자신들의 온라인 연합학습 플랫폼을 제공하며 실제 엣지 디바이스에서 FedML을 이용하여 연합학습이 동작되는 것을 튜토리얼로 쉽게 체험해볼 수 있다. 그리고 FATE의 경우 사용하는 방식이 복잡하지만, 그 만큼 매우 방대한 사용 사례를 제공한다.

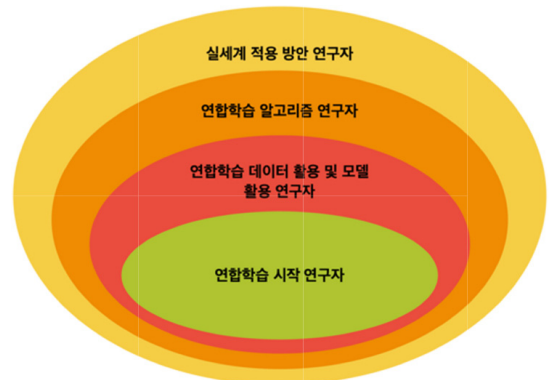
## 3. 연합학습 프레임워크 선택 가이드라인

### 3.1 연합학습 연구자 상황

연합학습을 적용하는 연구자의 상황은 매우 다양하다. 여러 분야의 연구자들이 각자의 목표를 가지고 연합학습을 적용하기 때문이다. 그 결과 다양한 연합학습 프레임워크가 존재하는 생태계에서 자신의 상황과 목적에 맞는 프레임워크를 선택해야 하는 문제에 도달하게 된다.

우리는 연합학습 프레임워크 선택 가이드를 제안하기 이전에, 연합학습을 적용하는 연구자들의 상황을 일반화 시켜야 했다. 일반화를 위해서는 다양한 분야와 목표를 가지고 연합학습을 적용하는 연구자들에게서 공통점을 찾아야 했고 우리는 그 공통적인 기준을 연합학습의 이론적 배경인 목적함수의 최적화에서 찾을 수 있었다. 연합학습 목적함수의 최적화를 기준으로 선정 한 이유는 다양한 분야의 연구자들이 연합학습을 적용할 때 각자의 상황도 다르고 목표도 다르지만, 결국 연합학습 목적함수 최적화의 각 항(term)에서 각자의 방식을 적용한다는 공통점을 가지기 때문이다.

본 연구에서는 연합학습 목적함수의 최적화를 기준으로 연합학습 연구자의 상황을 (그림 2)과 같이 4가지로 분류하고 3.2를 통해 자세히 설명한다. 그리고 각 상황에서 발생할 수 있는 문제들과 이에 해답을 줄 수 있는 적합한 프레임워크를 선택할 수 있도록 도움을 준다.



(그림 2) 연합학습 연구자 상황 분류  
(Figure 2) Federated Learning Researcher Classification

### 3.2 연합학습 연구자 상황 별 선택 가이드

#### Situation 1. 연합학습을 시작하는 연구자

연합학습을 시작하는 연구자는 연합학습을 시도해보는 단계의 상황으로 연합학습 목적함수 최적화에 큰 의미를 가지지 않는다. 오로지 연합학습이라는 개념을 실제로 구현하고 실현되는 것을 목표로 한다.

연합학습을 시작하는 초기 연구자의 경우 다음과 같은 공통적인 질문 사항이 발생할 수 있다.

#### Q1-1. 연합학습을 실행하는 간단한 방법은 무엇인가?

연합학습을 시작하는 연구자들은 연합학습의 개념을 실제 동작하는 간단한 환경을 통해 실현하고 동작 과정을 확인할 수 있어야 한다. 연합학습을 간단하게 실행하는 방법으로는 셀 형식의 연합학습 단계 실행과 적은 라인의 코드를 가지고 모듈화시켜 연합학습을 실행하는 방법이 있으며, 도커를 이용하여 연합학습에 필요한 환경을 설정없이 간단하게 실행하는 방법 역시 존재한다. (표 1)에서 보여지듯이, Flower는 적은 수의 코드로 연합학습 튜토리얼을 실행할 수 있고 도커 환경도 제공하여 쉽게 연합학습을 경험해 볼 수 있다.

TFF와 FedBench는 연합학습 튜토리얼을 실행하기 위해 주피터 노트북 형식의 파이썬 파일을 실행시켜야 된다. 하지만, 도커 환경을 지원하지 않아 연합학습을 시작하는 연구자 입장에서 연합학습 프레임워크에 맞는 환경 설정 필요하다. FATE는 도커 환경을 지원하여 간단하게 연합학습을 실행할 수 있지만, 실제계에서 연합학습을 운영할 수 있는 산업 등급의 다양한 기능을 제공하여 연합학습을 처음 접해보는 연구자가 사용하기에 어려움이 있다.

(표 1) 연합학습 초기 연구자를 위한 연합학습 프레임워크 비교 분석

(Table 1) Comparative Analysis of Federated Learning Frameworks

	Easy start	Tutorial Doc	Model Library
Flower	○	○	○
FATE	✓	○	○
FedML	○	○	✓
PaddleFL	○	○	X
TFF	✓	○	✓
EasyFL	○	✓	○
FedBench	✓	X	✓

\*✓는 제한적 지원을 의미

#### Q1-2. 접근성 높은 연합학습 환경을 지원하는가?

연합학습은 분산화 된 인공지능 학습 방식이기 때문에 기존의 인공지능 라이브러리인 Tensorflow와 Pytorch의 사용 유무가 중요하다. 연합학습 프레임워크들은 튜토리얼 문서 정보를 통해 각자 제공하는 연합학습을 위한 기능 구성과 기존 라이브러리와와의 접근성에 대해 설명한다. (표1)의 결과처럼 Flower은 연합학습 튜토리얼에 대한 문서화가 잘되어 있어 연합학습 프레임워크에 대한 구조를 이해하기 쉽고, 대표적인 인공지능 모델 라이브러리를 제공하여 접근성이 높은 연합학습 환경을 지원한다.

