J. Internet Comput. Serv.
ISSN 1598-0170 (Print) / ISSN 2287-1136 (Online)
http://www.jics.or.kr

개인화 디지털 트윈을 위한 연합학습 기반 클라이언트 훈련 가속 방식☆

Federated learning-based client training acceleration method for personalized digital twins

정 영 환[1] 최 원 기[1] 계 효 선[1] 김 지 형[1] 송 민 환[1] 이 상 신[1*]
YoungHwan Jeong Won-gi Choi Hyoseon Kye JeeHyeong Kim Min-hwan Song Sang-shin Lee

요 약

디지털 트윈은 현실세계의 물리적 객체를 디지털 세계의 가상객체로 모사하고 시뮬레이션을 통해 미래에 발생 가능한 현상을 예측함으로써, 현실세계의 문제를 해결 또는 최적화하기 위해 고안된 M&S(Modeling and Simulation) 기술이다. 디지털 트윈은 지금까지 도시, 산업 시설 등 대규모 환경에서 특정 목적을 달성하기 위해 수집된 다양한 데이터 기반으로 정교하게 설계되고 활용되어 왔다. 이러한 디지털 트윈 기술을 실생활에 적용하고 사용자 맞춤형 서비스 기술로 확장하기 위해서는 개인정보 보호, 시뮬레이션의 개인화 등 실질적이지만 민감한 문제를 해결해야 한다. 이러한 문제를 해결하기 위해 본 논문에서는 개인화 디지털 트윈을 위한 연합학습 기반의 클라이언트 훈련 가속 방식(FACTS)을 제안한다. 기본적인 접근 방식은 클러스터 기반의 적응형 연합학습 훈련 절차를 활용해 개인정보를 보호하면서 동시에 사용자와 유사한 훈련 모델을 선택하고 훈련을 가속하는 것이다. 다양한 통계적으로 이질적인 조건의 실험 결과 FACTS는 기존의 FL 방식에 비해 훈련 속도 및 자원 효율성 측면에서 우수한 것으로 나타난다.

☞ 주제어 : 디지털 트윈, 연합학습, 벡터데이터베이스, 훈련 최적화, 개인정보보호, 유사성 검색

ABSTRACT

Digital twin is an M&S (Modeling and Simulation) technology designed to solve or optimize problems in the real world by replicating physical objects in the real world as virtual objects in the digital world and predicting phenomena that may occur in the future through simulation. Digital twins have been elaborately designed and utilized based on data collected to achieve specific purposes in large-scale environments such as cities and industrial facilities. In order to apply this digital twin technology to real life and expand it into user-customized service technology, practical but sensitive issues such as personal information protection and personalization of simulations must be resolved. To solve this problem, this paper proposes a federated learning-based accelerated client training method (FACTS) for personalized digital twins. The basic approach is to use a cluster-driven federated learning training procedure to protect personal information while simultaneously selecting a training model similar to the user and training it adaptively. As a result of experiments under various statistically heterogeneous conditions, FACTS was found to be superior to the existing FL method in terms of training speed and resource efficiency.

☞ keyword : Digital twin, Federated Learning, Vector database, training optimization, privacy, similarity search

1. 서 론

디지털 트윈은 현실 세계를 3D로 모사한 가상 세계를 통해 현실 세계의 특정 현상을 모니터링하고 분석 및 예측할 수 있어 산업, 공공, 의료 등 다양한 분야에서 널리 활용된다. 최근 정보 기술의 발달로 디지털 트윈에서 현실 세계 사물이 실시간으로 연결되고 데이터 분석을 통한 시뮬레이션 방법이 점차 다양해지면서 디지털 트윈의 개념이 사물인터넷, 빅데이터, 그리고 인공지능과 같이 여러 분야와 연계가능한 방향으로 발전하고 있다. 특히,

1 Autonomous Intelligence System Research Center, Korea Electronics Technology Institute (KETI), Saenari-ro, Bundang-gu, Seongnam-si, 13509, Korea.
* Corresponding author (sslee@keti.re.kr)
[Received 21 March 2024, Reviewed 27 March 2024(R2 7 June 2024), Accepted 18 July 2024]
☆ 이 논문은 2024년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. RS-2022-II220545, 지능형 디지털 트윈 연합 객체 구성 및 데이터 프로세싱 기술 개발)
☆ 본 논문은 2023년도 한국인터넷정보학회 추계학술대회 우수논문 추천에 따라 확장 및 수정된 논문임.

사물인터넷(Internet of things, IoT)과 AI 기술의 발전은 디지털 트윈의 활용 가능성이 다양한 장치 단말과의 실시간 연결을 넘어, 각 단말에서 생산, 가공, 저장되는 데이터에 대한 응용으로 확장되게 한다. 이에 디지털 트윈에서 AI를 통한 현상 예측을 수행하고자 하는 시도가 각종 연구분야에서 활발하게 수행되고 있다.[1][2][3] 하지만 이러한 데이터 분석 및 AI 기술의 발전에도 불구하고 디지털 트윈을 개인화 서비스와 융합하고자 하는 시도는 많지 않았다. 이는 사용자 IoT 단말이 개인적인 선호를 반영하여 설치되고 데이터 수집을 수행하고 있기 때문이며, 그 규모가 방대하고, 수집된 데이터는 민감한 개인정보를 포함하고 있어 IoT 장치를 떠나 수집 및 활용되는 경우 송수신 데이터의 높은 처리 및 저장 비용은 물론, 프라이버시 문제까지 일으킬 수 있어 주의가 요구되기 때문이다. 따라서 디지털 트윈에 개인화 서비스를 융합하고 사용자 맞춤형 시나리오를 연계하기 위해서는 비용 효율적이면서 동시에 개인정보 문제에서 자유로운 접근 방식이 요구된다.

최근 DCML(Distributed and Collaborative Machine Learning)로 정의되는 분산 모델 학습 체계에서는 이러한 비용 및 민감정보처리에 대한 문제를 완화하기 위해 다양한 방법론들이 연구되었다. 먼저, 완전 동형 암호화(FHE) 기반 분산 모델 훈련 방식은 복호화 없이 암호화된 기울기에 대해 동형 연산을 지원하여 사용자 개인 정보를 유출하지 않고 모델을 병렬적으로 훈련한다.[4] 둘째로 연합학습(Federated learning)은 데이터를 local에 유지한 채 종단 간 훈련 모델의 교환을 통해 훈련에 참여하는 클라이언트의 데이터 특징을 학습한다[5]. 이때 모든 훈련 작업이 로컬에서 수행되므로 서버는 추가적인 비용 없이 훈련 규모를 수평적으로 확장할 수 있다. 셋째 분할학습(Split learning)은 훈련 모델의 상위 일부를 분할하여 장치 단말에 위치하고 서버-클라이언트간 협력적 훈련을 통해 모델을 훈련한다.[6] 이러한 접근 방식은 클라이언트에 로컬 데이터 유출을 방지하고 낮은 훈련 부하를 유지하면서 대규모 신경망을 훈련할 수 있게 한다. 넷째는 FL과 SL의 절충적인 방안으로 다수의 클라이언트에 훈련 모델의 일부를 배치하고 병렬적으로 서버와 협력하여 훈련한다.[7] 하지만 이러한 접근 방식들은 다양한 참여 클라이언트의 데이터 다양성을 수용하고 중앙서버에서 학습하고자 하는 훈련 모델의 일반화 성능을 극대화하는 방향으로 초점이 맞추어져 있어 클라이언트의 훈련 자원 다양성과 같은 실질적인 가정하에 사용자에게 최적화된 모델 훈련이라는 측면에서는 적합하지 않다. 따라서 디지털 트윈에 융합가능한 사용자 맞춤형 모델 훈련에 DCML 체계를 활용하기 위해서는 기존 DCML 알고리즘에 대한 이상적인 가정의 완화가 필요하다.

기존 DCML 훈련 체계에는 두가지 이상적인 가정이 있다. 첫번째 훈련 과정에서 참여하는 모든 클라이언트는 훈련이 끝날 때까지 안정적인 연결을 유지하고 문제 없이 훈련 결과를 반환하는 것으로 가정하는 것이다. 둘째, 모든 클라이언트의 데이터는 독립 동일 분포(IID)라는 것이다. 그러나 디지털 트윈 환경을 구성하는 각 클라이언트들은 일반적으로 이기종 시스템이다. 즉, 모든 클라이언트는 서로 다른 컴퓨팅 리소스와 네트워크 환경을 가지고 있어 DCML 훈련 체계가 목적으로 하는 획일적인 훈련 방식에서 일부 클라이언트의 동작 속도가 느리거나 응답하지 않을 수 있다. 또 디지털 트윈을 구성하는 각 클라이언트들은 서로 다른 환경에서 편향된 데이터를 수집하므로 각 클라이언트의 로컬 데이터 간의 분포와 크기에 통계적 이질성을 갖는 non-IID 로컬 데이터가 존재한다. 이러한 통계적 이질성은 클라이언트 측면에서 모델이 서로 다른 방향으로 훈련되며 수렴을 저해할 수 있다. 따라서 디지털 트윈에 실질적으로 활용가능한 개인화 모델 훈련을 위해서는 이와 같은 실질적인 환경을 반영하는 효과적인 클라이언트 훈련 방식이 필요하다.

