(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0083234
(43) 공개일자 2020년07월08일

(51) 국제특허분류(Int. Cl.)
G06N 20/00 (2019.01) *H04L 12/751* (2013.01)
H04L 29/06 (2006.01)
(52) CPC특허분류
G06N 20/00 (2019.01)
H04L 45/08 (2013.01)
(21) 출원번호 10-2019-0168078
(22) 출원일자 2019년12월16일
심사청구일자 2019년12월16일
(30) 우선권주장
1020180171470 2018년12월28일 대한민국(KR)

(71) 출원인
연세대학교 산학협력단
서울특별시 서대문구 연세로 50 (신촌동, 연세대학교)
(72) 발명자
김성륜
서울특별시 용산구 이촌로 303, 32동 1304호(이촌동, 현대아파트)
오승은
제주특별자치도 제주시 대원길 13, 101동 104호(아라일동,영도그린힐)
(뒷면에 계속)
(74) 대리인
특허법인우인

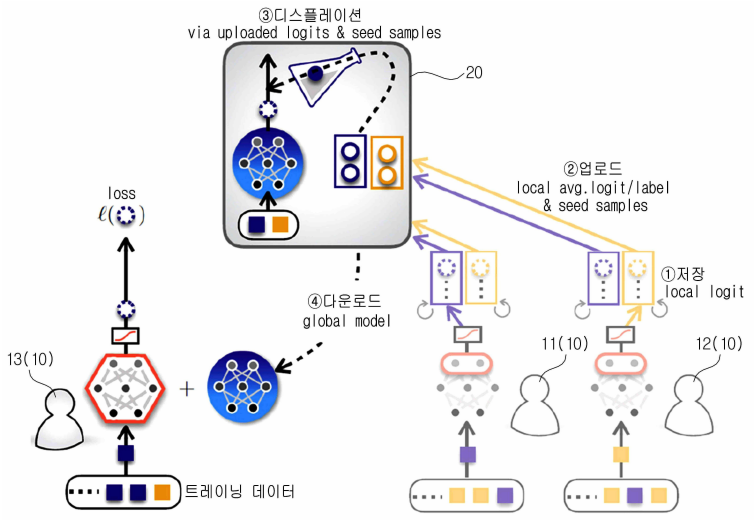
전체 청구항 수 : 총 10 항

(54) 발명의 명칭 **연합된 디스틸레이션 기반의 러닝 구동 방법, 러닝 구동 서버 및 러닝 구동 단말**

(57) 요 약

본 발명에 따르면, 단말이 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓과 시드 샘플들을 서버의 업링크로 전송하고, 상기 서버가 상기 시드 샘플과 상기 로컬 평균 로짓을 기반으로 글로벌 모델의 디스틸레이션(distillation)을 수행하여 분산 네트워크에 발생하는 프라이버시 및 통신 오버헤드 문제를 해결하는 연합된 디스틸레이션 기반의 러닝 구동 방법, 러닝 구동 서버 및 러닝 구동 단말이 개시된다.

대 표 도



(52) CPC특허분류
H04L 63/0407 (2013.01)
(72) 발명자
정은정
서울특별시 동작구 흑석한강로 27, 112동 1004호(흑석동,흑석한강푸르지오)
김혜성
서울특별시 서초구 잠원로 37-48, 207동 1203호(잠원동,신반포4차아파트)

이 발명을 지원한 국가연구개발사업
과제고유번호 1711083489
부처명 과학기술정보통신부
연구관리전문기관 정보통신기술진흥센터
연구사업명 방송통신산업기술개발(R&D)사업
연구과제명 (창조씨앗-2단계) 차세대 5G V2X 서비스 실현을 위한 정밀 측위탐색 연계 고효율 다중안테나 정보전송 및 네트워크 기술 연구
기 여 율 1/1
주관기관 연세대학교 산학협력단
연구기간 2019.01.01 ~ 2019.12.31

명 세 서

청구범위

청구항 1

서버와 다수의 단말들로 구성되는 분산 네트워크에서의 러닝 구동 방법에 있어서,

상기 단말이 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓을 상기 서버의 업링크로 전송하는 단계;

상기 단말이 시드 샘플들을 상기 서버의 업링크로 전송하는 단계; 및

상기 서버가 상기 시드 샘플과 상기 로컬 평균 로짓을 기반으로 글로벌 모델의 디스틸레이션(distillation)을 수행하는 단계;를 포함하는 것을 특징으로 하는 분산 네트워크에서의 러닝 구동 방법.

청구항 2

제1항에 있어서,

상기 서버가 글로벌 모델의 디스틸레이션(distillation)을 수행하는 단계 이전에,

정보 보호를 위해 상기 서버가 상기 시드 샘플들에 랜덤 노이즈를 부여하는 단계;를 더 포함하는 것을 특징으로 하는 분산 네트워크에서의 러닝 구동 방법.

청구항 3

제1항에 있어서,

상기 서버가 상기 시드 샘플과 상기 로컬 평균 로짓을 기반으로 글로벌 모델의 디스틸레이션(distillation)을 수행하는 단계는,

상기 로컬 평균 로짓을 글로벌 모델 파라미터로 변환하는 단계; 및

상기 글로벌 모델 파라미터와 상기 시드 샘플로 상기 글로벌 모델을 트레인(train) 하는 단계;를 포함하는 것을 특징으로 하는 분산 네트워크에서의 러닝 구동 방법.

청구항 4

제3항에 있어서,

상기 트레인(train)한 글로벌 모델을 상기 서버의 다운 링크로 전송하는 단계;를 더 포함하는 것을 특징으로 하는 분산 네트워크에서의 러닝 구동 방법.

청구항 5

제1항에 있어서,

상기 단말이 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓을 상기 서버의 업링크로 전송하는 단계는,

상기 단말이 데이터 샘플들 중에서 로컬 트레인(local train)을 진행하여 나온 로컬 로짓 별로 샘플을 구분하여 각각을 로컬 레이블로 저장하는 단계;

상기 단말이 각각의 로컬 레이블 별로 로컬 평균 로짓을 계산하는 단계; 및

상기 단말이 계산된 상기 로컬 레이블 별 로컬 평균 로짓을 서버로 전송하는 단계;를 포함하는 것을 특징으로 하는 분산 네트워크에서의 러닝 구동 방법.

청구항 6

제5항에 있어서,

상기 다수의 단말들은, 제1 단말 내지 제3 단말을 포함하며,

상기 서버가 상기 제1 단말 및 제2 단말로부터 각각 받은 상기 로컬 레이블 별 로컬 평균 로짓을 이용하여 글로벌 모델을 트레인(train) 하는 단계; 및

상기 제3 단말이 상기 트레인(train)한 글로벌 모델을 상기 서버로부터 전달 받아 손실 함수에 반영하여 제2 로컬 트레인(local train)을 진행하는 단계;를 더 포함하는 것을 특징으로 하는 분산 네트워크에서의 러닝 구동 방법.

청구항 7

제6항에 있어서,

상기 서버가 상기 제1 단말 및 제2 단말로부터 각각 받은 상기 로컬 레이블 별 로컬 평균 로짓을 이용하여 글로벌 모델을 트레인(train) 하는 단계는,

기 설정된 트레인 정확도(train accuracy)가 타겟 이상이 될 때까지 반복되는 것을 특징으로 하는 러닝 구동 방법.

청구항 8

분산 네트워크의 러닝 구동 서버에 있어서,

상기 서버는, 다수의 단말들과 무선 링크를 통해 연결되며,

상기 단말들로부터 상기 단말이 데이터 샘플들을 수집하여 산정한 로컬 평균 로짓을 업링크로 전달 받고,

상기 단말로부터 시드 샘플들을 업링크로 전달 받아,

상기 로컬 평균 로짓을 글로벌 모델 파라미터로 변환하고, 상기 글로벌 모델 파라미터와 상기 시드 샘플로 상기 글로벌 모델을 트레인(train) 하며, 상기 트레인(train)한 글로벌 모델을 상기 서버의 다운 링크로 전송하는 것을 특징으로 하는 분산 네트워크의 러닝 구동 서버.