PaddleFL은 연합학습 튜토리얼에서 제시하는 연합학습 구조에 대한 설명은 제공하지만, 로컬 모델 구성에 있어 PaddleFL의 자체 모델 구조를 따라야 하는 어려움이 있다. easyFL은 이 프레임워크의 전체 연합학습 구조에 대한 설명을 제공하지만, 튜토리얼에 대한 자세한 설명은 하지않는다. FedBench는 튜토리얼에 대한 설명을 제공하지 않고, Pytorch기반의 모델 라이브러리만을 제공하여 Tensorflow기반으로 모델을 생성하는 연구자에게는 어려움이 존재한다.

Q1과 Q2를 종합적으로 고려하였을 때, 연합학습을 시작하는 연구자에게 적합한 연합학습 프레임워크는 Flower이다. Flower는 손쉽게 연합학습 튜토리얼을 실행할 수 있을 뿐만 아니라, 튜토리얼 실행 과정에 따른 설명을 제공하고 사용자 친화적 특성에 맞게 다양한 인공지능 모델 라이브러리를 지원한다.

#### Situation 2. 연합학습 데이터 및 모델 연구자

연합학습 데이터 및 모델 연구자의 경우 연합학습의 개념을 이해하고 시도했으며, 이제 여러 학습 데이터와 다양한 분야의 모델(자연어 처리, 이미지 처리, 강화학습 등)의 적용을 목표로 하는 상황이다. 이 상황의 연구자들은 연합학습을 적용하여 로컬 업데이트와 글로벌 업데이트의 기댓값을 항상 집중하며 연합학습 목적함수 최적화의 수식(4) 항에 포함된다.

$$\min_W F(W), \text{ where } F(W) = E[L(W; D)] + \lambda * g(W) \quad (5)$$

데이터 및 모델 연구자들은 데이터를 제공해 주거나 모델을 기본적으로 제공해주는 프레임워크 및 벤치마크 툴을 선호할 것이다. 추가적으로 연합학습의 데이터의

이질성(Data Heterogeneity)을 위한 파티션 기능을 제공하거나 자신이 실험하고 싶은 커스텀 데이터와 모델을 쉽게 프레임워크에 적용시킬 수 있다면 좋은 선택지가 될 수 있을 것이다.

연합학습 데이터 및 모델 연구자 경우 다음과 같은 공통적인 질문사항이 발생할 수 있다.

Q2-1. 데이터가 존재하지 않을 경우 데이터셋을 제공할 것인가?

데이터를 보유하지 않은 연구자들은 다양한 프레임워크에서 오픈 데이터셋을 제공받을 수 있다. 그리고 제공되는 오픈 데이터는 대부분 데이터 이질성 조건을 만족하는 데이터이다. (표 2)에서 볼 수 있듯이 모든 프레임워크에서 기본적으로 데이터셋을 제공하며, 가장 많이 사용되고 링크되는 Leaf[18]의 6가지 데이터는 다른 프레임워크의 예제에서도 활발하게 사용되는 데이터이다. FedScale은 자신들의 강점으로 다양한 데이터셋 제공을 주장하고, 대중적이고 기본적인 데이터를 포함한 20가지 데이터셋을 제공한다. Fate의 경우 특수한 케이스로 다른 프레임워크와 중복되지 않는 데이터셋을 제공한다.

(표 2) 연합학습 프레임워크 데이터 샘플 및 파티션, 커스텀 형식 비교 표

(Table 2) FL Framework Data Sample and Partition, Custom Format Comparison Table

Framework	Sample Dataset	Data Partition	Custom format
Flower	2	O	informal
FedML	17	O	formal
FedScale	20	X	formal
FedBench	5	X	informal
Fate	13	X	formal
Leaf	6	O	informal
PaddleFL	10	X	formal
EasyFL	3	O	formal

Q2-2. 커스텀 데이터 및 모델을 쉽게 적용할 수 있는가?

연합학습 연구에서 중요한 부분은 자신의 데이터와 모델을 적용하는 것이다. 하지만 각 프레임워크마다 커스텀 모델과 데이터를 적용하는 방식에서 큰 차이가 존재한다. (표 2)에서처럼 Flower의 경우 가장 비형식적인 방식으로 자신의 데이터와 모델을 쉽게 적용할 수 있는 친화적인 프레임워크이다. FedBench 역시 특별한 적용 방식 없이 바로 연합학습을 적용할 수 있지만, 벤치마

킹을 위한 프레임워크로 시뮬레이션 환경이라는 한계점이 존재한다. 그 외의 다른 프레임워크는 각자의 프레임워크에서 제공하는 형식에 맞춰 데이터와 모델을 수정해야 한다.

특히 커스텀 데이터의 사용에서는 연합학습의 데이터 이질성(Non-iid) 조건으로 인해 데이터 파티션 기능이 존재하는데, 다양한 프레임워크에서 파티션 기능을 제공하지만 아직까지 명확히 모듈화 된 기능을 제공하는 프레임워크는 존재하지 않는다.

