계층적 연합학습을 위한 클러스터링 기법

김성하, 한승재
연세대학교
codingeee@yonsei.ac.kr, hanseungjae@yonsei.ac.kr

A Clustering Approach for Hierarchical Federated Learning

Sung-ha Kim, Seung-jae Han
Yonsei Univ.

요 약

본 논문은 계층적 연합학습(Hierarchical Federated Learning) 환경에서 클라이언트들을 효과적으로 클러스터링하여 엣지 서버에 할당하는 방법을 제안한다. HFL은 기존의 연합학습(Federated Learning)과 달리 클라이언트가 생성한 모델 파라미터를 직접 중앙 서버로 전송하지 않고, 엣지 서버에서 먼저 합친뒤에 상위 파라미터 서버로 전달하는 다계층 구조를 가진다. 이러한 구조에서는 엣지 서버 단계에서 발생하는 모델 편향(Model Drift)이 전체 학습 성능에 큰 영향을 미친다. 이에 본 연구에서는 엣지 서버에서 발생하는 모델 편향을 줄이기 위해 클라이언트 클러스터링 기반 할당방법을 제안한다. 실제 실험 결과, 제안 기법은 편향성을 완화하고, 학습 효율 향상에 기여하는 것으로 나타났다.

Ⅰ. 서 론

연합학습(Federated Learning, FL)은 다수의 클라이언트가 가진 로컬 데이터를 중앙 서버로 전송하지 않고 분산된 환경에서 머신러닝 모델을 학습하는 기법이다. 하지만 확장성의 한계로 인한 문제를 해결하기 위해서 엣지 서버를 중간 집계자로 활용하는 계층적 연합학습(Hierarchical Federated Learning, HFL) 구조가 제안되었다 [2]. HFL에서는 클라이언트들이 엣지 서버와 통신하고, 엣지 서버들이 중앙 서버와 통신함으로써 통신 효율성을 크게 향상시킨다. 하지만 클라이언트들의 데이터 분포가 IID(Independent and Identically Distributed)하지 않은 경우, 각 엣지 서버에서 집계된 모델이 서로 다른 방향으로 학습되어 심각한 모델 편향(Model Drift)을 유발할 수 있다. 이러한 편향은 글로벌 모델의 수렴을 방해하고 최종 성능을 저하시키는 주요 원인이 된다[3].

본 논문에서는 HFL 환경에서 엣지 서버의 모델 편향을 완화하기 위한 새로운 클라이언트 클러스터링 기법을 제안한다. 제안하는 기법은 각 클라이언트의 모델 업데이트 방향성을 기반으로, 각 클러스터의 집계된 업데이트가 글로벌 학습 목표와 최대한 유사해지도록 탐욕적(Greedy)으로 클라이언트를 할당한다. 이를 통해 클러스터(엣지 서버) 수준에서부터 모델의 이질성을 줄여 안정적이고 효율적인 학습을 가능하게 한다.

Ⅱ. 본론

본 연구에서 제안하는 클러스터링 기법은 HFL 환경에 참여하는 전체 클라이언트의 집합을 $C=\{c_1,c_2,\cdots,c_N\}$ 과 같이 정의하고, 이들을 K개의 클러스터 $S=\{S_1,S_2,\cdots,S_K\}$ 으로 분할하는 것을 목표로 한다. 각 클라이언트 c_i는 로컬 데이터셋으로 정해진 로컬 에포크(local epoch)만큼 학습을 수행한 뒤 그래디언트 Δi를 계산하며 이를 기반하여 클러스터링을 진행한다. 또한 각 클라이언트에 존재하는 데이터의 갯수는 w_i이다. 클러스터링의 기준이 되는 이상적인 방향성을 정의하기 위해, 모든 클라이언트의 그래디언트를 가중 평균한 가상 글로벌 그래디언트(Virtual Global Gradient) $\Delta_{v\text{-}global}$과 각 클러스터의 평균 그래디언트 Δs_i를 (1)과 같이 정의한다. 중요한 점은, $\Delta_{v\text{-}global}$는 실제 HFL 학습 과정에서 중앙 서버에 적용되는 업데이트가 아니라는 것이다. 실제 글로벌 업데이트는 각 엣지 서버에서 여러 라운드 동안 학습이 진행된 후 집계된 모델 가중치에 의해 이루어진다. 여기서 정의한 $\Delta_{v\text{-}global}$는 단지 클라이언트들을 가장 이상적인 그룹으로 나누기 위해, 클러스터링 단계에서만 사용되는 '이정표' 또는 '참조 벡터(Reference Vector)'의 역할을 수행한다.