본 논문에서는 디지털 트윈에 연계가능한 사용자 맞춤형 모델 훈련을 위한 연합학습 기반 클라이언트 훈련 가속 방식(Federated learning based Accelerated client training scheme, FACTS)을 제안한다. FACTS는 기존 연합학습의 클라이언트 연결성 문제와 non-IID 문제를 완화하고 기존의 일반화되어 훈련되는 전역 모델 훈련 방식을 사용자 맞춤형 모델 훈련으로 전환하기 위해서 서버와 클라이언트 양 측에서 다음과 같이 동작한다. 먼저, 훈련에 참여하는 클라이언트는 서버로부터 학습기한과 훈련의 상한을 할당받고 해당 기한동안 적응적으로 훈련하여 최대 학습 에포크 수를 클라이언트가 적극적으로 결정하도록 유도한다. 한편, 서버 측에서는 클라이언트의 이러한 적응적 훈련 방식으로 발생하는 모델 편향을 완화하고, 공정성을 확보하기 위해 Aging-term 기반의 클라이언트 선택 방식을 채택한다. 이를 통해 상대적으로 훈련이 덜 된 클라이언트에 더 많은 훈련기회가 주어져 공정성을 확보한다. 또 서버는 클라이언트의 모델 개인화를 가속하기 위해 유사한 훈련 모델을 검색하기위해 클러스터 중심의 유사도 기반 모델 선정 체계를 구축한다. 이러한 접근 방식은 클라이언트 데이터의 non-IIID에 의한 통계적 이질성을 완화하고 동시에 사용자 맞춤형으로 최적화된 모델 훈련

을 가속할 수 있도록 한다. FACTS의 성능을 평가하기 위해 보다 실질적인 조건(각 클라이언트의 이질적 데이터 크기 및 자원 환경)에서 실험한 결과, FACTS는 기존 FL 방식에 비해 모델 성능의 사용자화, 공정성 및 훈련 속도를 향상시키는 것을 보여준다.

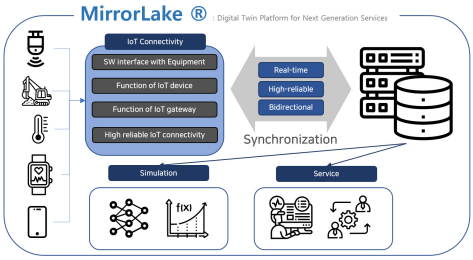
FACTS의 기여는 크게 3가지이다. 1) 적응형 훈련 방식을 사용하여 클라이언트의 로컬 훈련을 가속하고 집계 절차에 따른 클라이언트 낙오자를 완화한다. 2) Aging-term 기반의 클라이언트 선택 방식으로 전역 모델이 보다 다양한 데이터를 학습하여 모델의 일반화 성능을 향상시킨다. 3) 클러스터 기반의 유사 모델 선택 방식을 통해 클라이언트의 모델 과적합을 완화하고 개인화 모델 훈련을 가속한다.

본 논문의 나머지 부분은 다음과 같은 내용으로 구성된다. 2장에서는 디지털 트윈과 FL 그리고 Vector 데이터베이스와 관련된 기존 연구를 요약한다. 3장에서는 FACTS의 시스템 모델을 설명한다. 4장에서는 FACTS에 대한 알고리즘을 설명하고 공식화한다. 5장에서는 벤치마크 데이터에 대한 실험 결과를 도출 및 설명하고 6장에서는 결론을 내린다.

2. 관련 연구

2.1 디지털 트윈

디지털 트윈은 다양한 분야에 적용되고 있지만 그 개념은 유사한 방식으로 정의되고 있다. Grieves[8]는 디지털 트윈이 제품 생산 공정 절차에서 물리적 제품, 해당 제품의 가상 표현, 물리적 제품에서 가상 제품으로 데이터를 공급하는 양방향 데이터 연결이라는 세 가지 구성요소로 구성되어 있다고 설명하여 디지털 트윈의 정의를 연결성 측면에서 확장한다. He. Y[9]는 디지털 트윈을 물리적 자산, 프로세스 및 시스템의 전주기를 종합적으로 관제하는 디지털 복제본으로 정의했다. Bruynseels. K[10]는 디지털 트윈이 디지털 모델이 결합된 공학 패러다임으로서 개별 물리적 객체와 해당 객체의 상태를 동적으로 반영하는 가상 시스템이라 정의하였다. 이와 유사하게 Singh[11]은 디지털 트윈은 특정 지점에서 물리적 트윈의 정확한 상태를 나타내는 실제 주제 또는 객체(부품, 기계, 프로세스, 인간 등)의 역동적이고 자체 진화하는 디지털/가상 모델 또는 시뮬레이션으로 정의한다. 이처럼 디지털 트윈은 실제 현실 세계의 문제를 예측, 분석 및 최적화하기 위해 단순히 물리적 환경을 3D 모델링 기반으로 시각적 복제를 수행하는 것에 그치지 않고, 물리적 환경의 물체 및 현상을 디지털화하고 정제하여 현실 세계와 가상 공간을 실시간으로 연결하는 것을 목적으로 한다.



(그림 1) 디지털 트윈 동기화 플랫폼, MirrorLake
(Figure 1) Digital twin synchronization platform, MirrorLake

이러한 디지털 트윈 기술이 주목받으면서 동시에 사물인터넷 기술을 활용하여 현실 세계의 실시간 데이터를 디지털 트윈에 동기화하고 이를 기반으로 프로세스의 예측·최적화 과정에 효과적으로 연계하기 위한 다양한 연구가 진행되고 있다. 한국전자기술연구원은 그림 1과 같이 IoT 장치로부터 수집된 물리적 객체의 실제 데이터를 정제하고 시공간적으로 동기화하여 다양한 시뮬레이션 모델 및 서비스와 상호 작용하는 디지털 트윈 동기화 플랫폼 MirrorLake를 개발하였다. MirrorLake는 디지털 트윈의 단순 데이터 수집 및 모니터링 기능 뿐 아니라 물리적 객체의 양방향 제어가 가능하다는 점에서 주목받으며, 공공·산업계의 디지털 트윈 구축을 지원하고 있다. Han et al.[12]은 여러 IoT 단말이 그룹을 형성하고 그룹 간 상호 협업하여 실제 환경의 개체를 감지하는 디지털 트윈을 고안하고, 동적 계층화 프레임워크를 제안하여 사용자가 요구하는 강도로 동기화를 지원할 수 있게 한다. Jia et al.[13]은 디지털 트윈을 위한 지능형 클록 동기화 기법을 제안해 빠르게 변화하는 산업용 IoT 환경에서 분산 동기화와 관련된 리소스 소비정도를 완화한다. 해당 기법은 가상 클록 모델링을 수행하여 각 클록을 특성화하고 동적 운영 환경에서 클록의 동작을 예측하도록 구성되어 동기화를 위한 과도한 타임스탬프 교환을 방지한다. Yishuo et al.[14]은 건설 현장에서의 계획, 관리, 수행 등 작업의 전주기를 최적화 할 수 있는 디지털 트윈 시스템을 제안하여 실작업 현장 내의 다양한 자원이 시공간적으로 가장 적합한 작업에 할당되도록 한다. Elayan

et al.[15]은 디지털 트윈을 실시간 변환된 데이터를 통해 현재 상태를 반영하는 물리적 자산의 가상 복제본으로 정의하면서 디지털 트윈을 활용한 지능형 상황인식 의료 시스템을 제안한다. 해당 시스템은 심장 질환을 진단하고 심장 문제를 감지하기 위해 기계 학습 기반 시뮬레이션을 사용하여 심전도(ECG) 심장 박동 분류 모델을 구축하고 디지털 트윈과 융합한다.

한편, 현실 세계의 복잡성을 반영함과 동시에 보다 효과적으로 변화를 분석 및 대응하기 위해 디지털 트윈에 AI 기술을 융합하고자 하는 다양한 연구가 수행되고 있다. 디지털 트윈에 대한 AI 기술의 도입은 서로 다른 환경에서 수집되는 다양한 데이터의 상호 연관 관계를 분석하는데 효과적으로 작용한다. 왜냐하면, AI 모델 기반 분석은 기존의 고전적인 모델링 방식보다 다양한 잠재변수의 상호작용을 효과적으로 파악하고 예측 변수에 영향을 미치는 핵심 요인들이 복합적으로 연관된 프로세스에 적용되었을 때 성능이나 안정성에 미치는 영향을 포착하는데 활용될 수 있기 때문이다. Kharchenko et al.[16]은 산업용 IoT를 활용한 제조 분야의 디지털 트윈(DT)을 지능화하기 위해 산업에 큰 영향을 미치는 장비, 인력 및 프로세스 이 세 가지 주요 요소에 AI 모델 기반 데이터 분석 및 예측을 적용한다. 분석한 결과는 사용자의 의사결정을 지원하기 위해 가시화된다. Zhou et al.[17]는 디지털 트윈에서 원심 임펠러의 물리적 특징을 분석하고 주요 변수를 도출하여, 이를 기반으로 기계 가공성과 공기 역학적 성능의 최적값을 갖는 설계를 수행하기 위해 DDPG (Deep Deterministic Policy Gradient) 기반 기하학적 최적화를 통해 설계 및 제조 속도를 향상시켰다. Rao et al.[18]은 간 질환 위험 진단에서 의사결정을 지원하는 디지털 트윈과 Explainable AI를 결합해 LIME (Local Interpretable Model-Agnostic Explanations) 기반으로 간 세포 내 효소 AST와 ALT가 간 질환에 미치는 영향에 대한 요약을 시각적으로 제공한다.

