# 추천 시스템의 데이터 이동 최적화를 위한 임베딩 캐시 모델의 성능 분석

문현우 ○ 구건재

고려대학교 컴퓨터학과

ty5623@korea.ac.kr, gunjaekoo@korea.ac.kr

# Performance Analysis of Embedding Cache Models for Optimizing Data Movement in Recommendation Systems

Hyunwoo Moon○ Gunjae Koo

Department of Computer Science and Engineering, Korea University

**요 약**

딥러닝 기반 추천 시스템은 심층신경망(DNN) 레이어와 임베딩 레이어로 이루어져 있다. 임베딩 레이어는 범주형 입력값을 이에 대응하는 임베딩 벡터로 변환하는 연산을 수행하며, 임베딩 레이어의 임베딩 테이블 크기는 점진적으로 증가하여 최근에는 수 TB까지 증가하고 있다. 임베딩 레이어는 병렬프로세서를 활용한 성능 가속에 적합한 특성을 가지고 있으나 메모리 용량의 한계가 주요한 병목점이 되고 있다. 이를 해결하기 위한 방법으로 임베딩 테이블 접근 패턴의 지역성을 활용한 임베딩 캐시 모델을 적용할 수 있다. 임베딩 캐시 모델은 GPU, CPU, SSD로 이루어진 계층 구조를 활용하고 있다. 본 연구에서는 임베딩 캐시 모델의 계층별 성능 분석을 진행하였다. 이를 통해 GPU 메모리에 적재되는 임베딩 벡터 캐시 비율 증가가 특정 비율 이상부터는 성능 향상에 미치는 영향이 제한적임을 확인하였다. 또한 이번 연구에서 SSD와 GPU간 데이터 전달이 임베딩 캐시 모델의 주요 성능 병목점이 될 수 있음을 밝혀냈다.

## 1. 서 론

딥러닝 기반 추천 시스템은 영화, 광고, SNS 등 인터넷 기반 기업에서 활발히 활용되고 있는 어플리케이션이다. 딥러닝 기반 추천 시스템은 숫자형 데이터를 처리하는 심층 신경망(DNN) 레이어와 범주형 데이터를 처리하는 임베딩 레이어로 이루어져 있다. 이 중 임베딩에 사용되는 임베딩 벡터들은 각 항목에 매핑되는 테이블 형태로 저장되는데 추천 시스템의 정확도 향상을 위해 연산에 사용하는 임베딩 데이터가 증가함에 따라 이를 저장하는 임베딩 테이블의 크기는 최근 수 TB에 이르고 있다 [3]. 임베딩 테이블의 높은 메모리 사용량은 메모리 용량 확장이 불가능한 GPU에서 추천 시스템을 실행하는데 주요한 장벽이 되고 있다. 이를 극복하기 위해 임베딩 테이블 접근 패턴의 지역성을 활용한 임베딩 캐시 모델이 제안되고 있다. 임베딩 캐시 모델은 GPU, CPU, 스토리지로 이루어진 계층적 구조를 가지고 있다 [1]. 이는 전체 임베딩 테이블을 시스템 메모리에 적재하는 대신 접근 빈도가 높은 임베딩 벡터만을 GPU 메모리와 CPU 메모리에 계층적으로 적재하여 시스템의 메모리 용량이 부족한 환경에서도 추천 시스템 실행이 가능하게 하는 장점이 있다.

본 연구에서는 임베딩 캐시 모델의 성능 분석을 GPU 메모리 계층과 CPU 메모리 계층으로 나누어 진행하였다. 이를 통해 GPU 메모리에 캐시되는 임베딩 벡터의 비율을 증가시키는 것이 일정 비율 이상부터는 성능 향상에 미치는 영향이 줄어드는 것을 확인하였다. 또한 SSD와 GPU 간 데이터 전달 과정이 임베딩 캐시 모델의 주요 병목점이 될 수 있음을 밝혀내었다.

* 이 성과는 2023년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. 2021R1C101272)

## 2. 배 경

딥러닝 기반의 추천 시스템은 사용자의 과거 기록을 바탕으로 여러 다른 사용자가 상품을 선호할 확률을 계산한다. 추천 시스템의 입력값은 숫자형 데이터와 범주형 데이터로 이루어져 있다. 추천 시스템은 두 유형의 데이터를 각각 DNN 레이어와 임베딩 레이어에서 처리한 뒤 그 결과값을 합하여 단일 벡터를 생성한다. 이후 생성된 벡터를 Top-DNN 레이어의 입력값으로 넣어 최종적인 확률을 계산한다 [4]. 이 중 임베딩 레이어는 범주형 입력값을 이에 대응하는 임베딩 벡터로 변환하는 임베딩 과정을 수행한다. 임베딩 벡터란 범주형 데이터의 특징적 요소를 지정된 차원의 공간으로 투영시킨 벡터이다.