청구항 9

분산 네트워크의 러닝 구동 단말에 있어서,

상기 단말은, 서버와 무선 링크를 통해 연결되며,

데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓을 상기 서버의 업링크로 전송하며,

시드 샘플들을 상기 서버의 업링크로 전송하는 것을 특징으로 하는 분산 네트워크의 러닝 구동 단말.

청구항 10

분산 네트워크의 러닝 구동 단말에 있어서,

상기 단말은, 서버와 무선 링크를 통해 연결되며,

상기 서버로부터 상기 서버가 로컬 평균 로짓을 글로벌 모델 파라미터로 변환하고, 상기 글로벌 모델 파라미터와 상기 시드 샘플로 상기 글로벌 모델을 트레인(train) 하며, 상기 트레인(train)한 글로벌 모델을 다운 링크로 전달 받아 손실 함수에 반영하여 로컬 트레인(local train)을 진행하는 것을 특징으로 하는 분산 네트워크의 러닝 구동 단말.

발명의 설명

기 술 분 야

[0001] 본 발명은 러닝 구동 방법에 관한 것으로서, 특히 연합된 디스틸레이션 (Federated distillation) 기반 러닝 구동 및 통신 오버헤드 경감 방법에 관한 것이다.

배 경 기 술

[0002] 단말이 보유하고 있는 샘플 수가 제한되어 있는 분산 네트워크 상황에서 각 단말들이 로컬 트레인(local train)을 할 때, 가지고 있는 샘플들에 편향(bias)된 모델을 생성하는 문제점이 발생한다. 이 때, 각 단말들이 서로 정보 교환을 함으로써 로컬 러닝(local learning) 상황에서 발생하는 오버피팅 (overfitting) 문제를 해결하며 전체적인 테스트 정확도를 향상시킬 수 있다.

[0003] 분산 네트워크에서 단말들끼리 원시 데이터 샘플(raw data sample)들을 직접 교환하는 방식은 원시 데이터 샘플의 사이즈와 수를 고려하였을 때, 페이로드 (payload) 사이즈와 통신 오버헤드가 매우 크게 나타난다. 또한, 프라이버시 (privacy)에 대한 보호가 되지 않는다.

발명의 내용

해결하려는 과제

[0004] 본 발명은 러닝 구동 방법에 관한 것으로, 단말이 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓과 시드 샘플들을 서버의 업링크로 전송하고, 상기 서버가 상기 시드 샘플과 상기 로컬 평균 로짓을 기반으로 글로벌 모델의 디스틸레이션(distillation)을 수행하여 분산 네트워크에 발생하는 프라이버시 및 통신 오버헤드 문제를 해결하는 것을 그 목적으로 한다.

[0005] 본 발명의 명시되지 않은 또 다른 목적들은 하기의 상세한 설명 및 그 효과로부터 용이하게 추론할 수 있는 범위 내에서 추가적으로 고려될 수 있다.

과제의 해결 수단

[0006] 상기 과제를 해결하기 위해, 본 발명의 일 실시예에 따른 분산 네트워크에서의 러닝 구동 방법은, 서버와 다수의 단말들로 구성되는 분산 네트워크에서의 러닝 구동 방법에 있어서, 상기 단말이 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓을 상기 서버의 업링크로 전송하는 단계, 상기 단말이 시드 샘플들을 상기 서버의 업링크로 전송하는 단계 및 상기 서버가 상기 시드 샘플과 상기 로컬 평균 로짓을 기반으로 글로벌 모델의 디스틸레이션(distillation)을 수행하는 단계를 포함한다.

[0007] 여기서, 상기 서버가 글로벌 모델의 디스틸레이션(distillation)을 수행하는 단계 이전에, 정보 보호를 위해 상기 서버가 상기 시드 샘플들에 랜덤 노이즈를 부여하는 단계를 더 포함한다.

[0008] 여기서, 상기 서버가 상기 시드 샘플과 상기 로컬 평균 로짓을 기반으로 글로벌 모델의 디스틸레이션(distillation)을 수행하는 단계는, 상기 로컬 평균 로짓을 글로벌 모델 파라미터로 변환하는 단계 및 상기 글로벌 모델 파라미터와 상기 시드 샘플로 상기 글로벌 모델을 트레인(train) 하는 단계를 포함한다.

[0009] 여기서, 상기 트레인(train)한 글로벌 모델을 상기 서버의 다운 링크로 전송하는 단계를 더 포함한다.

[0010] 여기서, 상기 단말이 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓을 상기 서버의 업링크로 전송하는 단계는, 상기 단말이 데이터 샘플들 중에서 로컬 트레인(local train)을 진행하여 나온 로컬 로짓 별로 샘플을 구분하여 각각을 로컬 레이블로 저장하는 단계, 상기 단말이 각각의 로컬 레이블 별로 로컬 평균 로짓을 계산하는 단계 및 상기 단말이 계산된 상기 로컬 레이블 별 로컬 평균 로짓을 서버로 전송하는 단계를 포함한다.

[0011] 여기서, 상기 다수의 단말들은, 제1 단말 내지 제3 단말을 포함하며, 상기 서버가 상기 제1 단말 및 제2 단말로부터 각각 받은 상기 로컬 레이블 별 로컬 평균 로짓을 이용하여 글로벌 모델을 트레인(train) 하는 단계 및 상기 제3 단말이 상기 트레인(train)한 글로벌 모델을 상기 서버로부터 전달 받아 손실 함수에 반영하여 제2 로컬 트레인(local train)을 진행하는 단계를 더 포함한다.

[0012] 여기서, 상기 서버가 상기 제1 단말 및 제2 단말로부터 각각 받은 상기 로컬 레이블 별 로컬 평균 로짓을 이용하여 글로벌 모델을 트레인(train) 하는 단계는, 기 설정된 트레인 정확도(train accuracy)가 타겟 이상이 될 때까지 반복된다.

[0013] 본 발명의 일 실시예에 따른 분산 네트워크의 러닝 구동 서버는, 다수의 단말들과 무선 링크를 통해 연결되며, 상기 단말들로부터 상기 단말이 데이터 샘플들을 수집하여 산정한 로컬 평균 로짓을 업링크로 전달 받고, 상기 단말로부터 시드 샘플들을 업링크로 전달 받아, 상기 로컬 평균 로짓을 글로벌 모델 파라미터로 변환하고, 상기 글로벌 모델 파라미터와 상기 시드 샘플로 상기 글로벌 모델을 트레인(train) 하며, 상기 트레인(train)한 글로

벌 모델을 상기 서버의 다운 링크로 전송한다.

[0014] 본 발명의 일 실시예에 따른 분산 네트워크의 러닝 구동 단말은, 서버와 무선 링크를 통해 연결되며, 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓을 상기 서버의 업링크로 전송하며, 시드 샘플들을 상기 서버의 업링크로 전송한다.

[0015] 본 발명의 일 실시예에 따른 분산 네트워크의 러닝 구동 단말은, 서버와 무선 링크를 통해 연결되며, 상기 서버로부터 상기 서버가 로컬 평균 로짓을 글로벌 모델 파라미터로 변환하고, 상기 글로벌 모델 파라미터와 상기 시드 샘플로 상기 글로벌 모델을 트레인(train) 하며, 상기 트레인(train)한 글로벌 모델을 다운 링크로 전달 받아 손실 함수에 반영하여 로컬 트레인(local train)을 진행한다.

발명의 효과

[0016] 이상에서 설명한 바와 같이 본 발명의 실시예들에 의하면, 단말이 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓과 시드 샘플들을 서버의 업링크로 전송하고, 상기 서버가 상기 시드 샘플과 상기 로컬 평균 로짓을 기반으로 글로벌 모델의 디스틸레이션(distillation)을 수행하여 분산 네트워크에 발생하는 프라이버시 및 통신 오버헤드 문제를 해결할 수 있다.

[0017] 여기에서 명시적으로 언급되지 않은 효과라 하더라도, 본 발명의 기술적 특징에 의해 기대되는 이하의 명세서에서 기재된 효과 및 그 잠정적인 효과는 본 발명의 명세서에 기재된 것과 같이 취급된다.