2.2 연합학습 (Federated Learning, FL)

연합학습(FL)은 Brendan et al.[5]이 제안한 분산 학습 방법으로 연결된 클라이언트가 보유한 데이터를 로컬에 유지하여 개인정보를 보호하면서 종단 간 모델 교환을 통해 클라이언트 자원을 활용하여 서버 측 전역 모델을 훈련하도록 설계되었다. 이러한 장점으로 인해 FL은 의료, 교통, 통신 기술 등 다양한 분야에 적용할 수 있는 잠재력을 가지고 있다.[19][20][21] 그러나 기존의 단일 시스템에서의 중앙 집중식 모델 훈련 방식과 달리 분산 모델 학습으로서의 FL은 서버-클라이언트 간 모델 교환 및 훈련 구조를 띄고 있으므로 1)시스템 이질성, 2) 통계적 이질성과 같은 실질적인 문제를 해결하는 형태로 연구되어 왔다.

먼저 시스템 이질성과 관련한 연구는 FL의 훈련에 참여하는 클라이언트 가용자원의 다양성에 따라 훈련 절차를 효과적으로 최적화하는 방향으로 연구되었다. Reisizadehet al. [22]은 시스템 이질성을 가정하여 참여 클라이언트를 적응적으로 선택하는 낙오자 탄력적 FL을 제안했다. 제안 방식은 참여 클라이언트 간의 계산 속도를 고려하여 통신 환경에 따라 시스템 런타임을 확장하고 클라이언트의 통계적 특성을 통합하도록 구성된다. Tao et al. [23]은 시스템 이질성으로 인한 클라이언트 낙오 문제를 해결하기 위해 n-Cayley 트리를 통해 작업자와 선택 장치 간의 낙오자 비율을 제어하는 방법론을 제안했다. Li et al. [24]은 시스템 이질성을 고려하여 이종 클라이언트의 갖는 로컬 모델의 부분 모델 업데이트 결과를 서버에서 일괄적으로 집계하고 Proximal-term을 통해 부분 모델 업데이트를 전역 모델 gradient에 통합하는 FedProx를 제안했다.

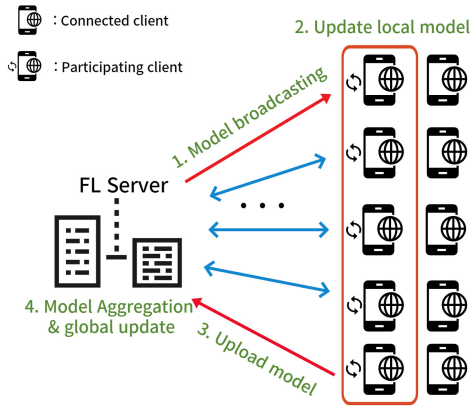
한편, 통계적 이질성과 관련된 연구는 FL의 훈련 참여 클라이언트의 로컬 데이터를 non-IID 데이터로 확장하는데 중점을 두었다. Yang et al. [25]는 경사 하강법을 기반으로 FL의 전역 모델 수렴 경계를 이론적으로 분석하고, Non-IID 데이터 분포에 따른 클라이언트 수렴 경계의 다양성을 통합하는 새로운 수렴 경계를 제안한다. Sattler et al. [26]은 기존의 Gradient Sparsity 압축 기법을 Sparse Ternary Compression(STC)을 통해 확장하여 통신 효율성을 높이고 제한된 대역폭을 갖는 학습 환경에서 전역 모델 훈련 최적화를 달성한다. Karimireddyet al. [27]은 non-IID한 클라이언트 데이터 분포 특성으로 인해 전역 모델의 수렴 속도가 느려지는 것을 분석하고 동시에 참여 클라이언트의 로컬 모델이 서로 다른 방향으로 수렴하는 클라이언트 드리프트를 완화하기 위해 훈련 참가자 간의 데이터 유사성을 활용하여 전역 훈련 횟수를 제어하기 위해 stochastic controlled averaging for on-device FL(SCAFFOLD)를 제안했다. Lai et al. [28]은 non-IID 데이터 분포를 갖는 클라이언트 훈련 환경에서 효과적인 클라이언트 선택을 위해 Oort를 제안하여 모델 수렴 가속화를 시도한다. 이 접근 방식은 클라이언트의 통계적 유사성을 분석하고 클라이언트 데이터 통계 메트릭과 훈련 효율성 사이의 trade-off를 조율하도록 구성되었다.

2.3 벡터 데이터베이스 (Vector DB)

최근 이미지, 음성, 비디오, 텍스트 등을 포함한 다양한 형식의 비정형 데이터가 여러 분야에서 생성 및 저장되고 있다. 이에 기존의 관계형데이터베이스의 구조화되지 않은 형태의 데이터를 저장 및 처리하는 것에 대한 한계가 지적되고, 동시에 각종 분야에 대한 인공지능의 활용성 증가로 인해 대용량의 비정형 데이터를 임베딩하고 벡터화하여 효율적으로 관리, 저장 및 검색하는 특수 목적의 데이터베이스에 대한 필요성이 크게 확산되고 있다. 벡터 데이터베이스는 기존 관계형데이터베이스에서 특성화할 수 없는 고차원의 비정형 데이터를 벡터로 저장하기 위해 설계되었다.[29][30][31] 벡터는 일반적으로 텍스트, 이미지, 오디오, 비디오 등과 같은 원시 데이터에 임베딩으로 정의되는 일종의 변환을 적용하여 생성된다. 이때 임베딩은 머신러닝 모델, 워드 임베딩, 특징 추출 알고리즘 등 다양한 방법을 기반으로 변환된다. 이러한 벡터 데이터베이스는 기존 DBMS의 일치 기반 검색 방식에 비해 유사성 검색이라는 측면에서 장점을 가져, 자연어 처리, 컴퓨터 비전, 추천 시스템 등과 같이 가장 관련성이 있는 데이터를 찾는 시스템에 적용된다. 유사한 특징을 갖는 벡터를 검색하기 위해 벡터 데이터 베이스는 색인 알고리즘과 유사도 알고리즘을 활용한다. 색인 알고리즘은 데이터를 특정한 차원 공간 내 영역에 할당하고 빠르게 접근할 수 있도록 구성하여 대규모 벡터 데이터에서도 효율적으로 검색을 수행할 수 있도록 한다. 대표적인 색인 방법은 해싱, 양자화, 그래프 기반이 있다.[32][33][34] 유사도 검색은 검색을 위한 질의에서 목적으로 하는 데이터를 임베딩하여 벡터로 변환하고 색인되어 저장되어 있는 벡터들과 비교하여 가장 근접한 벡터를 찾아 유사도 검색 기법에 따른 정렬을 수행하는 형태로 진행된다. 일반적으로 사용되는 유사도 검색 방법은 유클리디안 거리(Euclidean Distance), 코사인 유사도(Cosine Similarity) 등이 있다. 대표적인 벡터 데이터베이스로는 Milvus[35], Chroma[36], Pinecone[37], Weaviate[38] 등이 있으며 최근에는 Key-value 데이터베이스인 Redis[39]와 같은 NoSQL 데이터베이스 에도 벡터 형태의 데이터를 처리하는 추가 기능들이 활발하게 개발되고 있다.

3. System model

그림 2는 FL의 훈련 시스템 구조를 보여준다. FL 훈련 시스템은 하나의 서버와 다수 개의 클라이언트가 연결된



(그림 2) 연합학습 훈련 절차
(Figure 2) Federated learning training procedure

서버-클라이언트 구조를 채택한다. 여기서 중앙서버는 지역화된 클라이언트 측 훈련 데이터를 바탕으로 일반화된 전역 모델을 훈련하는 것을 목적으로 하며 이를 위해서 서버는 다수의 참여 클라이언트의 협력적 훈련을 조율하여 전역 모델을 훈련한다. 구체적으로 FL 훈련 시스템은 그림 2와 같이 크게 4개의 프로세스로 세분화 된다. 본 논문에서는 이러한 4개의 훈련 프로세스를 하나의 global iteration으로 정의하고 중앙서버에 연결된 여러 참여 클라이언트에게 분산 모델 훈련을 요청한다. t번째 global iteration이 시작될 때 첫 번째 훈련 절차에서 중앙서버는 전체 네트워크에 연결된 클라이언트 집합 G_t 중 랜덤하게 일정 비율 r_t 만큼의 참여 클라이언트를 선택한다. 이 때 참여 클라이언트 집합 g_t는 다음과 같이 표현된다.

$$g_t = \{c_1, c_2, c_3, \ldots c_n\}, n = [\mid G_t \mid \times r_t] \quad (1)$$