연합학습 데이터 및 모델 연구자의 경우 다양한 데이터를 경험해보고 시뮬레이션을 목표로 한다면 FedScale을 이용하면 좋은 선택이 될 것이다. 그리고 자신의 커스텀 데이터와 모델을 적용하고 실제 환경까지도 생각한다면 Flower와 같은 비형식적이고 자유로운 프레임워크를 사용하면 효율적으로 자신의 연구를 진행할 수 있다. 하지만 연합학습 데이터 파티션 기능은 아직까지 프레임워크에서 명확한 모듈로 제공하지 않기 때문에 적용하는데 난이도가 높기 때문에 실험에 주의해야 한다.

### Situation 3. 연합학습 알고리즘 연구자

연합학습 알고리즘 연구자의 경우 연구 방식과 목표가 최적화에서 수식(5)의 연합학습의 규제항(Regularization term) 연구에 포함된다. 클라이언트의 업데이트 방식을 개선하거나 서버의 모델 통합 과정에서 가중치를 추가하여 성능을 향상시키는 것이 목표이다.

$$\min F(W), \text{ where } F(W) = E[L(W, D)] + \lambda * g(W) \tag{6}$$

클라이언트 개선 방식으로는 로컬 모델 업데이트의 손실 함수의 계산 방식(FedProx, FedNova[19] 등), 개인화(Personalization) 방법 등 클라이언트 부분에 규제 항을 추가하여 성능 향상을 시도하며, 서버 모델 통합 과정에서는 모델 파라미터 통합을 위한 규제 알고리즘(FedAvg, FedSGD), 클라이언트 선택(Client Selection), 클라이언트 학습 기여도 평가(Contribution Evaluation) 등 다양한 규제 방식들이 연구되고 있다.

알고리즘 연구자의 경우 자신의 연합학습 규제 항의 논리를 쉽게 프레임워크에 구현할 수 있다면 좋은 선택지라고 생각할 것이다. 그렇기 때문에 개인화를 위한 로컬 업데이트와 라운드 마다 학습 클라이언트의 선택 기

능, 각 로컬 모델에서 수집할 수 있는 파라미터의 자유도 등 프레임워크의 연합학습 기능들을 쉽게 수정할 수 있게 한다면 높은 선호도를 보일 것이다.

연합학습 알고리즘 연구자의 경우 다음과 같은 공통적인 질문사항이 발생할 수 있다.

**Q3-1. 프레임워크 별 성능 차이가 존재하는가?**

연합학습 알고리즘을 연구하기 위해서는 실험을 위한 기본적인 셋팅이 필수조건이다. 같은 방식, 같은 모델, 같은 데이터로 실험을 진행하는데, 각 프레임워크에서 결과가 비슷하다면 이상이 없지만, 이상이 발생하는 프레임워크가 존재한다면 그 원인을 파악하고 사용 유무를 결정해야 한다.

(그림 3)은 MNIST 데이터셋을 Non-iid 파티션 후 각 클라이언트에게 배포하여 성능을 비교해본 그래프이다. (그림 3)의 그래프를 보면 클라이언트가 10명 참여한 경우 Flower와 FedML의 경우 82%의 정확도를 고점으로 수렴하지만, FLUTE의 경우 80%로 수렴하여 약 2%의 성능 차이를 보였다. 그리고 수렴하는데 필요한 라운드 역시 프레임워크별로 차이가 존재하는데, 이 역시도 Flower와 FedML의 경우 유사한 라운드에서 모델이 최고 정확도에 도달하지만, FLUTE의 경우에는 더 긴 시간의 라운드가 필요했다.

시뮬레이션 환경에서 프레임워크 별로 연합학습의 소요 시간에도 차이가 존재한다. (표4)에서는 참여 클라이언트 수에 따라서 성능 차이가 존재하는 것을 볼 수 있다. Flower의 경우 10명과 100명의 참여에서 가장 빠른 성능을 확인할 수 있었고, FLUTE의 경우 20명과 50명의 참여에서 빠른 성능을 보였다. 그리고 FedML의 경우 모든 경우에서 가장 오랜 시간 소비된 결과를 보였다.

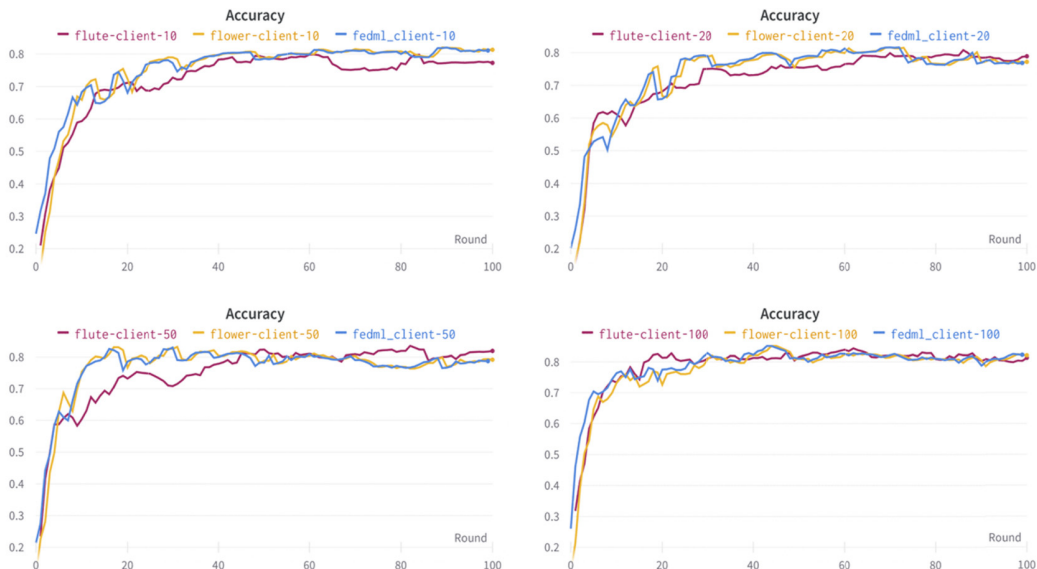
(표 3) 클라이언트 수에 따른 연합학습 프레임워크 수행 시간 비교 표

(Table 3) Comparison Table of Federated Learning Framework Runtime According to the Number of Client