임베딩 테이블은 추천 시스템 연산에 필요한 범주형 데이터를 행으로 각 범주형 데이터에 연결된 특성 아이템의 매핑 정보를 열로 가지고 있다. 범주형 데이터가 입력값으로 들어오면 임베딩 레이어는 범주형 입력값에 대응되는 임베딩 벡터를 임베딩 테이블에서 추출하는 임베딩 룩업(lookup) 연산을 수행한다. 임베딩 룩업 연산은 테이블 내에 많은 양의 임베딩 벡터 중 소수의 벡터를 추출하는 연산이기 때문에 불규칙하고 희소한 메모리 접근 특성을 가지고 있다. 따라서 임베딩 룩업 작업은 높은 메모리 대역폭이 요구된다. 기존의 추천 시스템 실행 환경은 메모리 용량 확장이 용이한 CPU 메모리에서 임베딩 레이어를 처리하는 방식을 주로 사용한다. 그러나 임베딩 룩업 작업은 높은 메모리 대역폭을 필요로 하기 때문에 CPU 메모리의 낮은 대역폭은 임베딩 레이어의 성능 저하를

일으키는 주요한 요소로 작용하고 있다 [5].

GPU는 딥러닝 연산을 실행하는 데 유리한 높은 연산력과 높은 대역폭의 메모리를 탑재하고 있어 추천 시스템을 효율적으로 실행하는데 적합한 기기이다. 그러나 임베딩 테이블의 높은 메모리 요구량은 GPU의 제한된 메모리 공간에서 추천 시스템을 실행하는데 제약이 되고 있다. 이를 해결하기 위해 임베딩 레이어의 지역성을 활용한 임베딩 캐시 모델이 제안되었다. 임베딩 캐시 모델은 메모리 용량이 적은 GPU에서도 큰 규모의 추천 시스템을 실행시킬 수 있다는 장점이 있어 많은 연구가 진행되고 있다.

## 3. 분석 과정 및 결과

본 연구에서 임베딩 캐시 모델의 성능 분석에 이용한 실험환경은 표 1과 같다. 실험에 사용된 추천 시스템은 Meta의 DLRM이며 실험에 사용된 임베딩 캐시 모델 또한 DLRM을 기반으로 구현하였다 [4].

**표 1. 실험 환경**

| Dataset | Criteo AI Labs Ad Kaggle Dataset |
|---|---|
| CPU | 11th Gen Intel i5-11400 (12) @ 4.400GHz |
| Memory | 32GB DDR4-3200 |
| GPU | NVIDIA T1000 8GB |
| Storage | SK hynix P31 1TB PCIe 3.0 4-lane (32GT/s) |

### 3.1 임베딩 레이어의 지역성 분석

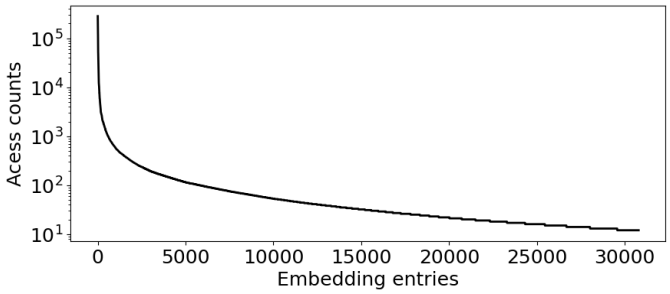


**그림 1. 임베딩 엔트리 접근 패턴**

임베딩 캐시 모델은 전체 임베딩 테이블을 GPU 메모리에 적재하지 않고 특정 임베딩 벡터만을 GPU 메모리에 적재하는 방식을 사용한다. 이는 임베딩 테이블 접근 패턴이 일정 수준의 지역성을 가지는 것을 기반으로 한다 [1].