여기서 c_i은 훈련에 참여하는 i번째 클라이언트이다. 참여 클라이언트 c_i는 장치에 독립된 로컬 데이터 D_i를 가지고 있다. 중앙서버는 전역 모델 W를 훈련하기 위해 서 다음과 같은 전역 손실함수$F(W)$의 최소화를 목적으로 한다.

$$F(W) = \frac{1}{\sum_{x=1}^{\mid G_t \mid} \mid D_x \mid} \sum_{i=1}^{\mid G_t \mid} \sum_{j=1}^{\mid D_i \mid} f(W, d_{i,j}) \quad (2)$$

여기서 $d_{i,j}$는 클라이언트 c_i의 j번째 데이터 포인트이다. 이를 위해 중앙서버는 g_t를 구성하는 참여 클라이언트에

게 전역모델 W_t의 복사본 w_t와 로컬 업데이트 횟수 K를 전송한다. 두 번째 절차에서 참여 클라이언트 c_i는 중앙 서버가 전송한 전역모델의 목사본 w_t를 로컬 모델 w_t^i로 할당한다. 그리고 참여 클라이언트 c_i는 지역화된 로컬 데이터 D_i를 통해 w_t^i를 최대 K회 훈련한다. 이를 위한 로컬 손실함수 $F_i(w_t^i)$는 다음과 같이 표현된다.

$$F_i(w_t^i) = F_i(w_t^i, D_i) = \frac{1}{|D_i|}\sum_{j=1}^{|D_i|} f(w_t^i, d_{i,j}) \quad (3)$$

로컬 업데이트를 통해 클라이언트 c_i는 수식 3을 점진적으로 줄이는 것을 목표로 한다. 해당 FL 시스템에서 클라이언트 c_i가 확률적 경사 하강법(Stochastic gradient descent, SGD)를 통해 로컬 모델을 최적화한다고 가정하면 k번째 로컬 업데이트에서 로컬 모델은 다음과 같이 업데이트 된다.

$$w_{t,k}^i \leftarrow w_{t,k}^i - \eta \nabla F_i(w_{t,k}^i, \beta) \quad (4)$$

여기서 η는 학습률이고 β는 클라이언트의 미니배치 데이터이다. 수식 4의 반복으로 클라이언트는 단일 로컬 업데이트를 1회 달성한다. 이와 같은 단일 로컬 업데이트의 K번의 반복을 통해 c_i는 다음과 같은 업데이트 된 로컬 모델 $W_t^i \leftarrow w_{t,K}^i$를 얻을 수 있다. 이후 세 번째 절차에서 참여 클라이언트 $c_i \in g_t$는 로컬 모델 W_t^i를 서버로 업로드하고 네 번째 프로세스에서 중앙서버는 집계된 로컬 모델을 집계한다. 이때 서로 다른 이질적인 참여클라이언트 훈련 방식을 고려하여 집계 시간의 상한을 결정할 수있다. 집계된 업데이트된 로컬 모델을 기반으로 중앙 서버는 다음과 같이 전역 모델을 업데이트한다.

$$W_{t+1} = \sum_{i=1}^{|g_t|} \frac{|D_i|}{D_t} W_t^i, D_t = \sum_{j=1}^{|g_t|} |D_j| \quad (5)$$

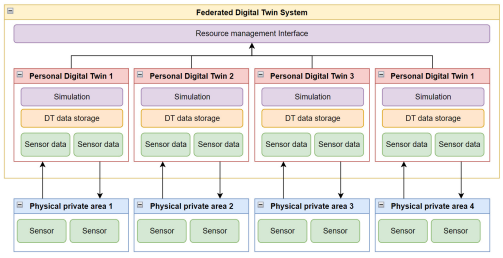
이때 각 참여 클라이언트의 로컬 데이터 규모의 다양성을 고려해 중앙서버는 각 클라이언트의 집계 결과에 데이터 크기를 반영한 수식 5와 같은 가중 평균을 수행할 수 있다. 이러한 절차의 반복을 통해 FL은 점진적으로 최적의 모델 W에 근사하게 된다. 하지만 이와 같이 고정적인 시나리오의 FL 훈련 절차에서는 크게 두 가지 문제가 발생할 수 있는데, 첫 번째는 이기종 참여 클라이언트의 자원 다양성을 고려하지 않아, 가장 느린 클라이언트에 의해 global iteration이 지연되거나 일부 참여 클라이언트가 로컬 모델 업데이트에도 불구하고 집계되지 않는 낙오자 문제의 발생할 수 있다. 두 번째는 FL 시스템의 전역 모델이 다양한 클라이언트의 로컬 데이터를 기반으로 훈련되므로 non-IID한 로컬 데이터 특성에 의해 전역 모델 발산에 대한 노이즈의 강건성이 부족해, 장치 단말에 모델을 배포했을 때 사용자 활용 패턴에 따른 수집된 로컬 데이터의 상이한 수렴 방향성을 고려한 높은 비용의 추가적인 모델 훈련 작업이 요구된다는 점이다.

4. FACTS

본 절에서는 디지털 트윈에 연계 가능한 사용자 맞춤형 모델 훈련을 위한 Federated learning-based Acclerated Client Training Scheme(FACTS)를 제안한다. FACTS의 주요 목적은 다양한 클라이언트(개인화 디지털 트윈)의 모델 훈련의 시간적인 다양성을 고려하면서 동시에 독립적으로 훈련되는 개인화 모델의 일반화 성능을 극대화하는 것이다. 이에 본 절에서는 제안하는 FACTS가 적용되는 연합 개인화 디지털 트윈 시스템을 설명하고, FACTS를 구성하는 일련의 알고리즘과 각 구성을 공식화한다.

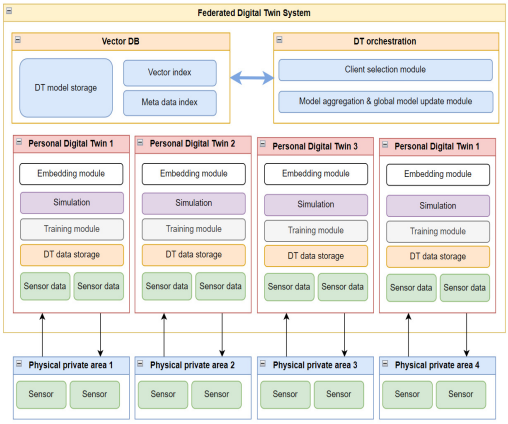
4.1 FACTS overview

개인화 디지털 트윈 시스템(PDT)은 그림 3과 같이 연합 디지털 트윈 시스템(FDT)의 하위 시스템을 이루고 있다. PDT는 다양한 사용자의 물리적으로 개인적인 영역에 위치하는 IoT 단말에서 생성된 데이터를 수집하고 저장, 분석을 수행하므로 각 PDT는 서로 독립적으로 구성된다. FDT는 이러한 PDT가 운용되는 시스템의 리소스를 모니터링하고 관리한다. 이러한 FDT 시스템에서 PDT가 개인화 시뮬레이션을 위한 AI 모델 훈련을 수행하는 경우 DT의 데이터 저장소에 수집된 데이터에 의존하게 되는데



(그림 3) 연합 및 개인화 디지털 트윈 시스템 구조
(Figure 3) Federation and personalized digital twin system structure

대부분의 사용자 수집 데이터는 사용자의 활용 패턴에 따라 데이터 분포가 편향되어 non-IID한 특성을 띄고있고, 그 규모가 상이해 AI 모델이 과적합 또는 과소적합될 우려가 있다. 이러한 문제를 해결하면서 동시에 사용자의 민감 데이터 유출을 방지하기 위해 FACTS에서는 FL 시스템의 훈련 방식을 도입한다.



(그림 4) FACTS에 의한 개인화 모델 훈련
(Figure 4) Personalization model training via FACTS

그림 4는 FACTS에 의한 PDT의 개인화 모델 훈련 방법을 보여준다. FACTS는 크게 FDT 측(서버)에서의 전역 모델 훈련 과정과 PDT 측(클라이언트)에서의 로컬 모델 훈련 과정으로 나뉜다. FDT 측에서 전역 모델 훈련 절차는 기존의 FL 시스템의 단일 전역 모델 훈련 방식을 PDT 로컬 모델 유사도에 따른 부분적 훈련 및 집계 방식으로 변경한다. 따라서 FDT는 각 PDT의 로컬 모델의 유사도를 추정하기 위해 PDT 데이터 저장소의 데이터에 대한 임베딩 벡터를 사전에 수집하고 인덱싱한다. FDT에서 모델 훈련을 수행할 때 PDT의 로컬 모델은 인덱싱된 임베딩 벡터와 유사도를 비교하여 가장 유사한 벡터를 가진 PDT 중 우선순위가 높은 클라이언트를 참여 클라이언트 그룹으로 선정하여 훈련한다. 이 때 클라이언트 선택의 우선순위는 aging-term 기반으로 가장 오랫동안 선택을 받지 못하거나 훈련을 완료하지 못한 클라이언트에 가중되어 산정된다. 또 PDT 측에서의 로컬 모델 훈련은 기존 고정된 FL 훈련 방식이 아닌 낙오자 문제를 완화를 고려하여 적응형 훈련 방식을 채택한다. 이러한 훈련 절차의 변경은 FACTS가 PDT의 자원 다양성 및 공정성을 고려하면서 로컬 모델의 일반화 성능을 최대화할 수 있게 한다.