Number of Client	Flower	FedML	FLUTE
Client 10	<b>94.1 s</b>	186.5 s	112.9 s
Client 20	129.7 s	192.7 s	<b>118.4 s</b>
Client 50	152.3 s	204.2 s	<b>143.1 s</b>
Client 100	<b>155.7 s</b>	231.9 s	174.6 s

**Q3-2. 집계 최적화, 개인화, 클라이언트 선택 등 규제 조건을 추가하기 용이한가?**

연합학습 알고리즘에서 규제조건을 추가하기 위한 방



(그림 3) 시뮬레이션 환경에서 연합학습 프레임워크 성능 비교

(Figure 3) Federated Learning Framework Performance Comparison in Simulation Environment

법에는 다양한 방식이 존재한다. 그리고 규제조건은 클라이언트 파트, 서버 파트 그리고 학습 도중 등 다양한 부분에서 적용될 수 있다.

FedProx, FedNova와 같은 집계 최적화 방식은 클라이언트의 로컬 업데이트에서 손실함수를 통해서 진행된다. (표 4)를 보면 대부분의 프레임워크에서 예제를 제공하고 연구자는 예제를 통해 자신의 집계 최적화 방식을 실험할 수 있다.

개인화 알고리즘 역시 클라이언트의 로컬 모델 업데이트에서 로컬 데이터에 최적화된 모델을 찾는 연구이다. Federated Scope은 이런 개인화 연합학습 연구에 최적화된 프레임워크로 pFL-Bench[20] 논문을 통해서 Meta learning, Transfer Learning, Multi-task Learning이 프레임워크에서 적용될 수 있다고 설명한다.

클라이언트 선택(Client Selection)의 경우 서버의 글로벌 모델의 통합 과정에서 연합학습에 참여하는 클라이언트의 로컬 데이터, 디바이스 성능, 통신 상태 등의 조건을 통해 가중치를 부여해 통합 모델을 학습하는 연구이다. 연합학습의 Cross-Silo 구조에서는 클라이언트 선택의 영향이 적을 수 있지만, 수 많은 클라이언트가 연합학습에 참여하는 Cross-Device 구조에서는 매우 유의미한 영향을 볼 수 있다. (표 3)을 보면 모든 프레임워크에는 연합학습 클라이언트를 선택하는 과정이 존재한다. 하지만 선택 과정을 연구자가 접근 및 수정할 수 있도록 구조화한 프레임워크는 Flower, FedBench, Leaf, EasyFL 등으로 분류된다. 그리고 특히 FedScale의 경우 클라이언트 선택 알고리즘 Oort[21]의 저자가 참여하고 있기 때문에 다른 프레임워크에 비해 클라이언트 선택 부분에서 강점을 가지고 있다.

연합학습 프레임워크의 경우 자신들의 효율성 높은 연합학습 구조를 위해 개발되었지만, 또 다른 이유로는 각자 자신의 알고리즘 연구를 자신의 손에 맞는 프레임워크를 이용하기 위하여 개발되기도 하였다. 이 상황의 연구자는 다양한 프레임워크들의 장단점을 비교하여 선택을 하는 경우 연구의 첫 걸음에서는 좋은 방향일 수 있지만, 프레임워크의 숙련도가 높아진 상황이라면 구조를 이해하고 자신의 손에 맞는 프레임워크를 사용하는 것이 더 좋은 결과를 볼 수 있는 방법일 수 있다. 또한, 프레임워크 성능 실험을 통해서 프레임워크 선택을 고려한다면 본 논문에서는 시뮬레이션 환경에서 정확도와 소요 시간을 비교한 실험이기 때문에 연합학습의 중요 변수들(통신비용, 로컬 디바이스 성능)에 대해서는 고려되지 않았다는 한계점을 인지해야 한다.

(표 4) 연합학습 프레임워크의 집계 최적화 예시 및 클라이언트 선택 모듈화 비교 표

(Table 4) Aggregation Optimization Example of Federated Learning Framework and Client Selection Modularization Comparison Table

Frame work	Example of Aggregation Optimization	Client Selection Module
Flower	O	O
FedML	O	X
FedScale	O	O
FedBench	O	O
Fate	X	X
Leaf	X	O
FLUTE	X	O
PaddleFL	X	X
EasyFL	X	O

Situation 4. 연합학습 실세계 적용 연구자

연합학습의 실세계 적용 연구자의 경우 앞에서 제시한 상황들을 모두 경험했을 가능성이 높다. 이 상황의 연구자는 자신이 앞의 과정에서 겪은 경험과 기반 지식을 통해서 실제 연합학습 목적함수의 최적화를 달성하고 이 모델을 실 세계에 적용하고 테스트하는 부분이 수식(7)에 포함된다.

$$\min_W F(W), \text{ where } F(W) = E[L(W;D)] + \lambda * g(W) \tag{7}$$

실세계 적용 단계의 연구자들은 프레임워크의 선택의 초점을 자신이 적용한 연합학습 모델이 실세계에서 어떻게 학습되고 각 클라이언트는 어떻게 참여되고 있는지, 그리고 학습된 데이터의 분포나 클라이언트의 디바이스의 종류, 시스템 성능 등 실세계 테스트와 관련된 기능을 가진 프레임워크를 선호할 것이다.