도면의 간단한 설명

[0018] 도 1 및 도 2는 본 발명의 일 실시예에 따른 연합된 디스틸레이션 기반의 러닝 구동 방법의 분산 네트워크를 나타낸 도면이다.

도 3은 본 발명의 일 실시예에 따른 로짓 벡터의 포맷을 나타낸 도면이다.

도 4는 본 발명의 일 실시예에 따른 FD 알고리즘을 나타낸 도면이다.

도 5는 본 발명의 일 실시예에 따른 FLD 알고리즘을 나타낸 도면이다.

도 6은 본 발명의 일 실시예에 따른 학습 곡선을 나타낸 도면이다.

도 7 및 도 8은 본 발명의 일 실시예에 따른 연합된 디스틸레이션 기반의 러닝 구동 방법을 나타낸 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0019] 이하, 본 발명에 관련된 연합된 디스틸레이션 기반의 러닝 구동 방법, 러닝 구동 서버 및 러닝 구동 단말에 대하여 도면을 참조하여 보다 상세하게 설명한다. 그러나, 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며, 설명하는 실시예에 한정되는 것이 아니다. 그리고, 본 발명을 명확하게 설명하기 위하여 설명과 관계없는 부분은 생략되며, 도면의 동일한 참조부호는 동일한 부재임을 나타낸다.

[0020] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다.

[0021] 제1, 제2 등의 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 구성요소들은 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다.

[0022] 본 발명은 연합된 디스틸레이션 기반의 러닝 구동 방법, 러닝 구동 서버 및 러닝 구동 단말에 관한 것이다.

[0023] 도 1 및 도 2는 본 발명의 일 실시예에 따른 연합된 디스틸레이션 기반의 러닝 구동 방법의 분산 네트워크를 나타낸 도면이다.

[0024] 도 1 및 도 2를 참조하면, 분산 네트워크는 다수의 단말들(10)과 서버(20)로 구성된다. 여기서, 단말의 개수는 본 발명의 일 실시예에 한정되는 것이 아니며, 다수개의 단말을 포함할 수 있다.

[0025] 분산 네트워크에 발생하는 프라이버시 및 통신 오버헤드 문제를 해결하기 위해 페이로드 사이즈가 작으며 갖고 있는 샘플을 직접 전송하지 않으며 교환을 진행했을 때 전체 시스템 테스트 정확도를 향상시킬 수 있는 정보가 필요하다. 본 발명의 일 실시예에 따른 FD(Federated Distillation) 동작 방식은 검증 자료 레이블(Ground-

truth label)을 활용하여 레이블 별로 샘플을 묶고 각 샘플에 해당하는 로짓을 평균내어 얻은 레이블 별 평균로짓벡터를 활용하여 분산 네트워크의 주요 문제점을 해결함과 동시에 각 단말의 테스트 정확도를 끌어올릴 수 있다.

[0026] 본 발명에서는 분산 네트워크에서 각 단말들이 통신 코스트가 적은 정보를 교환하고 이를 바탕으로 러닝을 구동하는 방법을 제안한다. 이를 통해 각 단말의 테스트 정확도(test accuracy)를 보장하며 단말 간 정보 교환 시 발생하는 통신 오버헤드를 감소시킬 수 있다. 또한, 분산 네트워크에서 발생하는 프라이버시 문제를 해결할 수 있다.

[0027] 종래의 경우 단말이 보유하고 있는 샘플 수가 제한되어 있는 분산 네트워크 상황에서 각 단말들이 로컬 트레인(local train)을 할 때, 가지고 있는 샘플들에 편향(bias)된 모델을 생성하는 문제점이 발생한다. 이 때, 각 단말들이 서로 정보 교환을 함으로써 로컬 러닝(local learning) 상황에서 발생하는 오버피팅 (overfitting) 문제를 해결하며 전체적인 테스트 정확도를 향상시킬 수 있다.

[0028] 대표적인 방식으로 단말들이 서로 가지고 있는 원시 데이터 샘플(raw data sample)들을 직접 교환하는 방식이 있다. 직접적으로 원시 데이터 샘플(raw data sample)들을 교환하지 않는 대신에 로컬 트레이닝(local training)을 진행하며 일정 주기마다 중앙의 서버(server)에 학습한 모델의 가중치(weight)를 전송해주고 서버는 여러 단말로부터 받은 모델 가중치(weight)를 평균내어 각 단말로 전송해주는 평균 가중치(averaging weight) 기반 연합 학습(federated learning)이 있다.

[0029] 그 외, 온라인 디스틸레이션(online distillation(co-distillation))의 경우, 일정 주기마다 단말들은 가지고 있는 원시 데이터 샘플(raw data sample)들과 그것을 로컬 러닝 모델(local learning model)에 대입했을 때 나오는 로짓 벡터를 서버에 업로드해주고 서버는 샘플-로짓 페어를 평균내어 저장해둔다. 그 후, 단말들이 로컬 트레인(local train)을 진행할 때 서버에 샘플을 요청해주고 서버는 샘플에 해당하는 로짓을 단말에 전송해준다.

[0030] 종래 기술에서 분산 네트워크에서 단말들끼리 원시 데이터 샘플(raw data sample)들을 직접 교환하는 방식은 원시 데이터 샘플(raw data sample)의 사이즈와 수를 고려하였을 때, 페이로드(payload) 사이즈와 통신 오버헤드가 매우 크게 나타난다. 또한, 프라이버시(privacy)에 대한 보호가 되지 않는다.

[0031] 연합 학습(federated learning)의 경우, 모델 가중치(weight)를 교환하기 때문에 원시 데이터 샘플(raw data sample)을 교환하는 방식 대비 프라이버시가 보장된다. 페이로드 사이즈 또한 비교적 줄어들지만 실제 변동(fluctuation)이 심한 채널에서 전송하기에 한계가 있다.

[0032] 온라인 디스틸레이션(Online distillation)의 경우, 다운링크(downlink, DL)에서 페이로드 사이즈가 작으며 프라이버시가 보장된다. 그러나 업링크(uplink, UL)에서 페이로드 사이즈가 매우 크며 프라이버시 보호 또한 되지 않는다. 또한, 단말이 요청한 원시 데이터 샘플(raw data sample)들을 서버가 가지고 있어야 이득이 생기는 구조이기 때문에 단말들이 가진 샘플들끼리의 상관관계(correlation)에 따라 성능 상승폭이 결정되는 제약이 추가적으로 발생한다.

[0033] 본 발명의 실시예들에 의하면, 검증자료 레이블(Ground-truth label)을 활용하여 레이블 별로 샘플을 묶고 각 샘플에 해당하는 로짓을 평균하여 얻은 레이블 별 평균 로짓 벡터를 활용하여 분산 네트워크에 발생하는 프라이버시 및 통신 오버헤드 문제를 해결할 수 있다.

[0034] 제1 단말(11)과 제2 단말(12)은 각각 데이터 샘플들을 수집하여 로컬 로짓으로 저장한다.

[0035] 이후, 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓을 상기 서버의 업링크로 전송한다.

[0036] 구체적으로, 단말이 데이터 샘플들 중에서 로컬 트레인(local train)을 진행하여 나온 로컬 로짓 별로 샘플을 구분하여 각각을 로컬 레이블로 저장하고, 상기 단말이 각각의 로컬 레이블 별로 로컬 평균 로짓을 계산한 후 상기 단말이 계산된 상기 로컬 레이블 별 로컬 평균 로짓을 서버로 전송한다.

[0037] 제1 단말 및 제2 단말은 각각 로컬 트레인(local train)을 진행하며 나온 로짓을 레이블 별로 저장한다.

[0038] 로짓은 수학식 1을 이용하여 구현할 수 있다. 수학식 1은 예를 들어, 랜덤하게 뽑은 샘플 x의 검증자료 레이블(Ground-truth label)이 n인 경우이다.

수학식 1

[0039] $$L(n, count(n)) = logit(x), count(n) = count(n) + 1$$

[0040] 여기서, logit(x)는 x를 모델에 입력했을 때의 출력 값이며, count(n)은 검증자료 레이블(Ground-truth label)이 n인 샘플의 수를 저장하는 값이다. 위 과정은 뽑은 모든 샘플에 대해 반복된다.