4.2 FACTS training procedure

4.2.1 Initialize stage

FDT 시스템에는 다수의 PDT가 등록되고, 각 PDT는 서로 다른 환경에서 독립적으로 운용된다. FACTS를 디지털 트윈에 연계하기 위해 FDT는 PDT가 등록되는 시점에 PDT의 메타데이터 정보를 등록함과 동시에 초기화 단계를 수행한다. 초기화 단계는 PDT에서 훈련 요청이 오기 전 FDT 측에서의 훈련 준비 과정으로서, 전역 모델을 복사하고 저장한다. 여기서 전역 모델은 훈련되지 않은 초기 모델로 PDT의 시뮬레이션의 목적 및 활용 방식에 따라 다양한 구조를 취할 수 있다. 한편, FDT에서 등록된 PDT 간 로컬 모델 유사도를 추정하기 위해서 PDT 데이터 저장소의 데이터에 대한 임베딩 벡터를 요청한다. 각 PDT의 로컬 데이터에 대한 임베딩 모델에 따른 임베딩 벡터는 다음과 같이 생성된다.

$$\frac{1}{d_{out}(L)}\sum_{i=1}^{d_{out}(L)}(f^{(L)}(\sum_{j=1}^{d_{in}(L)}W_{i,j}^{(L)} \cdot f^{(L-1)}(\ldots f^{(2)}(W_{i,j}^{(2)} \cdot f^{(1)}(W_{i,j} \cdot X + b_i^{(1)}) + b_i^{(2)})\ldots)) \quad (6)$$

여기서 $X \in R^{d_{in}}$는 입력데이터이고 L은 임베딩 모델의 레이어 개수이다. $W^{(l)} \in R^{d_{out}(l) \times d_{in}(l)}$은 각 레이어의 가중치 행렬이고 l은 레이어의 인덱스이다. $b^{(l)}$은 편향 벡터, $f^{(l)}$은 각 레이어의 활성화 함수, $d_{in}(L)$은 첫 레이어의 입력 차원, $d_{out}(L)$은 마지막 레이어의 출력 차원이다. 이렇게 생성된 임베딩 벡터를 FDT는 벡터 데이터베이스에 저장하고 인덱싱한다. PDT의 로컬 데이터의 변경 및 업데이트를 고려해 각 PDT는 로컬 데이터를 증분 데이터 세그먼트와 저장 데이터 세그먼트로 분할하는데, 증분 데이터 세그먼트가 사전 정의된 임계값에 도달하면 저장 데이터 세그먼트와 통합하고 임베딩 벡터를 재생성해 FDT에 갱신요청을 보낸다. FDT 또한 변경된 임베딩 벡터의 수 또는 시간이 일정 임계값에 도달하면 재 인덱싱을 수행한다. 이때 벡터의 유형에 관계없이 인덱스 구축에는 Kmeans와 같은 클러스터 기반 인덱싱을 활용한다. 마지막으로 FDT는 등록된 PDT의 우선순위를 0으로 초기화한다. 우선순위는 참여 클라이언트의 훈련 공정성을 확보하고 보다 다양한 데이터를 훈련하기 위해 Aging-term based client selection method (ACS)기반으로 결정된다. 이때 ACS는 적응형 모델 훈련 방식을 채택하여 부분업데이트를 집계하도록 고안되었다. ACS를 통한

PDT의 우선순위 $P_i[t]$는 다음과 같이 결정된다.

$$P_i[t] = (K \times t - u_i + 1) \times (A_i[t-1] + 1) \times (1 - I_i[t]) \quad (7)$$

여기서 K는 서버가 지정한 최대 local update 횟수이며 u_i는 클라이언트 i가 훈련한 누적 local update 성공횟수이다. 또 $A_i[t]$는 스케쥴링 요소 i의 시간 단위 t에서의 aging term으로 다음과 같이 표현된다.

$$A_i[t+1] = (A_i[t] + 1) \times (1 - I_i[t]), I_i[t] \in \{0,1\} \quad (8)$$

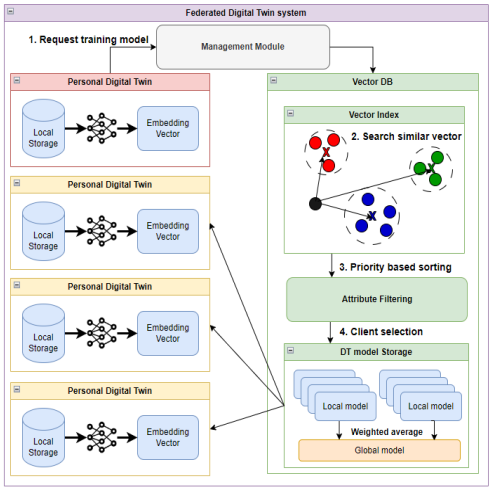
이 때, $I_i[t]$는 스케쥴링 요소 i의 시간 단위 t에서의 자원 할당 여부를 의미하는 지시변수 (Indicator variable)로 다음과 같다.

$$I_i[t] = \begin{cases} 1, & If\ client\ i\ was\ selected\ in\ (t-1) \\ 0, & Otherwise \end{cases} \quad (9)$$

이와 같은 절차를 통해 FDT 시스템는 등록된 PDT를 관리하고 훈련 내역을 추적해 공정한 클라이언트 선택을 수행한다.

4.2.2 Client selection & broadcast stage

FACTS의 클라이언트 선택은 PDT의 훈련 요청으로부터 시작된다. 그림 5와 같이 FACTS의 클라이언트 선택



(그림 5) FACTS의 참여 클라이언트 선택
(Figure 5) Participating client selection in FACTS

은 4개의 절차로 구성된다. 먼저, 첫 번째 절차에서는 특정 PDT의 시뮬레이션이 활성화됨에 따라 PDT는 시뮬레이션 모델 생성을 위해 FDT에 훈련 요청과 함께 임베딩 벡터를 전송한다. 다음으로 FDT는 훈련 요청에 따라 벡터 데이터베이스의 검색엔진에 PDT의 임베딩 벡터와 가장 유사한 $N \times \alpha$개의 이웃을 질의한다. 여기서 N은 한 번의 global iteration에 참여하는 클라이언트 수이며 α는 사전 설정된 배수항이다. 세 번째 절차에서 FDT는 속성 필터링을 통해 우선순위가 높은 $N-1$개의 PDT를 선택하고 훈련을 요청한 PDT는 고정으로 훈련에 참여한다. 그 후 FDT는 DT 모델 저장소에서 해당 PDT의 로컬 모델을 불러와 PDT의 데이터 크기에 따른 가중 평균을 수행해 전역 모델을 생성한다. 그리고 각 참여 PDT에 전역 모델과 훈련기한, 최대 업데이트 수를 전송한다. 이와 같은 절차를 통해 FACTS는 훈련을 요청한 PDT와 유사한 로컬 데이터를 갖는 참여 클라이언트 집합을 선정해 훈련 모델의 과소적합 또는 과대적합을 방지하고, 훈련모델의 일반화 성능을 향상시킬 수 있다.

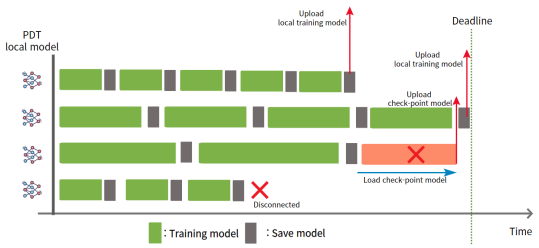
4.2.3 Local model training stage

FDT에 의해 모델과 훈련 기한, 최대 업데이트 수를 전송받은 참여 클라이언트로 선정된 PDT는 각자의 연산 자원을 활용해 로컬 모델을 훈련한다. PDT는 서로 다른 데이터 사이즈 및 분포를 가지고 있으며, 훈련에 활용가능한 연산 자원 또한 서로 상이하다. 이에 FACTS는 각 PDT의 자원 다양성을 고려해 ACS의 적응형 모델 훈련 방식을 채택한다. 즉, 각 PDT는 로컬 자원을 통해 내부적으로 최대 로컬 업데이트 수 미만의 적응형 결정을 내린다. 기존의 FL 시스템의 훈련 방식은 고정된 수의 로컬 업데이트 수를 배포했는데, 이 경우 연산 자원이 부족하거나 데이터가 대규모인 참여 클라이언트의 낙오자 문제가 발생했다. 적응형 로컬 업데이트를 적용하면 컴퓨팅 자원의 손실을 방지하고 보다 많은 PDT의 부분 업데이트 결과를 집계할 수 있다. 적응형 로컬 업데이트를 위해 PDT의 로컬 모델의 손실함수 수식 3은 다음과 같이 수정된다.

$$F_i(w_t^i) = F_i(w_t^i, D_i) = \frac{1}{|D_i|} \sum_{j=1}^{|D_i|} f(w_t^i, d_{i,j}) + \frac{\mu}{2} \| w_t^i - w_t \|^2 \quad (10)$$

여기서 μ는 로컬 모델과 훈련 모델 사이의 일관성을 조율하는 파라미터이다. 한편, FACTS는 모델의 훈련 편향

과 훈련 기한에 따른 로컬 업데이트의 조기종료를 고려하여 PDT의 로컬 모델을 크게 두 개로 나눈다. 하나는 check-point 모델이고 다른 하나는 로컬 훈련 모델이다. 한 번의 로컬 업데이트가 완료될때마다 PDT는 로컬 훈련 모델을 복제하여 check-point 모델로 저장하며, 훈련 기간의 종료가 임박한때에 로컬 업데이트가 완료되지 않은 경우, check-point 모델을 업로드 한다. 이와 같은 절차를 통해 FACTS는 PDT의 로컬 모델 훈련 과정에서 부분 모델 업데이트의 분산에 따른 client drift 문제를 완화할 수 있다.