연합학습 실 세계 적용 연구자 경우 다음과 같은 공통적인 질문사항이 발생할 수 있다.

Q4-1.연합학습의 클라이언트를 관리, 모니터링 할 수 있는 기능이 존재하는가?

연합학습에 참여하는 클라이언트를 관리하기 위해서는 클라이언트에서 데이터 처리를 할 수 있고, 클라이언



트를 구분할 수 있는 ID를 제공하는지 확인할 수 있어야 하며, 클라이언트 관리 정보를 가지고 클라이언트의 성능과 학습 과정을 모니터링하는 것 역시 중요하다. 또한, 연합학습은 라운드 개념으로 반복적인 프로세스가 진행되기 때문에 클라이언트가 라운드에 참여하기 위해 자신이 온라인 상태인지, 학습하고 있는 상태인지에 대한 정보도 공유되어야 한다.

(표 5)의 비교표와 같이 모든 프레임워크는 클라이언트의 입장에서 로컬 데이터를 사용할 수 있는 파이프라인을 제공한다. EasyFL 외에 모든 프레임워크에서 클라이언트 ID를 제공하며 클라이언트를 관리한다. 클라이언트의 성능과 학습 과정을 모니터링할 수 있는 도구를 제공하는 프레임워크는 FedML, FATE, TFF이다. 특히, FedML과 FATE는 자체 개발한 웹 화면을 통해 클라이언트의 온라인/학습 상태를 확인할 수 있다.

**Q4-2.Edge 디바이스에서 연합학습이 가능한가?**

모바일과 같은 Edge 디바이스의 성능 향상으로 로컬 환경에서 인공지능 학습을 수행할 수 있다. 이에 연합학습 프레임워크에서 모바일과 같은 Edge 디바이스를 이용한 실 세계 적용 예제를 제공한다. Flower, FedML 프레임워크에서는 모바일 환경에 대한 연합학습 예제를 제공한다. PaddleFL은 모바일 환경에 대한 예시가 공식문서에 나와있지만, 안드로이드/iOS이 아닌 시뮬레이터 환경을 제공한다. TFF 역시 모바일 환경에 적용할 수 있는 API만 제공할 뿐, 모바일 연합학습 적용 예제가 없으므로, 모바일 환경에서 연합학습을 수행하기에 어려움이 존재한다. Q1과 Q2를 종합적으로 판단하였을 때, 연합학습 실제 적용 연구자들에게 적합한 프레임워크는 FedML이다.

FedML은 시뮬레이션 환경 뿐만 아니라 클라이언트

관리, 모니터링, 모바일 환경과 같은 실제 연합학습 운영에 필요한 기능도 지원한다.

연합학습은 반복 프로세스이므로 연합학습 전체 과정을 관리할 수 있는 수명주기관리 기능이 필요하다. 하지만, 현재 FedML을 포함한 다양한 연합학습 프레임워크들은 부분적으로 연합학습에 필요한 기능을 지원할 뿐 연합학습 환경을 효율적으로 운영할 수 있는 환경은 제공하지 않는다. 이에 대한 세부적인 내용은 4에서 설명한다.

**4. 연합학습 프레임워크 한계 및 실 세계 적용 방안**

**4.1 기존 프레임워크의 한계점**

다양한 프레임워크가 존재하는 연합학습 생태계에는 각각 다른 목표를 가지고 개발되었기 때문에 프레임워크마다 강점이 다르다. 하지만 이런 강점들 역시 완벽하지 않고 한계점이 존재한다.

TFF는 친숙한 환경인 Tensorflow와 함께 연합학습 기능을 제공하는 강점이 존재하지만, 커스텀 데이터와 모델을 적용한 다양한 연합학습 상황에 적용하기 위한 시스템을 직접 구축해야 하는 한계점이 존재한다. FATE는 연합학습 환경을 배포하기 위해 쿠버네티스와 도커를 사용하지만[4], 복잡한 시스템 설계로 실제 연구하고 싶은 연합학습 상황에 활용하기 어렵다. FedScale과 EasyFL은 연합학습을 적은 코드 수로도 쉽게 시뮬레이션 할 수 있지만, 캡슐화된 코드에 접근하여 커스텀 데이터와 모델을 적용하기 어렵고 지속적인 통합/배포/학습에 대한 기능을 지원 및 관리하지 않는다. Flower는 다수의 클라이언트와 서버를 간편하게 연결하여 연합학습 라운드를 수

(표 5) 연합학습 프레임워크의 클라이언트 관리, 모니터링, 상태 확인, 모바일 환경 지원 비교 표  
(Table 5) Comparison Table of Client Management, Monitoring, Status Check, and Mobile Example Support of Federated Learning Framework

Framework	Data Preprocessing	Client ID	Monitoring	Status Check	Mobile Example
Flower	○	○	X	X	○
FedML	○	○	○	○	○
PaddleFL	○	○	X	X	X
FATE	○	○	○	○	X
EasyFL	○	X	X	X	X
TFF	○	○	○	X	X

행할 수 있다. 하지만, 연합학습의 커뮤니케이션 기능을 지원할 뿐, 연합학습을 운영 관리할 수 있는 생태계 생명 주기 관리 기능을 지원하지 않는다.