알고리즘의 목적 함수는 각 클러스터의 평균 그래디언트 Δs_i의 방향을 이 참조 벡터 $\Delta_{v\text{-}global}$의 방향에 최대한 근사시키는 것이다. 이를 이용한 최종 목적 함수는 (3)과 같다. 즉, 실제 HFL의 글로벌 업데이트는 각 엣지 서버에서 점진적으로 이루어지지만, 학습 시작 전 클러스터링 단계에서 각 엣지 서버의 초기 학습 방향성을 이상적인 '이정표'에 정렬시킴으로써, 장기적으로 엣지 서버 간의 모델이 서로 다른 방향으로 멀어지는 '모델 편향'을 원천적으로 억제할 수 있다. 엣지 서버에서 발생하는 이러한 모델 편향이 전체 학습 성능에 미치는 영향은 여러 연구를 통해 강조되어 왔으며, 특히 FedUC[4]와 같은 연구는 HFL에서 클러스터간 데이터 이질성이 모델의 학습을 저하시키는 주요 원인임을 보인다.

위 목적 함수를 최적화하기 위해, 제안하는 알고리즘은 3단계의 절차를 거친다(그림 1). 첫 번째 초기화 단계에서는 K개의 클라이언트를 무작위로 선택하여 K개의 클러스터를 생성한다. 두 번째 탐욕적 할당 단계에서는 나머지 각 클라이언트를 순회하며, 해당 클라이언트를 추가했을 때 목적 함수 (3)의 값을 가장 크게 감소시키는, 즉 모델 편향을 최소화하는 최적의 클러스터에 할당한다. 마지막으로 반복적 개선 단계에서는 모든 할당이 완료된 후, 각 클라이언트를 다른 클러스터로 이동시켰을 때 전체 목적 함수 값이 개선되는지를 반복적으로 확인하여 최종 클러스터 구성을 최적화한다.

$$\Delta_{v-global} = \frac{\sum_{i=1}^{N} w_i \Delta_i}{\sum_{i=1}^{N} w_i} \qquad \Delta_{S_j} = \frac{\sum_{c_i \in S_j} w_i \Delta_i}{\sum_{c_i \in S_j} w_i} \tag{1}$$

$$\mathrm{minimize}_S \frac{1}{K} \sum_{j=1}^{K} \left(1 - \frac{\Delta_{S_j} \cdot \Delta_{v-global}}{\|\Delta_{S_j}\| \|\Delta_{v-global}\|}\right) \tag{2}$$

Algorithm 1 그래디언트 기반 탐욕적 클러스터링

입력: 클라이언트 집합 $C = \{c_1, \ldots, c_N\}$, 클러스터의 수 K, 개선 반복 횟수 P
출력: 분할된 클라이언트 집합 $S = \{S_1, \ldots, S_K\}$

목적 함수 $\mathcal{L}(S)$를 식 (3)으로 정의한다.
클러스터 $S_1, \ldots, S_K \leftarrow \emptyset$ 로 초기화한다.

▷ *1단계: 시드(Seed) 설정*
K개의 클라이언트를 무작위로 선택하여 C_{seed}를 구성하고, $S_1, \ldots, S_K$를 초기화한다.
$C_{unassigned} \leftarrow C \setminus C_{seed}$

▷ *2단계: 탐욕적 할당*
for $c_i \in C_{unassigned}$ **do**
  $j^* \leftarrow \arg\min_j \mathcal{L}(S$에 c_i를 S_j로 할당했을 경우$)$
  $S_{j^*} \leftarrow S_{j^*} \cup \{c_i\}$
end for

▷ *3단계: 반복적 개선 (P번 순회)*
for $p \leftarrow 1$ to P **do**
  for 각 클라이언트 $c_i \in C$ (무작위 순서) **do**
    $S_{curr} \leftarrow c_i$의 현재 클러스터
    $S_{best} \leftarrow c_i$를 이동시켜 $\mathcal{L}(S)$을 최소화하는 클러스터
    if $\mathcal{L}(S_{best}$으로 이동 후$) < \mathcal{L}(S_{curr})$ **then**
      c_i를 S_{best}로 이동시킨다.
    end if
  end for
end for

return $\{S_1, \ldots, S_K\}$

그림 1. 클러스터링 알고리즘

제안하는 클러스터링 기법의 성능을 검증하기 위해, 이미지 분류 작업에 널리 사용되는 CIFAR-10 데이터셋을 이용한 시뮬레이션을 수행하였다. 총 100개의 클라이언트를 10개의 클러스터(엣지 서버)로 분할하는 계층적 연합학습 환경을 가정하였으며, 모든 클라이언트는 동일한 구조의 CNN(Convolutional Neural Network) 모델을 학습한다. 연합학습 환경의 특징인 데이터 이질성(Non-IID)을 모사하기 위해, 각 클라이언트의 라벨 분포를 디리클레 분포(Dirichlet distribution)를 이용하였고, α 값을 0.01로 설정하였다. 클라이언트에서의 로컬 학습은 배치 사이즈(batch size) 32, 로컬 에포크(local epoch) 2로 설정하여 진행했다. 계층적 구조에 따라, 각 엣지 서버는 자신에게 할당된 클러스터로부터 총 10번의 라운드동안 모델 파라미터를 집계하여 중앙서버로 보내고 이를 반복한다. 또한 알고리즘에서 반복적 개선은 총 5번 이루어진다.