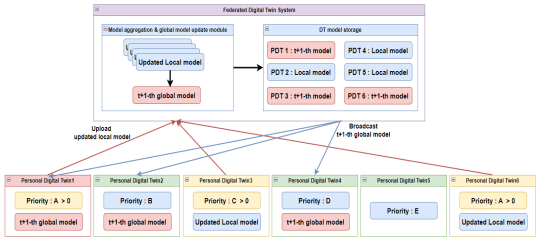


(그림 6) FACTS의 적응형 로컬 업데이트
(Figure 6) Adaptive local update of FACTS

그림 6은 FACTS의 적응형 로컬 업데이트의 절차를 보여준다. 각 PDT 로컬 모델은 1회의 모델 훈련 후, 1회의 check-point 모델 생성을 반복한다. FDT에서 송신된 최대 로컬 업데이트 횟수를 5라고 가정하면 1번째 PCT 로컬 모델의 경우 훈련 기한 내에 5회의 모델 업데이트를 완료하고 성공적으로 결과를 업로드했다. 2번째의 경우 4회의 모델 업데이트를 수행하고 훈련 기한의 마감으로 결과를 업로드 했다. 3번째 PCT 로컬 모델의 경우, 3번째 업데이트 도중 훈련 기한 마감으로 인해 훈련을 중단하고 2번 업데이트된 check-point 모델을 업로드 했다. 4번째의 경우 훈련 도중 연결이 끊긴 PDT이다. 이와 같은 적응형 훈련절차를 통해 기존 FL 시스템의 고정형 훈련 방식에서는 낙오되었을 2,3번째와 같은 클라이언트의 보다 다양한 훈련 결과를 집계할 수 있다.

4.2.4 Aggregation & update model stage

훈련 기한의 마감으로 각 PDT의 업데이트된 로컬 훈련 모델은 FDT로 전송된다. 그림 7과 같이 FDT는 수신된 로컬 훈련 모델을 집계하고 수식 5와 같은 가중평균을 통해 전역 모델을 업데이트한다. 업데이트 된 전역 모델은 복사되어 훈련에 참여한 각 로컬 모델과 교체된다. $t+1$번째 global iteration이 시작되면 FDT는 수식 7에 의



(그림 7) FACTS의 전역 모델 업데이트 & 재배포
(Figure 7) Global model update & broadcasting of FACTS

해 각 PDT의 우선순위 $P_i[t+1]$을 갱신하고 훈련을 요청한 PDT를 제외한 $N-1$개의 참여 클라이언트를 우선순위를 기반으로 재선정하여 전역 모델을 배포하고 훈련을 계속한다. 이와 같은 절차의 반복으로 전역 모델은 PDT의 데이터 특성을 반영하면서 동시에 일반화된 개인화 모델로 점차 수렴한다.

5. Simulation results

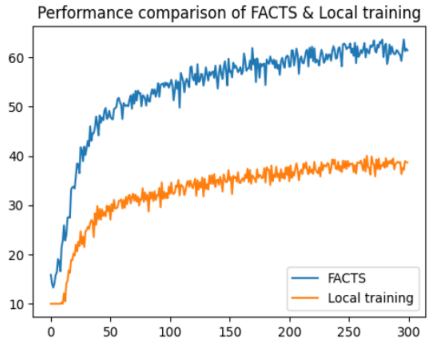
본 절에서는 FACTS의 성능을 평가하기 위해 1) FACTS에 의한 개인화 모델 일반화 성능, 2) ACS에 따른 전역 모델 훈련 성능, 3) 통계적 이질성 정도에 따른 FACTS의 유사도 기반 클라이언트 선택의 개인화 모델 훈련 성능과 같은 광범위한 시뮬레이션을 수행한다. 시뮬레이션 환경을 설정하기 위해 1개의 FDT(서버)에 연결된 500개의 PDT(클라이언트)를 가정한다. 각 클라이언트는 [100, 500] 사이의 로컬 데이터를 보유하고 있으며 각 로컬 데이터는 실험 환경 설정에 따라 [2, 5, 7, 10]의 임의의 class로 구성된다. 이때 훈련 요청 클라이언트의 개인화 성능 테스트를 위해 로컬 데이터에 대한 증강 데이터 1000개를 구성한다. 벤치마크 데이터 셋은 Cifar-10을 활용하며, 총 10개의 class로 구성되어 있다. 각 class는 5000개의 훈련 데이터와 1000개의 테스트 데이터로 구성되어 있는데, 데이터 중복성을 완화하고 성능평가를 위해 데이터 증강을 수행한다. 증강 설정은 image rotation range = [−15, 15], image horizontal flip = 50%, image width shift range = [-15, 15], image height shift range = [-15, 15]이다. FDT는 매 global iteration마다 20개의 클라이언트를 선택해 훈련한다. 이 시뮬레이션에는 두가지 성능 지표가 있다. 하나는 정확도로 훈련 요청 클라이언트의 증강 데이터에 대한 추론 적중률이다. 또 다른 하나

는 훈련 속도로 300번의 global iteration을 수행하고 최종 정확도의 90%를 달성하는데 걸리는 시간이다. 주요 FACTS의 파라미터 설정으로는 α는 2, μ는 0.5이고 최대 훈련 횟수 K는 5, 훈련 기한 T는 K가 5일 때 모든 PDT의 훈련 시간의 평균에 110%로 설정한다.

5.1 Impact of FACTS

(표 1) FACTS에 의한 개인화 모델 성능 비교
(Table 1) Personalization model performance comparison by FACTS

	FACTS	Local training
Accuracy	64.46%	35.67%
Training speed	25 epoch	232 epoch



(그림 8) FACTS에 의한 개인화 모델 성능 비교
(Figure 8) Personalization model performance comparison by FACTS

이 절에서는 FACTS에 의한 개인화 모델의 일반화 성능을 분석한다. 그림 8과 표 1에서 확인할 수 있듯이 FACTS에 의한 개인화 모델 일반화 성능이 Local training한 개인화 모델 성능보다 우수하다. 여기서 Local training은 다른 데이터의 개입 없이 단일 PDT(클라이언트)에서 로컬 데이터 만으로 훈련된 개인화 모델이다. 구체적으로, FACTS는 Local training보다 정확도 측면에서 28.79% 높고, 훈련 속도 측면에서 9.28x 빠르다. 이는 Local training 모델이 non-IID 데이터 분포를 가정했을때 사용자 선호에 편향되고 데이터 다양성이 부족해 다수 번의 모델 업데이트를 수행했을 때 개인화 모델의 로컬데이터에 대한 과대적합이 일어났기 때문이다. 이에 Local training은 모델 업데이트가 반복됨에도 정확도가 상승하지 않고 포화되는 것을 확인할 수 있다. 한편, FACTS는 유사도 기반 클라이언트 선택과 ACS 및 적응형 모델 업데이트를 통해 다양한 클라이언트 로컬 데이터를 통해 모델을 훈련하여 Local training보다 상대적으로 일반화 성능이 향상됨을 확인할 수 있다.

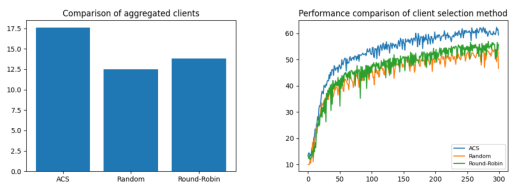
5.2 Impact of ACS

이 절에서는 ACS에 의한 전역 모델 훈련 성능을 분석한다. 그림 9의 왼쪽은 global iteration의 서버 측 Aggregation & global update 단계에서 집계된 참여 클라이언트 수를 비교한 것이다.