현재 연합학습의 프레임워크 중 실세계에서 연합학습을 적용할 수 있고 사용자 친화적인 UI/UX 환경을 지원하는 프레임워크는 FedML이다. FedML은 Cross-silo, Cross-device 시나리오에 맞는 클라이언트 등록과 코드 배포 환경을 지원한다. 등록된 클라이언트에 내에서 선택하여 연합학습을 실행할 수 있다. 연합학습에 대한 결과는 로그와 모니터링 대쉬보드를 제공하여 연합학습 성능을 비교 분석할 수 있다. 하지만, FedML 역시 실제 산업 및 연구에서 발생하는 각 클라이언트의 실시간 데이터를 처리하고 로컬 모델에 적용하여 자동적으로 연합학습을 운영 관리할 수 없다는 한계점이 존재한다.

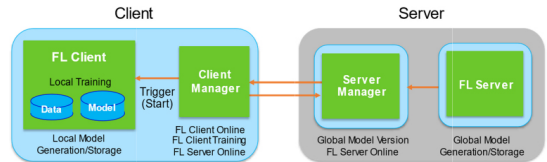
#### 4.2 실 세계 적용 및 운용 방안

실세계에 연합학습 환경을 적용하기 위해서는 연합학습을 지속적으로 운영 관리할 수 있는 기능이 필요하다. 지속적으로 연합학습을 운영 관리하기 위해서 여러 클라이언트의 상태를 지속적으로 관리하고 최신 버전의 로컬 모델 및 글로벌 모델을 계속해서 통합/배포/학습할 수 있어야 된다. 최신 데이터와 모델을 보유하고 있는 클라이언트의 상태를 관리하고 주기적으로 연합학습 라운드를 생성하여 로컬 학습이 진행될 수 있는 환경을 제공해야 한다. 이러한 연합학습의 전체 과정을 기록하고 모니터링하여 연합학습의 수명주기를 관리해야 한다.

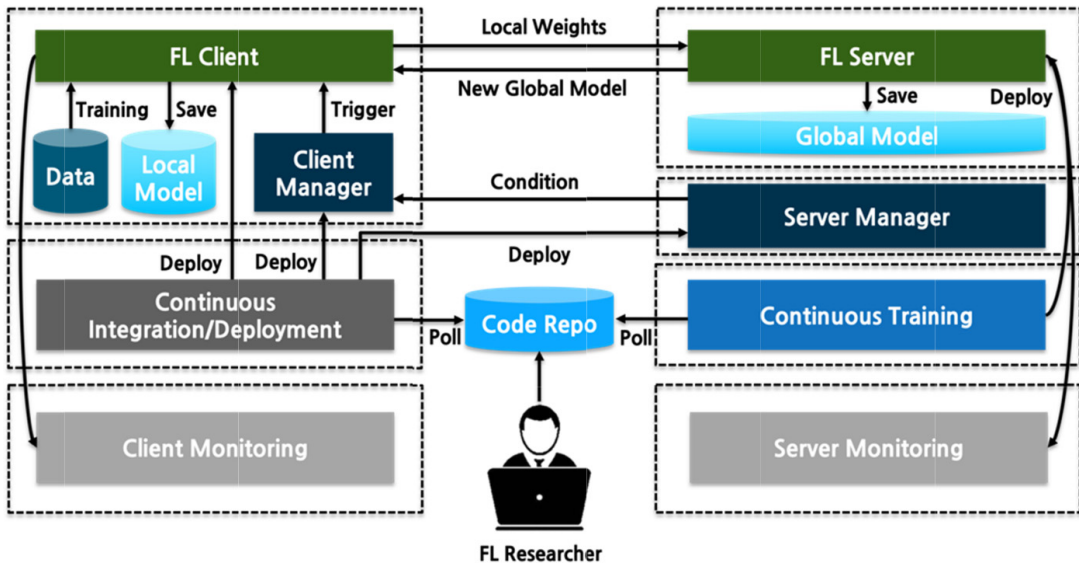
연합학습 운영을 위한 수명주기관리 플랫폼의 구조는 (그림 4)와 같이 다양한 구성 요소를 확장하고 재정의 할 수 있도록 시스템을 구축해야 한다. 구체적인 각 컴포넌트의 기능과 프로세스는 아래와 같다.

**FL Client & FL Server** : 연합학습 환경에서 손쉽게 최신 데이터와 모델을 적용가능하도록 하며, 연합학습의 라운드를 관리하고 진행하는 역할을 담당하는 컴포넌트

**Client/ServerManager** : FL Client와 FL Server를 관리하는 컴포넌트로서, 클라이언트 상태를 지속적으로 파악하고, 연합학습의 라운드 진행 시 필요한 Trigger를 각 클라이언트에 전달하는 역할을 하는 모듈.