[0041] 본 발명의 일 실시예에 따른 로짓 벡터의 포맷은 도 2에서 설명한다.

[0042] 제1 단말 및 제2 단말(10a, 10b)은 각각 로컬 트레인(local train)을 진행하며 나온 로짓을 레이블 별로 저장한다.

[0043] 단말은 매 T_p 반복(iteration)마다 로컬 레이블 별 평균 로짓 벡터를 계산한다.

[0044] 로컬 레이블 별 평균 로짓 벡터 계산은 수학식 2를 이용하여 구현할 수 있다. 수학식 2는 예를 들어, 단말 d와 검증자료 레이블(ground-truth label) n에 대해 나타낸 것이다.

수학식 2

[0045] $$sum(n) = \sum_{k=1}^{count(n)} L(n,k)$$
$$local(d,n) = sum(n)/count(n)$$

[0046] 여기서, sum(n)은 검증자료 레이블(ground-truth label)이 n인 샘플들에 해당하는 로짓벡터들의 벡터 합이고, local(d, n)은 단말 d에서 검증자료 레이블(ground-truth label) n에 대한 로컬 레이블 별 평균 로짓 벡터이다.

[0047] 위 과정은 모든 검증자료 레이블(ground-truth label)들에 대해 시행된다.

[0048] 제1 단말 및 제2 단말은 계산된 로컬 레이블 별 평균로짓벡터를 서버로 전송한다.

[0049] 서버(20)는 단말들로부터 받은 로컬 레이블 별 평균로짓벡터를 바탕으로 글로벌 레이블 별 평균로짓벡터를 계산한다.

[0050] 글로벌 레이블 별 평균로짓벡터 계산은 수학식 3을 이용하여 구현할 수 있다. 수학식 3은 예를 들어, 검증자료 레이블(ground-truth label) n에 대해 나타낸 것이다.

수학식 3

[0051] $$global(n) = \sum_{d=1}^{D} sum(n,d)/D$$

[0052] 여기서, global(n)은 검증자료 레이블(ground-truth label)이 n에 대한 글로벌 레이블 별 평균로짓벡터이며 D는 분산 네트워크에 참여하는 모든 단말들의 수이다.

[0053] 위 과정은 모든 검증자료 레이블(ground-truth label)들에 대해 시행된다.

[0054] 제3 단말(30)은 서버로부터 받은 글로벌 레이블 별 평균로짓벡터를 손실 함수에 반영하여 로컬 트레인(local train)을 진행하며, 단말의 트레인 정확도(train accuracy)가 타겟 이상이 될 때까지 도 1에 나타난 과정을 반복한다.

[0055] 또한, 제1 단말(11)과 제2 단말(12)은 시드 샘플들을 상기 서버의 업링크로 전송한다.

[0056] 제1 단말(11)과 제2 단말(12)은, 서버와 무선 링크를 통해 연결되며, 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓을 상기 서버의 업링크로 전송하며, 시드 샘플들을 상기 서버의 업링크로 전송한다.

[0057] 여기서, 시드 샘플을 업링크로 전송하기 위해 단말이 서로 다른 라벨을 갖는 시드 샘플들을 무작위로 선택하고, 선택한 상기 시드 샘플들을 기 설정된 혼합비로 선형 결합한다.

[0058] 또한, 서버는 글로벌 모델의 디스틸레이션(distillation)을 수행하는 단계 이전에, 정보 보호를 위해 상기 서버가 상기 시드 샘플들에 랜덤 노이즈를 부여할 수 있다.

[0059] 서버(20)는 시드 샘플과 상기 로컬 평균 로짓을 기반으로 글로벌 모델의 디스틸레이션(distillation)을 수행한다.

[0060] 구체적으로, 서버는, 다수의 단말들과 무선 링크를 통해 연결되며, 상기 단말들로부터 상기 단말이 데이터 샘플들을 수집하여 산정한 로컬 평균 로짓을 업링크로 전달 받고, 상기 단말로부터 시드 샘플들을 업링크로 전달 받아, 상기 로컬 평균 로짓을 글로벌 모델 파라미터로 변환하고, 상기 글로벌 모델 파라미터와 상기 시드 샘플로 상기 글로벌 모델을 트레인(train) 하며, 상기 트레인(train)한 글로벌 모델을 상기 서버의 다운 링크로 전송한다.

[0061] 글로벌 모델의 디스틸레이션(distillation)을 수행하는 것은, 상기 로컬 평균 로짓을 글로벌 모델 파라미터로 변환하고, 상기 글로벌 모델 파라미터와 상기 시드 샘플로 상기 글로벌 모델을 트레인(train) 한다.

[0062] 이후, 서버는 상기 트레인(train)한 글로벌 모델을 상기 서버의 다운 링크로 전송하고, 제3 단말(13)은 트레인(train)한 글로벌 모델을 상기 서버로부터 전달 받아 손실 함수에 반영하여 트레이닝 데이터로 로컬 트레인(local train)을 진행하게 된다.

[0063] 제3 단말(13)은, 서버와 무선 링크를 통해 연결되며, 상기 서버로부터 상기 서버가 로컬 평균 로짓을 글로벌 모델 파라미터로 변환하고, 상기 글로벌 모델 파라미터와 상기 시드 샘플로 상기 글로벌 모델을 트레인(train) 하며, 상기 트레인(train)한 글로벌 모델을 다운 링크로 전달 받아 손실 함수에 반영하여 로컬 트레인(local train)을 진행한다.

[0064] 즉, 서버가 상기 제1 단말 및 제2 단말로부터 각각 받은 상기 로컬 레이블 별 로컬 평균 로짓을 이용하여 글로벌 모델을 트레인(train) 하고, 상기 제3 단말이 상기 트레인(train)한 글로벌 모델을 상기 서버로부터 전달 받아 손실 함수에 반영하여 제2 로컬 트레인(local train)을 진행하게 되는 것이다.

[0065] 기존 연합 학습(federated learning)과 비교하였을 때, 업링크 및 다운링크의 페이로드 사이즈를 줄이는 것이 가능하나 러닝의 최종 테스트 정확도(test accuracy) 측면에서 보았을 때 손실이 발생한다.

[0066] 서버와 디바이스들로 구성된 일반적인 단말 시스템(cellular system)에서는 단말(device)들의 업링크 전송 파워(uplink transmission power)가 균등하게 나타난다. 채널 용량(channel capacity)이 부족한 업링크에서 레이블 별 평균 로짓 벡터 전송을 활용하고, 용량(capacity)이 상대적으로 넉넉한 다운링크(downlink)에서는 연합 학습(federated learning)에서처럼 모델 가중치(weight)전송을 활용하여 다운링크(downlink)- 업링크(uplink)의 채널 용량 제한(channel capacity constraint)를 만족시키며 최종 테스트 정확도(test accuracy)에서 향상된 성능을 기대할 수 있다. 이러한 구조가 성립되기 위해서는 단말(device)들의 업링크 전송 시 시드 샘플(seed sample) 몇 개를 추가로 보내줌으로써 시드 샘플과 평균 로짓 벡터 값을 바탕으로 중앙의 서버가 글로벌 모델(global model)을 트레인(train)하여 이의 모델 가중치(model weight)를 다운링크로 전송해 줄 수 있다.

[0067] 도 3은 본 발명의 일 실시예에 따른 로짓 벡터의 포맷을 나타낸 도면이다.

[0068] 로짓 벡터의 사이즈는 단말이 지도 학습(Supervised learning)을 통해 분류하고자 하는 총 레이블 수와 같다.

[0069] 입력 샘플에 대해 로짓 벡터(110)가 정해졌을 때, 벡터 내 각 원소의 값이 의미하는 바는 현재 단말이 가지고 있는 모델이 샘플을 해당 레이블(100)로 분류할 확률과 같다.

[0070] 예를 들어, 단말 d의 총 데이터 샘플 수가 N이며, 분류하고자 하는 레이블의 집합(120)이 {1, 2, 3}이라 주어졌을 때, 로짓벡터는 도 3에 나타난 바와 같이 구현된다.