(표 2) ACS에 의한 전역 모델 훈련 성능 비교
(Table 2) Global model training performance comparison by ACS

Scheme	Accuracy	TS *	AC **
ACS	61.13%	58	17.5
Random	55.70%	212	12.4
Round-Robin	56.97%	142	13.7

* TS : Training Speed
** AC : number of Aggregated Clients



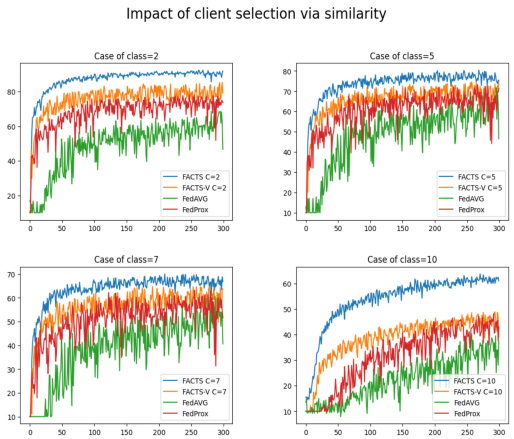
(그림 9) ACS에 의한 전역 모델 훈련 성능 비교
(Figure 9) Global model training performance comparison by ACS

그림에서 확인할 수 있듯이 ACS에 의한 클라이언트 선택이 랜덤 선택이나 Round-Robin 방식과 비교했을 때 적응형 모델 업데이트로 인해 더 많은 클라이언트가 집계된다. 이는 기존 FL scheme의 고정된 훈련방식과는 달리 ACS는 각 참여 클라이언트의 모델 업데이트에 대한 적응적 결정으로 인해 상대적으로 더 많은 클라이언트가 낙오되지 않고 부분 업데이트 된 모델의 훈련 결과를 송부하기 때문이다. 또한, 그림 9의 오른쪽은 ACS에 의한 일반화 성능을 비교하고 있다. 그림 9 및 표 2에서 확인할 수 있듯이 제안된 ACS 방법이 다른 클라이언트 선택 방식보다 정확도 및 훈련 속도 측면에서 우수한 것을 확인할 수 있다. 구체적으로 ACS는 기존 FL scheme의 random, round-robin 선택 방식보다 각각 [5.43%, 4.16%] 더 높다. 이는 random, round-robin 선택 방식의 경우 훈련

기한 내에 모델 업데이트를 완료하지 못한 낙오 클라이언트를 집계과정에서 배제하기 때문에 전역 모델이 상대적으로 더 적은 데이터를 학습하기 때문이다. 한편, ACS는 적응형 모델 훈련으로 인해 낙오되는 클라이언트를 최소화하고 다양한 클라이언트의 훈련 결과를 집계함과 동시에, 우선순위에 따라 상대적으로 훈련이 덜된 클라이언트를 선택하여 보다 공정한 훈련을 수행해 일반화 성능이 향상되었다.

5.3 Impact of client selection via similarity

이 절에서는 유사도에 따른 FACTS의 클라이언트 선택의 성능을 평가한다. 그림은 각 PDT의 로컬 데이터가 [2, 5, 7, 10]개의 class를 가질 때의 개인화 모델 훈련 성능을 비교한다. FACTS-V는 FACTS의 vanilla 버전으로 유사도 기반으로 클라이언트를 선택하지 않고 ACS기반의 적응형 로컬 모델 업데이트를 수행한다.



(그림 10) 유사도 기반 클라이언트 선택의 성능 비교
(Figure 10) Performance comparison of similarity-based client selection

(표 3) 유사도 기반 클라이언트 선택의 모델 성능 비교
(Table 3) Model performance comparison of similarity-based client selection

Scheme	C=2
FACTS	86.73%
FACTS-V	79.17%
FedAVG	66.99%
FedProx	71.93%

실험 결과, 그림 10 및 표 3과 같이 제안하는 FACTS의 유사도 기반 클라이언트 선택 방식이 FACTS-V나 다른 legacy FL scheme (FedAVG[5], FedProx[24])보다 훈련 속도 및 정확도 측면에서 우수함을 확인할 수 있다. 구체적으로, 제안된 방법은 유사도 기반 클라이언트 선택을 수행함으로써 정확도 측면에서 FACTS-V와 비교했을 때 class 수 변화에 따라 각각 [7.56%, 5.64%, 6.25%, 16.89%] 더 높으며, FedAVG[5], FedProx[24]보다 각각 [19.74%, 12.42%, 10.33%, 23.67%], [14.8%, 11.13%, 5.38%, 19.15%] 더 높다. 이는 FACTS가 랜덤하게 클라이언트를 선택하는 대신 유사한 임베딩 벡터를 갖는 클라이언트를 ACS 기반으로 우선적으로 선택하여 데이터의 통계적 이질성을 완화하고 보다 전역 모델이 다양한 데이터를 학습해 일반화 성능을 향상시켰기 때문이다. 한편, FACTS-V는 ACS 기반의 클라이언트 선택에도 불구하고 다른 legacy FL scheme보다 정확도가 더 높은 것을 확인 할 수 있는데, 이는 FACTS-V가 적응형 모델 업데이트를 통해 낙오자를 완화해 하나의 global iteration에서 최대한 많은 클라이언트를 집계해 전역 모델 업데이트가 일부 클라이언트에 편향되는 것을 막고, Aging-term 기반으로 훈련에 참여하지 않은 클라이언트를 우선적으로 선택했기 때문이다.

6. Conclusion

본 논문에서는 개인화 디지털 트윈을 위한 연합학습 기반 클라이언트 훈련 가속 방식을 제안했다. 제안된 방법은 클러스터 기반의 유사 클라이언트 선택 방식과 적응형 모델 훈련 방식을 통해 전역모델의 훈련 성능을 확보함은 물론 로컬 모델의 개인화 가속을 유도한다. 광범위한 시뮬레이션 결과, 유사도 및 Aging-term 기반 참여 클라이언트 선정과 적응형 모델 훈련 방식으로 인해 전역 모델 훈련 시 학습 데이터의 통계적 이질성을 최소화하는 한편 그 규모를 최대화하여 부분 업데이트 클라이언트의 훈련 집계과정에서의 분산을 줄이고 일반화 성능 및 훈련속도를 향상 시켜 기존의 legacy FL체계와 비교했을 때 훈련 효율성 및 성능 측면에서 우수함을 실험결과를 통해 확인한다. 향후 과제로는 디지털 트윈 환경에서 다양한 IoT 기기의 비정형 데이터를 융합하여 개인정보를 보호하면서 사용자 특성을 효과적으로 반영하는 성장가능한 모델 훈련 체계를 설계할 수 있도록 발전시킬 예정이다.

참고문헌(Reference)

[1] Jones, D., Snider, C., Nassehi, A., Yon, J. and Hicks, B., “Characterising the Digital Twin: A systematic literature review,” CIRP Journal of Manufacturing Science and Technology, Vol. 29, pp. 36-52, 2020. https://doi.org/10.1016/j.cirpj.2020.02.002

[2] 정득영 외, “디지털 트윈 기술 K-로드맵 ver 1.0,” 정보 통신기획평가원, 2021. https://iitp.kr/kr/1/knowledge/openReference.it

[3] Barricelli, B. R., Casiraghi, E. and Fogli, D., “A survey on digital twin: Definitions, characteristics, applications, and design implications,” IEEE Access, Vol. 7, pp. 167653-167671, 2019. https://doi.org/10.1109/ACCESS.2019.2953499

[4] Fang, H., and Qian, Q., “Privacy preserving machine learning with homomorphic encryption and federated learning,” Future Internet, Vol. 13, No.4, 2021. https://doi.org/10.3390/fi13040094

[5] McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A., “Communication-efficient learning of deep networks from decentralized data,” In Artificial intelligence and statistics, PMLR, pp. 1273-1282, 2017. https://proceedings.mlr.press/v54/mcmahan17a?ref=https://githubhelp.com

[6] Vepakomma, P., Gupta, O., Swedish, T., and Raskar, R., “Split learning for health: Distributed deep learning without sharing raw patient data,” arXiv preprint arXiv:1812.00564, 2018. https://doi.org/10.48550/arXiv.1812.00564

[7] Thapa, C., Arachchige, P. C. M., Camtepe, S., & Sun, L. “Splitfed: When federated learning meets split learning,” In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, No. 8, pp. 8485-8493, 2022. https://doi.org/10.1609/aaai.v36i8.20825

[8] Grieves, M. “Digital twin: manufacturing excellence through virtual factory replication,” White paper, Vol. 1, pp. 1-7, 2014. https://doi.org/10.5281/zenodo.1493930

[9] He, Y., Guo, J. and Zheng, X., “From surveillance to digital twin: Challenges and recent advances of signal processing for industrial internet of things,” IEEE Signal Processing Magazine, Vol. 35, No. 5, pp. 120-129, 2018. https://doi.org/10.1109/MSP.2018.2842228

[10] Bruynseels, K., Santoni de Sio, F. and Van den Hoven, J., “Digital twins in health care: ethical implications of an emerging engineering paradigm,” Frontiers in genetics, Vol. 9, No. 31, 2018 https://doi.org/10.3389/fgene.2018.00031

[11] Singh, M., Fuenmayor, E., Hinchy, E. P., Qiao, Y., Murray, N., and Devine, D., “Digital twin: Origin to future,” Applied System Innovation, Vol. 4, No. 2, 2021. https://doi.org/10.3390/asi4020036

[12] Han, Y., Niyato, D., Leung, C., Kim, D. I., Zhu, K., Feng, S. and Miao, C., “A dynamic hierarchical framework for IoT-assisted digital twin synchronization in the metaverse,” IEEE Internet of Things Journal, Vol. 10, No. 1, pp. 268-284, 2022. https://doi.org/10.1109/JIOT.2022.3201082