(그림 5) Manager Communication Process (Figure 5) Manager Communication Process

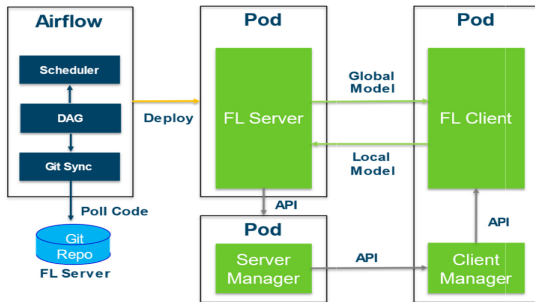


(그림 4) 실세계에서 연합학습 운영을 위한 연합학습 수명주기관리 플랫폼 구조

(Figure 4) Federated Learning Lifecycle Management Platform Structure for FL Operation in the Real World

**Continuous Integration/Deployment (CI/CD)** : 실 세계에서 발생할 수 있는 연합학습 클라이언트 환경을 구성하여 연합학습의 두가지의 시뮬레이션 환경 제공한다. Client pod에 CPU/GPU/Memory와 같은 리소스를 설정하여 시스템 이질성 시뮬레이션 환경을 구현할 수 있다. 또한, 클라이언트를 복제하여 참여 클라이언트의 수를 간편하게 설정할 수 있고, 클라이언트마다 ID가 부여되어 각 클라이언트가 서로 다른 데이터를 보유할 수 있다. 즉, 데이터 이질성 시뮬레이션 환경을 지원한다. 이렇게, 다양한 연합학습 테스트에 실 세계에서 발생할 수 있는 연합학습 환경 구성이 가능하다.

**Continuous Federated Learning** : 다양한 연합학습 테스트를 생성하고 이를 관리하며, 최신 버전의 FL Server Pod를 배포하는 역할을 담당한다. 온라인 상태의 Client pod는 그림 6과 같이 FL Server pod의 start 정보를 받고 연합학습 라운드에 참여하게 되어 continuous training이 가능한 Continuous Federated Learning(CFL)을 수행할 수 있다.



(그림 6) Continuous Federated Learning Process  
(Figure 6) Continuous Federated Learning Process

**Client/Server Monitoring** : 연합학습의 현재 참여에 대한 정보와 진행사항, 최종적인 결과에 대해서 각 서버와 클라이언트는 효율적으로 판단할 수 있어야한다. 모니터링 기능은 그 부분을 채워주는 역할을 한다. 기존 프레임워크에서 모니터링 부분이 존재해도, 서버에 한정된 정보만 제공하지만 각 클라이언트도 마찬가지로 자신의 로컬 모델과 로컬 데이터가 어떻게 학습되고 참여되는지 알아야 할 필요성이 존재한다.

## 5. 결 론

최근 연합학습 연구 동향은 각 연합학습 프레임워크

와 벤치마크의 기능과 성능 차이를 비교하고 있다. 하지만, 이러한 연구 동향들은 각 연합학습 프레임워크의 통신 비용과 정확도와 같은 성능에 초점을 두기만 할 뿐, 실질적으로 연합학습 연구자에게 선택을 위한 프레임워크 정보를 제공하지 않는다.

본 연구에서는 연합학습 연구자 상황을 연합학습 목적함수 최적화 수식과 연관지어서 4가지 상황으로 일반화시키고, 각 상황에 발생할 수 있는 질문들과 연관지어 비교 실험을 통해 연합학습 프레임워크를 선택할 수 있는 가이드라인을 제안하였다. 활발히 개발 및 관리되는 Flower와 FedML, Fate의 경우에는 대부분의 상황에서 가장 많이 언급된 프레임워크이다. 하지만 위 프레임워크 외에도 다양한 프레임워크들이 각자의 강점을 강조하면서 다양한 상황에서 충분히 활용 될 수 있는 부분들이 존재한다.

하지만 아직 통합적으로 사용되는 연합학습 프레임워크가 존재하지 않는 이유로써 각 프레임워크에는 아직까지 한계점이 존재하고 특히 실세계 적용에는 통합/배포/학습을 위한 지속적인 수명주기 관리와 좀더 자세하고 세분화 된 모니터링 그리고 다양한 엣지 디바이스에서의 활용성 등 해결해야 할 부분들이 존재한다.

본 연구에서는 연합학습 프레임워크 비교와 실용적인 가이드를 중심으로 진행되었지만, 추후 이어지는 연구에서는 각 프레임워크의 한계점을 보완하고 다양한 분야에서 통합적으로 사용되는 프레임워크 설계 및 개발하는 것을 목표로 나아갈 것이다.

## References

- [1] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloé Kiddon, Jakub Konečný, Stefano Mazzocchi, Brendan McMahan, Timon Van Overveldt, David Petrou, Daniel Ramage, Jason Roselander, "Towards Federated Learning at Scale: System Design", Proceedings of Machine Learning and Systems, Vol. 1, pp. 374-388, 2019.  
<https://arxiv.org/abs/1902.01046>
- [2] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In Third Conference on Machine Learning and Systems (MLSys), 2020. <https://arxiv.org/abs/1812.06127>