[0071] 도 4는 본 발명의 일 실시예에 따른 FD 알고리즘을 나타낸 도면이다.

[0072] 도 4에 나타난 바와 같이 연합된 디스틸레이션(Federated Distillation) 알고리즘은 예측 함수(Prediction

function): F(w, input), 손실 함수(Loss function): Φ(F, label), Ground-truth label: y_{input}을 요구한다.

[0073] 설정된 S는 모든 장치의 전체 데이터 세트를 나타내며, B는 각 장치에서 묶인 집단을 나타낸다.

[0074] 함수 F(w, a)는 소프트맥스 함수(softmax function)에 의해 정규화된 로짓 벡터로서, 여기서 w와 a는 모델의 무게와 입력이다.

[0075] 함수 Φ(p, q)는 p와 q 사이의 교차 엔트로피로서, 손실 함수(Loss function)와 distillation 정규화(regularizer)에 모두 사용된다.

[0076] 여기서, η는 학습율(learning rate)상수, γ는 distillation 정규화(regularizer)의 가중치 파라미터이다.

[0077] i번째 디바이스에서 $\bar{F}_{k,\ell}^{(i)}$는 트레이닝 샘플이 l번째 ground-truth label에 해당하고, k번 반복한 로컬 레이블 별 평균 로짓 벡터이다.

[0078] $\hat{F}_{k,\ell}^{(i)}$는 글로벌 레이블 별 평균 로짓 벡터이며, 수학식 4로 구현된다.

수학식 4

[0079] $$\hat{F}_{k,\ell}^{(i)} = \sum_{j \neq i} \bar{F}_{k,\ell}^{(j)} / (M-1)$$

[0080] 여기서, M은 분산 네트워크에 참여하는 모든 단말들의 수 이다.

[0081] 또한, $cnt_{k,\ell}^{(i)}$는 ground-truth label이 l인 샘플의 수이다.

[0082] 도 5는 본 발명의 일 실시예에 따른 FLD 알고리즘을 나타낸 도면이다.

[0083] 도 5에 나타난 바와 같이 FLD(Federated Learning after Distillation) 알고리즘은 아웃풋 업로드, 믹스업, 아웃풋-모델 변환, 역-믹스업, 모델 다운로드 과정을 포함한다.

[0084] 아웃풋-모델 변환의 핵심 아이디어는 $G^{P}_{out,n}$의 지식을 가중치 벡터 G^{P}_{mod}를 가진 글로벌 모델로 변환하는 것이다.

[0085] 이를 활성화하려면 처음에 (예: p = 1) 각각의 단말들은 로컬 데이터 세트에서 임의로 선택된 N_s시드 샘플들을 업로드한다.

[0086] 글로벌 가중치 벡터 $w_s^{(k)}$는 수학식 5로 나타난다.

수학식 5

[0087] $$\mathbf{w}_s^{(k+1)} = \mathbf{w}_s^{(k)} - \eta \nabla \left(\phi(\mathbf{F}_{s,n}^{[i_k]}, \mathbf{L}_n^{[i_k]} | \mathbf{w}_s^{(k)}) + \beta \psi(\mathbf{F}_{s,n}^{[i_k]}, \mathbf{G}_{\text{out},n}^{p}) \right)$$

[0088] 여기서, $F_{s,n}^{[ik]}$ 은 n 번째 레이블 인 경우 글로벌 모델의 출력 벡터이다.

[0089] 서버는 모든 장치에서 다운로드 한 $G^{P}_{mod} = w_s^{(ks)}$ 를 산출한다.

[0090] 도 6은 본 발명의 일 실시예에 따른 학습 곡선을 나타낸 도면이다.

[0091] 도 6은 IID가있는 비대칭 및 대칭 (P up = P dn = 40 dBm, W up = W dn = 10 MHz) 채널에서 FL, FD 및 MixFLD와 비교하여 Mix2FLD에서 임의로 선택된 장치의 학습 곡선 및 비 IID 데이터 세트를 나타낸 것이다.

[0092] 도 6은 Mix2FLD가 비대칭 및 대칭 채널 조건에서 최고 정확도와 가장 빠른 수렴을 달성함을 보여준다. FL 업로드 모델 가중치와 비교하여 Mix2FLD의 모델 출력 업로드는 업 링크 페이로드 크기를 최대 622.4 배 줄인다. 업

링크 용량이 제한적인 비대칭 채널 (도 6의 (a) 및 (c))에서는 보다 빈번하고 성공적인 업로드가 가능하여 최대 12 % 더 높은 정확도와 4.6 배 빠른 수렴을 달성한다.

[0093] FD와 비교하여 Mix2FLD는 글로벌 모델 가중치를 다운로드하기 위해 높은 다운 링크 용량을 활용하는데, 이는 모델 출력을 다운로드하는 것보다 더 높은 정확도를 제공한다. 또한 Mix2FLD의 글로벌 정보는 단순히 FD에서 사용되는 로컬 출력을 평균하는 것이 아니라 시드 샘플을 수집하고 글로벌 데이터 분포를 반영하여 구성된다. 이에 따라 Mix2FLD는 FD보다 최대 15 % 높은 정확도와 36 % 빠른 수렴을 달성합니다.

[0094] IID 데이터 세트가 있는 대칭 채널 (도 6의 (b))에서 Mix2FLD와 FL은 가장 높은 정확도를 달성한다. 그럼에도 불구하고 Mix2FLD는 더 작은 업 링크 페이로드 크기와 더 빈번한 업데이트 덕분에 FL보다 3.1 배 더 빠르게 수렴한다.

[0095] 지연 시간, 프라이버시 및 정확도 트레이드 오프의 모든 경우에 Mix2FLD 및 MixFLD에서 시드 샘플 양 (N_s = 10)을 줄이면 정확성이 저하되어 빠른 수렴 시간이 제공되어 지연 시간 정확도의 트레이드 오프가 발생한다.

[0096] 도 7 및 도 8은 본 발명의 일 실시예에 따른 연합된 디스틸레이션 기반의 러닝 구동 방법을 나타낸 흐름도이다.

[0097] 도 7을 참조하면, 본 발명의 일 실시예에 따른 연합된 디스틸레이션 기반의 러닝 구동 방법은, 단계 S110에서 단말이 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 상기 로컬 평균 로짓을 상기 서버의 업링크로 전송한다.

[0098] 단계 S120에서 상기 단말이 시드 샘플들을 상기 서버의 업링크로 전송한다.

[0099] 단계 S130에서 상기 서버가 상기 시드 샘플과 상기 로컬 평균 로짓을 기반으로 글로벌 모델의 디스틸레이션(distillation)을 수행한다.

[0100] 단계 S140에서 상기 트레인(train)한 글로벌 모델을 상기 서버의 다운 링크로 전송한다.

[0101] 단계 S150에서 단말이 상기 트레인(train)한 글로벌 모델을 상기 서버로부터 전달 받아 손실 함수에 반영하여 로컬 트레인(local train)을 진행한다.

[0102] 구체적으로 설명하면, 도 8을 참조하면, 본 발명의 일 실시예에 따른 연합된 디스틸레이션 기반의 러닝 구동 방법은, 단계 S210에서 상기 단말이 데이터 샘플들 중에서 로컬 트레인(local train)을 진행한다.

[0103] 단계 S220에서 단말이 로컬 로짓 별로 샘플을 구분하여 각각을 로컬 레이블로 저장한다.

[0104] 단계 S230에서 상기 단말이 각각의 로컬 레이블 별로 로컬 평균 로짓을 계산한다.

[0105] 단계 S240에서 상기 단말이 계산된 상기 로컬 레이블 별 로컬 평균 로짓을 서버로 전송한다.

[0106] 단계 S250에서 서버가 상기 제1 단말 및 제2 단말로부터 각각 받은 상기 로컬 레이블 별 로컬 평균 로짓을 이용하여 글로벌 모델을 트레인(train)한다.

[0107] 단계 S260에서 상기 제3 단말이 상기 트레인(train)한 글로벌 모델을 상기 서버로부터 전달 받아 손실 함수에 반영하여 제2 로컬 트레인(local train)을 진행한다.