[13] Jia, P., Wang, X., and Shen, X. “Digital-twin-enabled intelligent distributed clock synchronization in industrial IoT systems,” IEEE Internet of Things Journal, Vol. 8, No. 6, pp. 4548-4559, 2020. https://doi.org/10.1109/JIOT.2020.3029131

[14] Jiang, Y., Li, M., Li, M., Liu, X., Zhong, R. Y., Pan, W. and Huang, G. Q., “Digital twin-enabled real-time synchronization for planning, scheduling, and execution in precast on-site assembly,” Automation in Construction, Vol. 141, No. 104397, 2022. https://doi.org/10.1016/j.autcon.2022.104397

[15] Elayan, H., Aloqaily, M., and Guizani, M., “Digital twin for intelligent context-aware IoT healthcare systems” IEEE Internet of Things Journal, Vol. 8, No. 23, pp. 16749-16757, 2021. https://doi.org/10.1109/JIOT.2021.3051158

[16] Kharchenko, V., Illiashenko, O., Morozova, O., and Sokolov, S., “Combination of digital twin and artificial intelligence in manufacturing using industrial IoT,” In IEEE 11th international conference on dependable systems, services and technologies, pp. 196-201, IEEE, 2020. https://doi.org/10.1109/DESSERT50317.2020.9125038

[17] Zhou, Y., Xing, T., Song, Y., Li, Y., Zhu, X., Li, G. and Ding, S., “Digital-twin-driven geometric optimization

of centrifugal impeller with free-form blades for five-axis flank milling," Journal of Manufacturing Systems, Vol. 58, pp. 22-35, 2021. https://doi.org/10.1016/j.jmsy.2020.06.019

[18] Rao, D. J. and Mane, S., "Digital twin approach to clinical dss with explainable ai," arXiv preprint arXiv:1910.13520, 2019. https://arxiv.org/abs/1910.13520

[19] Peter, K. McMahan, B. and Brendan, A. et al., "Advances and open problems in federated learning," Found. Trends Mach. Learn., Vol. 14, pp. 1-210, 2021. http://dx.doi.org/10.1561/2200000083

[20] Jakub, K., McMahan, B., Daniel, R. et al., "Federated optimization: Distributed machine learning for on-device intelligence," arXiv:1610.02527, 2016. https://doi.org/10.48550/arXiv.1610.02527

[21] Nguyen, D.C., Ding, M., Pathirana, P.N., Seneviratne, A., Li, J. and Poor, H.V., "Federated learning for internet of things: A comprehensive survey," IEEE Commun. Surv. Tutor, Vol. 23, pp. 1622-1658, 2021. https://doi.org/10.1109/COMST.2021.3075439

[22] Amirhossein, R., Isidoros, T., Hamed, H., Aryan, M. and Ramtin, P., "Straggler-Resilient Federated Learning: Leveraging the Interplay Between Statistical Accuracy and System Heterogeneity," In Proceedings of the 38th International Conference on Machine Learning, Virtual, pp. 18-24, 2021. https://doi.org/10.1109/JSAIT.2022.3205475

[23] Tao, Y. and Zhou, J., "Straggler Remission for Federated Learning via Decentralized Redundant Cayley Tree," In Proceedings of the 2020 IEEE Latin-American Conference on Communications (LATINCOM), pp. 1-6, 2020. https://doi.org/10.1109/LATINCOM50620.2020.9282334

[24] Li, T., Sahu, A.K., Sanjabi, M., Zaheer, M., Talwalker, A. and Smith, V., "Federated optimization in heterogeneous networks," Proc. Mach. Learn. Syst., Vol. 2, pp. 429-450, 2020. https://doi.org/10.48550/arXiv.1812.06127

[25] Yang, H., Fang, M. and Liu, J., "Achieving Linear Speedup with Partial Worker Participation in Non-IID Federated Learning," In Proceedings of the 9th International Conference on Learning Representations, Virtual, pp. 3-7, 2021. https://doi.org/10.48550/arXiv.2101.11203

[26] Felix, S., Simon, W., Klaus, R.M. and Wojciech, S., "Robust and Communication-Efficient Federated Learning From Non-i.i.d. Data," IEEE Trans. Neural Netw. Learn. Syst. Vol. 31, pp. 3400-3413, 2020. https://doi.org/10.1109/TNNLS.2019.2944481

[27] Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S. J., Stich, S. U., & Suresh, A. T., "SCAFFOLD: Stochastic Controlled Averaging for Federated Learning," arXiv preprint arXiv:1910.06378, 2019. https://doi.org/10.48550/arXiv.1910.06378

[28] Lai, F., Zhu, X., Madhyastha, H.V. and Chowdhury, M., "Oort: Efficient federated learning via guided participant selection," In Proceedings of the 15th USENIX Symposium on Operating Systems Design and Implementation, pp. 19-35, 2021. https://www.usenix.org/conference/osdi21/presentation/lai

[29] Taipalus, Toni, "Vector database management systems: Fundamental concepts, use-cases, and current challenges," Cognitive Systems Research, No. 101216, 2024. https://doi.org/10.1016/j.cogsys.2024.101216

[30] Han, Yikun, Chunjiang Liu, and Pengfei Wang, "A comprehensive survey on vector database: Storage and retrieval technique, challenge," arXiv preprint arXiv: 2310.11703, 2023. https://doi.org/10.48550/arXiv.2310.11703

[31] Taipalus, Toni, "Vector database management systems: Fundamental concepts, use-cases, and current challenges," arXiv preprint arXiv:2309.11322, 2023. https://arxiv.org/abs/2309.11322

[32] Andoni, Alexandr et al., "Practical and optimal LSH for angular distance," in Proc. of Advances in neural information processing systems, Vol. 28, 2015. https://doi.org/10.48550/arXiv.1509.02897

[33] Jégou, Herve, Matthijs Douze, and Cordelia Schmid, "Product quantization for nearest neighbor search," IEEE transactions on pattern analysis and machine intelligence, Vol. 33 No. 1, pp. 117-128, 2011. https://doi.org/10.1109/TPAMI.2010.57

[34] Malkov, Yu A., and Dmitry A. Yashunin, "Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs," IEEE

transactions on pattern analysis and machine intelligence, Vol. 42, No. 4, pp. 824-836, 2020. https://doi.org/10.1109/TPAMI.2018.2889473
[35] Milvus. (n.d.). Milvus. Retrieved June 7, 2024, https://milvus.io/
[36] Chroma. (n.d.). Chroma. Retrieved June 7, 2024, https://www.trychroma.com/
[37] Pinecone. (n.d.). Pinecone. Retrieved June 7, 2024, https://www.pinecone.io/
[38] Weaviate. (n.d.). Weaviate. Retrieved June 7, 2024, https://weaviate.io/
[39] Redis Labs. (n.d.). Redis. Retrieved June 7, 2024, https://redis.io/

◐ 저 자 소 개 ◑

정 영 환(Young-hwan Jeong)
2021년 단국대학교 모바일시스템공학 졸업(학사)
2023년 단국대학교 컴퓨터학과 졸업(석사)
2023년~현재 한국전자기술연구원 연구원
관심분야 : 디지털 트윈, 분산기계학습, 인공지능시스템 및 응용
E-mail : cjstntjd@keti.re.kr

최 원 기(Won-gi Choi)
2014년 연세대학교 컴퓨터과학과 졸업(학사)
2021년 연세대학교 컴퓨터과학과 졸업(박사)
2021년~현재 한국전자기술연구원 선임연구원
관심분야: 디지털 트윈, 빅데이터 플랫폼, 데이터베이스
E-mail: cwk1412@keti.re.kr

계 효 선(Hyoseon Kye)
2021년 숭실대학교 전자정보공학부 IT융합학과 졸업(학사)
2023년 숭실대학교 정보통신공학과 졸업(석사)
2023년~현재 한국전자기술연구원 연구원
관심분야: 디지털트윈, 인공지능, 이상탐지기술, 연합학습
E-mail: hs.kye@keti.re.kr

◐ 저 자 소 개 ◑

김 지 형(Jee-Hyeong Kim)
2015년 한양대학교 ERICA 컴퓨터공학과 졸업(학사)
2020년 한양대학교 컴퓨터과학과 졸업(박사)
2021년~현재 한국전자기술연구원 선임연구원
관심분야: 디지털 트윈, 딥러닝, 강화학습, 자율네트워크
E-mail: jkim8@keti.re.kr

송 민 환(Min-hwan Song)
2003년 건국대학교 정보통신공학과 졸업(학사)
2005년 건국대학교 정보통신공학과 졸업(석사)
2005년~현재 한국전자기술연구원 책임연구원
관심분야: 디지털 트윈, 사물인터넷, 지능형 서비스
E-mail: mhsong@keti.re.kr

이 상 신(Sang-Shin Lee)
1997년 한국외국어대학교 수학과 졸업(학사)
2000년 한국외국어대학교 컴퓨터공학과(공학석사)
2012년 한국외국어대학교 컴퓨터공학과(공학박사)
2000년~현재 한국전자기술연구원 센터장
관심분야: 디지털트윈, 사물인터넷, 웹기반 서비스, 네트워크 시스템
E-mail : sslee@keti.re.kr