- [ 3 ] Bingyan Liu, Nuoyan Lv, Yuanchun Guo, Yawen Li, “Recent Advances on Federated Learning: A Systematic Survey”, arXiv, 2023.  
<https://doi.org/10.48550/arXiv.2301.01299>
- [ 4 ] Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, Yuan Gao, “A survey on federated learning”, Knowledge-Based Systems, Vol. 216, 2021.  
<https://doi.org/10.1016/j.knosys.2021.106775>
- [ 5 ] Ahmed Saidanim, “A Systematic Comparison of Federated Machine Learning Libraries”, DEPARTMENT OF INFORMATICS TECHNISCHE UNIVERSITÄT MÜNCHEN, 2023.
- [ 6 ] Xiaoyuan Liu, Tianneng Shi, Chulin Xie, Qimbin Li, Kangping Hu, Haoyu Kim, Xiaojun Xu, Bo Li, Dawn Song, “UniFed: A Benchmark for Federated Learning Frameworks”, arXiv, 2022.  
<https://doi.org/10.48550/arXiv.2207.10308>
- [ 7 ] Fan Lai, Yinwei Dai, Sanjay S. Singapuram, Jiachen Liu, Xiangfeng Zhu, Harsha V. Madhyastha, Mosharaf Chowdhury, “FedScale: Benchmarking Model and System Performance of Federated Learning at Scale”, International Conference on Machine Learning (ICML), 2022. <https://doi.org/10.48550/arXiv.2105.11367>
- [ 8 ] <https://github.com/OnizukaLab/FedBench>
- [ 9 ] Yuexiang Xie, Zhen Wang, Dawei Gao, Daoyuan Chen, Liuyi Yao, Weirui Kuang, Yaliang Li, Bolin Ding, Jingren Zhou, “FederatedScope: A Flexible Federated Learning Platform for Heterogeneity”, arXiv, 2022. <https://doi.org/10.48550/arXiv.2204.05011>
- [10] W. Zhuang, X. Gan, Y. Wen and S. Zhang, “EasyFL: A Low-Code Federated Learning Platform for Dummies,” in IEEE Internet of Things Journal, vol. 9, no. 15, pp. 13740-13754, 1 Aug.1, 2022.  
<https://doi.org/10.1109/JIOT.2022.3143842>
- [11] <https://github.com/tensorflow/federated>
- [12] Mohammad Al-Quraan, Ahsan Khan, Anthony Centeno, Ahmed Zoha, Muhammad Ali Imran, Lina Mohjazi, “FedTrees: A Novel Computation-Communication Efficient Federated Learning Framework Investigated in Smart Grids”, arXiv, 2022.  
<https://doi.org/10.48550/arXiv.2210.00060>
- [13] Mirian Del Carmen Hipolito Garcia, Andre Manoel, Daniel Diaz Madrigal, Robert Sim, Dimitrios Dimitriadis, “FLUTE: A Scalable, Extensible Framework for High-Performance Federated Learning Simulations”, NeurIPS, 2022.  
<https://doi.org/10.48550/arXiv.2203.13789>
- [14] <https://github.com/FederatedAI/FATE>
- [15] Chaoyang He, Songze Li, Jinhyun So, Xiao Zeng, Mi Zhang, Hongyi Wang, Xiaoyang Wang, Praneeth Vepakomma, Abhishek Singh, Hang Qiu, Xinghua Zhu, Jianzong Wang, Li Shen, Peilin Zhao, Yan Kang, Yang Liu, Ramesh Raskar, Qiang Yang, Murali Annamavaram, Salman Avestimehr, “FedML: A Research Library and Benchmark for Federated Machine Learning”, arXiv, 2020. <https://doi.org/10.48550/arXiv.2007.13518>
- [16] Daniel J. Beutel, Taner Topal, Akhil Mathur, Xinchu Qiu, Javier Fernandez-Marques, et al. “FLOWER: A FRIENDLY FEDERATED LEARNING FRAMEWORK”. 2022. <https://hal.science/hal-03601230>
- [17] <https://github.com/PaddlePaddle/PaddleFL>
- [18] Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečný, H. Brendan McMahan, Virginia Smith, Ameet Talwalkar, “LEAF: A Benchmark for Federated Settings”, arXiv, 2019.  
<https://doi.org/10.48550/arXiv.1812.01097>
- [19] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H. Vincent Poor, “Tackling the objective inconsistency problem in heterogeneous federated optimization”, Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS’20), No. 638, pp. 7611 - 7623, 2020.  
<https://doi.org/10.48550/arXiv.2007.07481>
- [20] Daoyuan Chen, Dawei Gao, Weirui Kuang, Yaliang Li, Bolin Ding, “pFL-Bench: A Comprehensive Benchmark for Personalized Federated Learning”, arXiv, 2022.  
<https://doi.org/10.48550/arXiv.2206.03655>
- [21] Fan Lai, Xiangfeng Zhu, Harsha V. Madhyastha, Mosharaf Chowdhury, “Oort: Efficient Federated Learning via Guided Participant Selection”, arXiv, 2020. <https://doi.org/10.48550/arXiv.2010.06081>

● 저 자 소 개 ●



**김진수(JinSoo Kim)**  
2018년 가천대학교 IT융합공학과 석사  
2020년 (주)티비스툼 연구원  
2023년 (주)Innopia Tech 연구원  
관심분야 : 인공지능, 딥러닝  
Email : wlstn25092303@gmail.com



**양세모(SeMo Yang)**  
2022년 가천대학교 컴퓨터공학과(공학사)  
2022년~현재 가천대학교 컴퓨터공학과 석사과정  
관심분야 : 인공지능, 머신러닝  
E-mail : tpah20@gachon.ac.kr



**이강윤(KangYoon Lee)**  
1986년 연세대학교 전자공학과(공학사)  
1996년 연세대학교 전자계산학과(공학석사)  
2010년 숭실대학교 IT정책경영학과(공학박사)  
1991년~2016년 한국 IBM 연구소(연구소장)  
2016년~현재 가천대학교 컴퓨터공학과 교수  
관심분야 : 인공지능, IoT, 빅데이터, 클라우드, 커그너티브컴퓨팅  
E-mail : keylee@gachon.ac.kr



**이광기(Kwangkee Lee)**  
1994년 삼성종합기술원, 삼성전자  
2016년 산업부/KEIT 산업융합 PD  
2019년 (주)티비스툼 이사  
2022년 (주)Innopia Tech 이사  
E-mail : kwangkeele@gmail.com