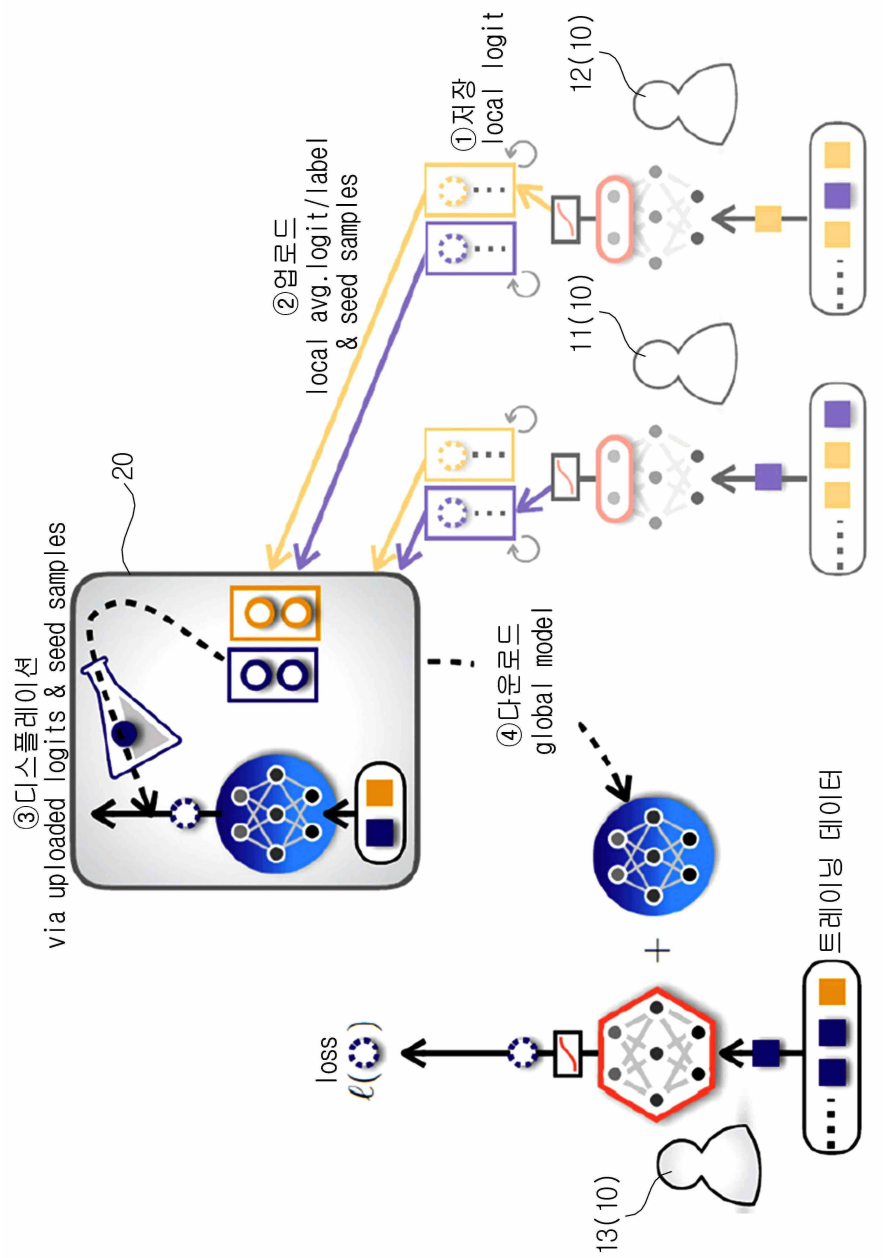
[0108] 단계 S270에서 기 설정된 트레인 정확도(train accuracy)가 타겟 이상이 되는지 확인하며, 타겟 미만인 경우 될 때까지 단계 S210 내지 단계 S260을 반복한다.

[0109] 이상의 설명은 본 발명의 일 실시예에 불과할 뿐, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자는 본 발명의 본질적 특성에서 벗어나지 않는 범위에서 변형된 형태로 구현할 수 있을 것이다. 따라서 본 발명의 범위는 전술한 실시예에 한정되지 않고 특허 청구 범위에 기재된 내용과 동등한 범위 내에 있는 다양한 실시 형태가 포함되도록 해석되어야 할 것이다.

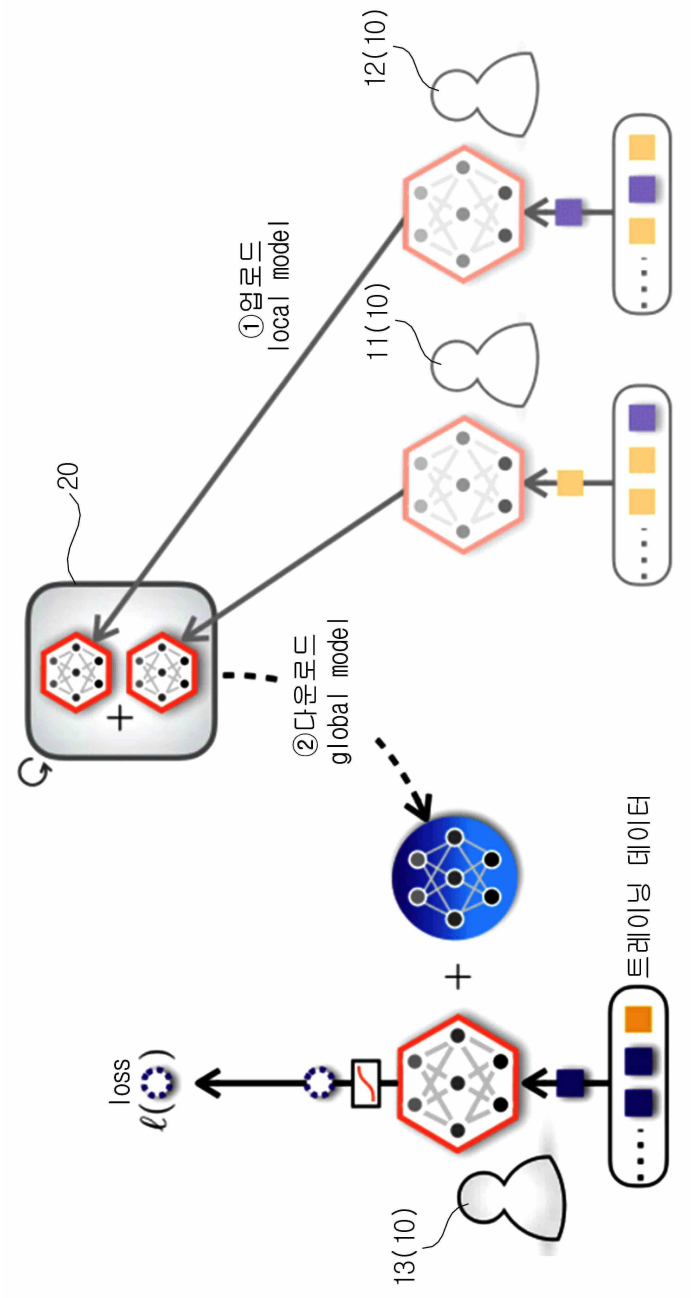
부호의 설명

도면

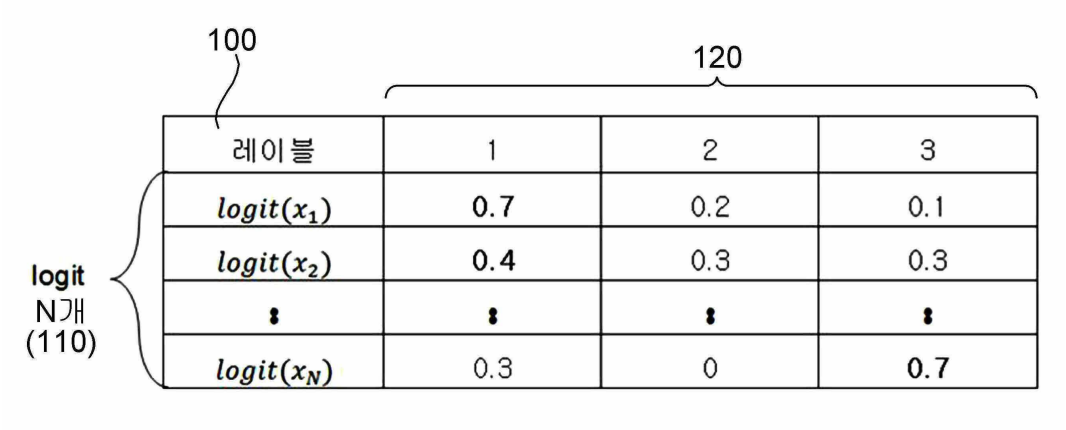
도면1


①저장
local logit
②업로드
local avg.logit/label
& seed samples
③디스틸레이션
via uploaded logits & seed samples
④다운로드
global model
20
12(10)
11(10)
13(10)
loss
$\ell(\cdot)$
트레이닝 데이터