탐욕적 클러스터링 기법(Greedy-Clustering)은 다음과 같은 두 가지 방식과 성능을 비교하였다. 첫째, K-Means 클러스터링은 클라이언트들의 그래디언트 벡터를 유클리드 거리 기반으로 그룹화하는 일반적인 방식이다. 둘째, 라운드-로빈 클러스터링(RR-Clustering)은 클러스터 내 모델 편향의 정도를 의도적으로 조절하여, 제안하는 탐욕적 방식과 일반적인 K-Means 방식의 중간 수준의 편향성을 갖도록 설계한 특수한 방식이다. 이는 클러스터의 편향 수준과 최종 학습 성능 간의 명확한 상관관계를 보이기 위함이다. 이 방식의 동작 원리는 다음과 같다. 먼저, K-Means를 수행하여 그래디언트가 유사한 K개의 동질적인 초기 그룹을 형성한다. 그 후, 각 초기 그룹에 속한 클라이언트들을 최종 클러스터 1번부터 K번까지 순차적으로 할당하는 라운드-로빈 방식을 적용한다. 이때 핵심 파라미터인 레벨(L)은, 하나의 최종 클러스터에 동일한 초기 그룹 소속 클라이언트가 최대 몇 개까지 연속으로 할당될 수 있는지를 결정한다. 예를 들어 L=2인 경우, K-Means의 1번 그룹 클라이언트 중 2개를 최종 클러스터 1번에 할당하고, 다음 2개를 최종 클러스터 2번에 할당하는 과정을 반복한다. L 값이 작을수록 동질 그룹이 여러 클러스터에 널리 분산되어 클러스터 모델간 편향이 감소한다. 본 실험에서는 L=2, 5 두 가지 경우를 통해 편향 수준에 따른 성능 변화를 관찰하고자 하였다.

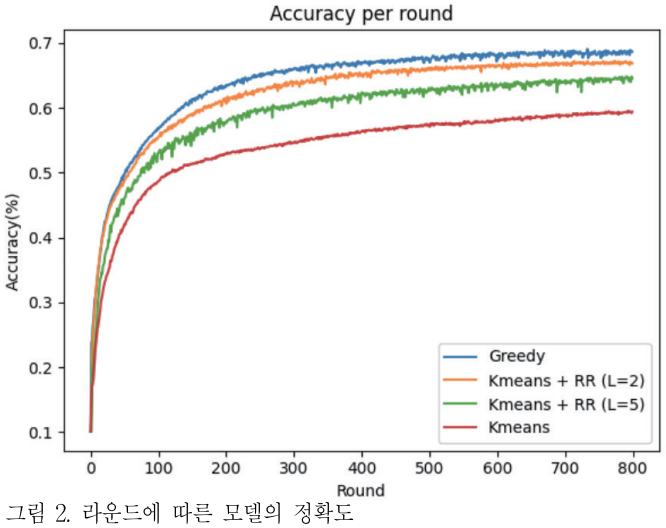

그림 2. 라운드에 따른 모델의 정확도

	Variance	Final accuracy	Round (63%)
Greedy	0.0053	69.08%	242
RR(L=2)	0.0344	67.30%	260
RR(L=5)	0.09826	66.14%	486
Kmeans	0.2684	63.36%	2024

표 1. 클러스터링에 따른 Variance, Final accuracy, Round(63%)

그림2은 클러스터링 알고리즘에 따른 모델의 정확도의 양상을 800라운드까지 돌려보면서 관찰한 결과이고, 표1은 해당 클러스터링 알고리즘을 수렴할때까지 돌려본 결과. 각각의 클러스터 레벨에서의 그래디언트들의 분산(Variance), 최종정확도(Final Accuracy), 그리고 63%를 달성하기까지 걸린 라운드수를 나타낸다. Variance와 학습양상을 미루어볼때, 클러스터레벨에서의 모델의 편향성이 학습에 영향을 많이 준다는것을 확인할 수 있었고, 제안하는 탐욕적인 알고리즘이 클러스터 레벨에서의 그래디언트를 가장 최소화시키며, 이에 따라 모델 파라미터를 합치는 과정에서 최적의 파라미터를 찾는것이 용이해져 더 빨리 학습하고, 최종 정확도 또한 높아지는것을 볼 수 있다.

Ⅲ. 결론

본 논문에서는 클러스터 레벨에서의 모델 편향성을 줄이는 클러스터링 기법을 제시하고, 다른 비교군들과 비교하여 해당 알고리즘이 실제로 이 편향성을 줄이며, 학습 효율을 높일 수 있다는것을 보였다.

참 고 문 헌

[1] H. B. McMahan, et al., "Communication-efficient learning of deep networks from decentralized data," in Proc. AISTATS, pp. 1273-1282, 2017.

[2] L. Liu, et al., "Client-edge-cloud hierarchical federated learning," in Proc. IEEE ICC, pp. 1-6, 2020.

[3] Y. Zhao, et al., "Federated learning with non-IID data," arXiv preprint arXiv:1806.00582, 2018.

[4] Q. Ma, et al., "FedUC: A unified clustering approach for hierarchical federated learning," IEEE Trans. Mobile Comput., vol. 23, no. 10, pp. 9737-9756, Oct. 2024.