도면2


12(10)
①업로드
local model
11(10)
20
②다운로드
global model
트레이닝 데이터
loss
ℓ( )
13(10)

도면3



도면4

Algorithm 1 Federated distillation (FD)

Require: Prediction function: $F(w, input)$, Loss function: $\phi(F, label)$, Ground-truth label: y_{input}

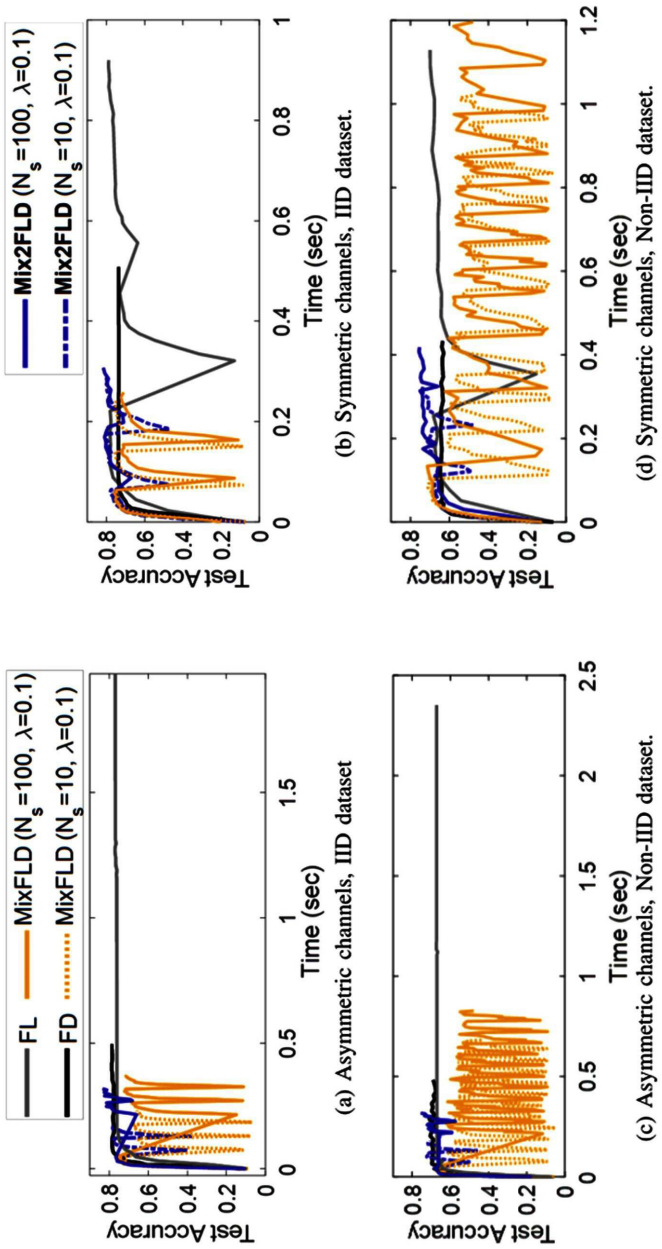
while not converged **do**
  procedure LOCAL TRAINING PHASE (at each device)
    for n steps **do** : $B, y_B \leftarrow \mathbb{S}$
      for sample $b \in B$ **do**
        $w^{(i)} \leftarrow w^{(i)} - \eta\nabla\{\phi(F(w^{(i)}, b), y_b) + \gamma \cdot \phi(F(w^{(i)}, b), \hat{F}^{(i)}_{k,y_b})\}$
        $F^{(i)}_{k,y_b} \leftarrow F^{(i)}_{k,y_b} + F(w^{(i)}, b)$, $cnt^{(i)}_{k,y_b} \leftarrow cnt^{(i)}_{k,y_b} + 1$
    for label $\ell = 1, 2, \cdots, L$ **do**
      $\bar{F}^{(i)}_{k,\ell} \leftarrow F^{(i)}_{k,\ell}/cnt^{(i)}_{k,\ell}$: **return** $\bar{F}^{(i)}_{k,\ell}$ to server
  procedure GLOBAL ENSEMBLING PHASE (at the server)
    for each device $i = 1, 2, \cdots, M$ **do**
      for label $\ell = 1, 2, \cdots, L$ **do**
        $\bar{F}_{k,\ell} \leftarrow \bar{F}_{k,\ell} + \bar{F}^{(i)}_{k,\ell}$
    for each device $i = 1, 2, \cdots, M$ **do**
      for label $\ell = 1, 2, \cdots, L$ **do**
        $\hat{F}^{(i)}_{k+1,\ell} \leftarrow \bar{F}_{k,\ell} - \bar{F}^{(i)}_{k,\ell}$, $\hat{F}^{(i)}_{k+1,\ell} \leftarrow \hat{F}^{(i)}_{k+1,\ell}/(M-1)$: **return** $\hat{F}^{(i)}_{k+1,\ell}$ to device i
end while

도면5

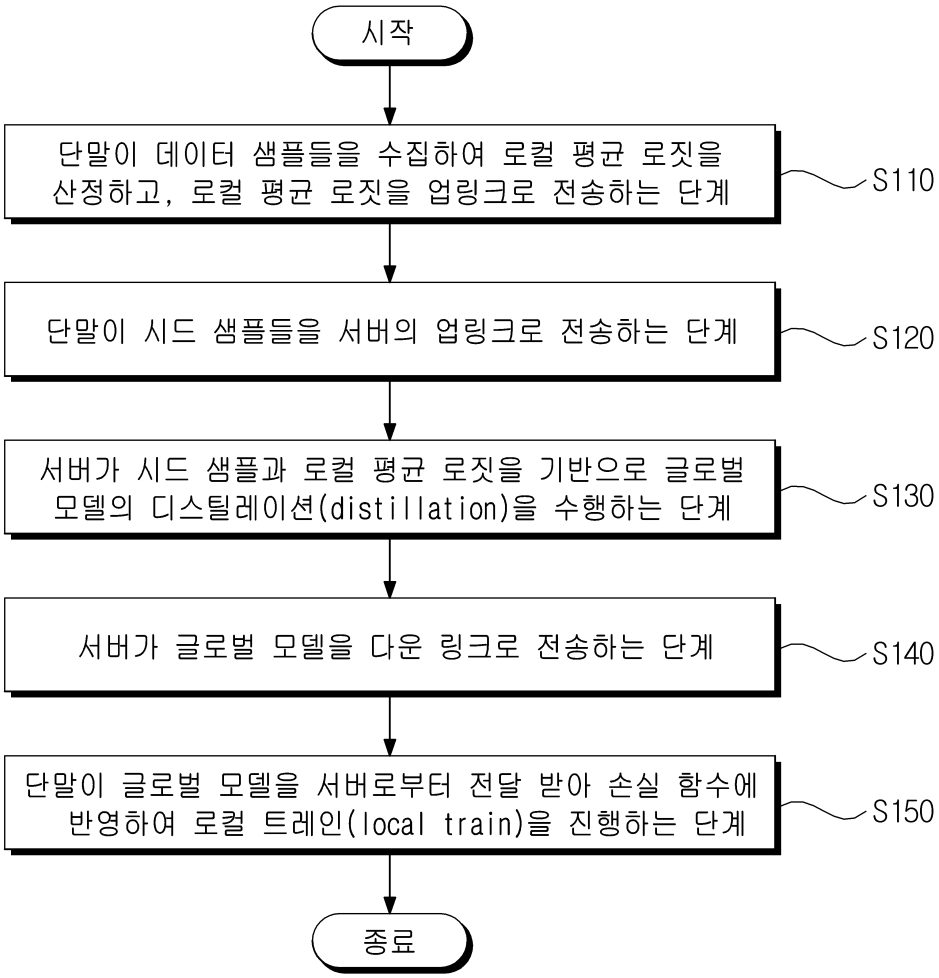
Algorithm 1 FLD with Mix2up (**Mix2FLD**)

Require: $\{\mathbf{S}_d\}$ with $d \in \mathcal{D}$, $\lambda \in (0, 1)$
while $|\mathbf{G}_{\text{out},n}^{p} - \mathbf{G}_{\text{out},n}^{p-1}|/|\mathbf{G}_{\text{out},n}^{p-1}| \geq \varepsilon$ **do**
Device $d \in \mathcal{D}$: ▷ *Output upload*
  if $p=1$ **generates** $\{\hat{\mathbf{s}}_d^{[i,j]}\}$ via (6) **end if** ▷ *Mixup*
  updates $\mathbf{w}_d^{(k)}$ in (1) and $\bar{\mathbf{F}}_{d,n}^{p}$ in (2) for K iterations
  unicasts $\{\bar{\mathbf{F}}_{d,n}^{p}\}$ (with $\{\hat{\mathbf{s}}_d^{[i,j]}\}$ **if** $p = 1$) to the server
Server: ▷ *Output-to-model conversion*
  if $p=1$ **generates** $\{\tilde{\mathbf{s}}_{d,d',n}^{[i,j][i',j']}\}$ via (7) **end if** ▷ *Inverse-Mixup*
  computes $\mathbf{G}_{\text{out},n}^{p}$
  updates $\mathbf{w}_s^{(k)}$ via (5) for K_s iterations
  broadcasts $\mathbf{G}_{\text{mod}}^{p} = \mathbf{w}_s^{(K_s)}$ to all devices
$p \leftarrow p + 1$
Device $d \in \mathcal{D}$ **substitutes** $\mathbf{w}_d^{(0)}$ with $\mathbf{G}_{\text{mod}}^{p}$ ▷ *Model download*
end while

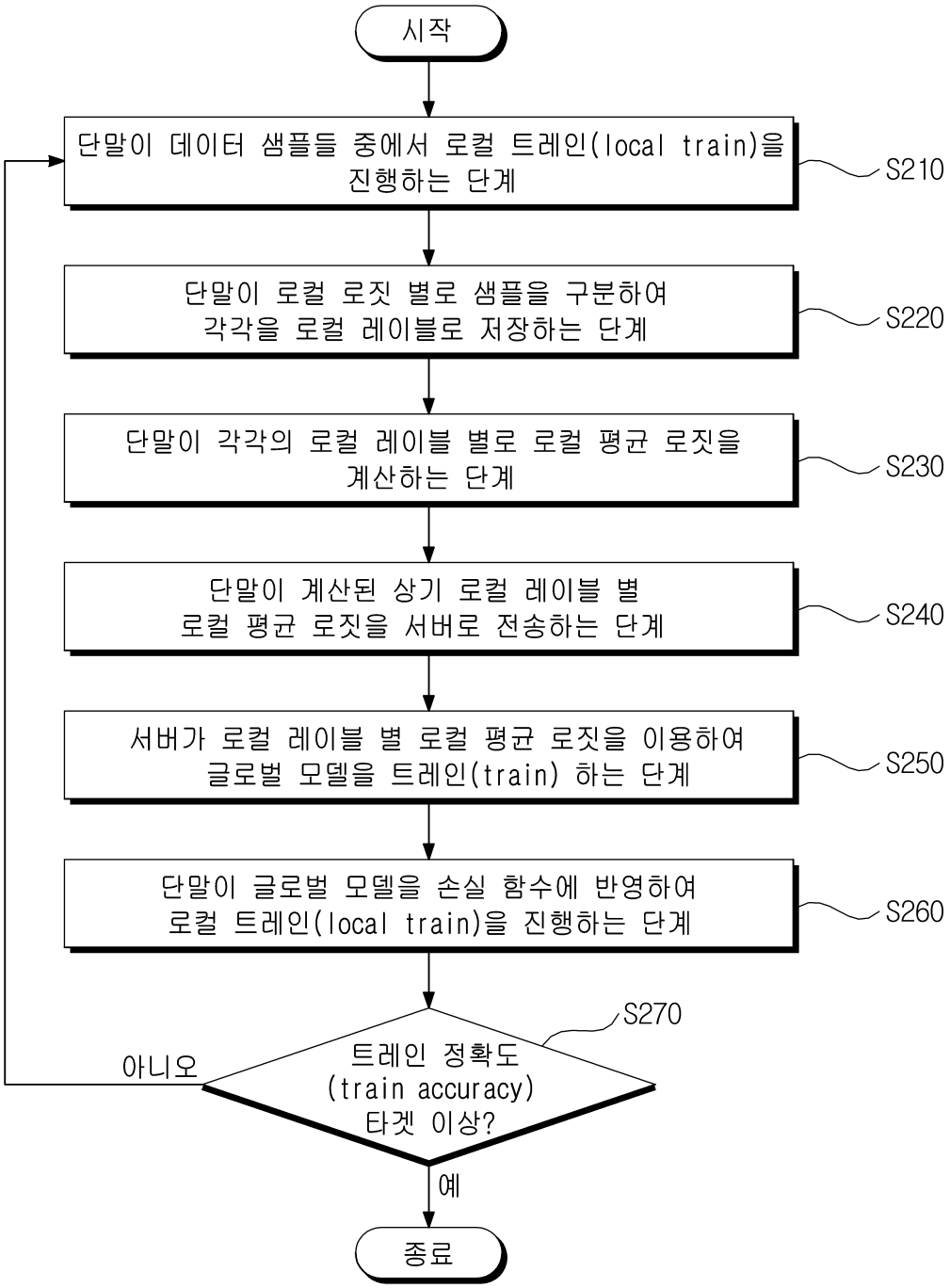
도면6


FL
FD
MixFLD (N_s=100, λ=0.1)
MixFLD (N_s=10, λ=0.1)
Mix2FLD (N_s=100, λ=0.1)
Mix2FLD (N_s=10, λ=0.1)
Test Accuracy
Time (sec)
(a) Asymmetric channels, IID dataset
(b) Symmetric channels, IID dataset.
(c) Asymmetric channels, Non-IID dataset.
(d) Symmetric channels, Non-IID dataset.

도면7


시작
단말이 데이터 샘플들을 수집하여 로컬 평균 로짓을 산정하고, 로컬 평균 로짓을 업링크로 전송하는 단계
S110
단말이 시드 샘플들을 서버의 업링크로 전송하는 단계
S120
서버가 시드 샘플과 로컬 평균 로짓을 기반으로 글로벌 모델의 디스틸레이션(distillation)을 수행하는 단계
S130
서버가 글로벌 모델을 다운 링크로 전송하는 단계
S140
단말이 글로벌 모델을 서버로부터 전달 받아 손실 함수에 반영하여 로컬 트레인(local train)을 진행하는 단계
S150
종료

도면8


시작
단말이 데이터 샘플들 중에서 로컬 트레인(local train)을 진행하는 단계
S210
단말이 로컬 로짓 별로 샘플을 구분하여 각각을 로컬 레이블로 저장하는 단계
S220
단말이 각각의 로컬 레이블 별로 로컬 평균 로짓을 계산하는 단계
S230
단말이 계산된 상기 로컬 레이블 별 로컬 평균 로짓을 서버로 전송하는 단계
S240
서버가 로컬 레이블 별 로컬 평균 로짓을 이용하여 글로벌 모델을 트레인(train) 하는 단계
S250
단말이 글로벌 모델을 손실 함수에 반영하여 로컬 트레인(local train)을 진행하는 단계
S260
S270
트레인 정확도 (train accuracy) 타겟 이상?
아니오
예
종료