**그림 1**은 임베딩 테이블 엔트리의 접근 패턴을 보여주는 그래프이다. 전체 테이블 엔트리를 접근 횟수를 기준으로 정렬한 뒤 그 중 접근 횟수가 가장 많은 0.1%의 엔트리들의 접근 패턴을 나타내고 있다. 이 그래프는 매우 소수의 임베딩 테이블 엔트리가 대다수의 접근 횟수를 차지하는 power-law 분포 형태를 보여주고 있다. 또한 이 실험에서 0.1%의 임베딩 엔트리가 전체 테이블 접근 횟수 중 89%를 차지하는 것을 확인하였다. 이렇게 접근 횟수가 많은 엔트리를 hot-embedding 엔트리라고 하며 이 엔트리를 메모리에 적재하여 모델의 cache hit-rate를 높일 수 있다.

### 3.2 GPU 메모리 계층 분석

임베딩 캐시 모델은 전체 모델을 메모리에 적재하는 대신 임베딩 테이블의 엔트리 중 hot-embedding 엔트리를 메모리에 캐싱하는 방식을 사용한다. 여러 선행 연구에서 임베딩 캐시 모델은 GPU, CPU, 스토리지를 구성 요소로 하는 계층적 캐시 구조 형태를 보이고 있다. 이 중 스토리지는 모든 임베딩 테이블을 저장하고 있으며 GPU와 CPU는 메모리 용량에 따라 일정 비율의 임베딩 벡터를 캐싱하는 방식이 사용되고 있다. 본 연구에서 임베딩 레이어는 GPU에서 수행되므로 GPU 메모리가 가장 높은 계층을 가지는 메모리 계층구조를 갖는다.

여러 선행 연구에서 임베딩 캐시 모델의 hit-rate를 높이기 위해 CPU 메모리에서 GPU로 임베딩 벡터를 prefeching 하는 방식 또는 GPU 하드웨어에 최적화된 임베딩 캐시 구조 등을 제안하였다. 반면 본 연구에서는 임베딩 캐시 모델의 GPU와 CPU의 메모리 사용 패턴을 분석하기 위한 naïve한 임베딩 캐시 모델을 구현하고 그 특징을 분석하였다. 실험에 사용된 모델은 임베딩 레이어 연산만을 수행한다. 임베딩 캐시 모델의 캐시는 16-way set-associative 형태를 가지고 있으며 교체 알고리즘은 선행 연구와 마찬가지로 Least Recently Used(LRU) 방식을 사용하였다.

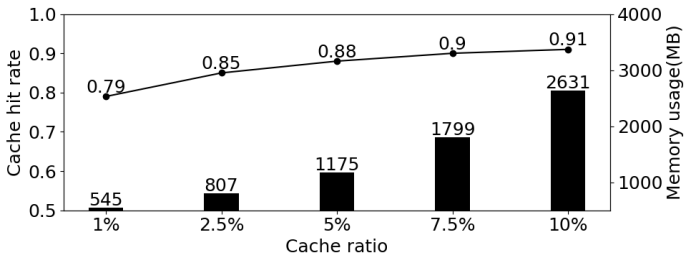


**그림 2.GPU 캐시 hit-rate 및 메모리 사용량**

**그림 2는** GPU 메모리에 캐시된 임베딩 테이블의 비율을 1%에서 10%까지 증가시키는 과정에서 GPU 메모리의 캐시 hit-rate 변화를 나타낸 그래프이다. Cache ratio는 전체 임베딩 엔트리 중 GPU 메모리에 캐시된 임베딩 엔트리의 비율을 의미한다. 이 실험에서 GPU 메모리의 cache ratio가 증가함에 따라 캐시 hit-rate도 점진적으로 증가하였으나 cache ratio가 일정 수준 이상인 시점부터는 hit-rate 향상폭이 크게 줄어 드는 것을 확인하였다. 이에 비해 임베딩 캐시의 메모리 사용량은 cache ratio가 증가함에 따라 큰 폭으로 증가하였다. 예를 들어 cache ratio가 5%에서 10%로 증가하는 동안 hit-rate는 0.03% 증가한데 비해 메모리 사용량은 2배가량 증가하였다. 이러한 결과는 GPU 메모리에서의 임베딩 테이블 캐시 비율 증가가 GPU 메모리의 사용량 증가 대비 일정 수준 이상부터는 큰 영향을 끼치지 못하고 있음을 시사한다.

### 3.4 CPU 메모리 계층 분석

임베딩 캐시 모델을 구현한 선행 연구는 CPU 메모리를 2차 캐시로 사용하는 구조를 가지고 있으며 CPU 메모리에서

cache-miss가 발생할 경우 스토리지에서 miss가 발생한 임베딩 벡터를 가져오는 구조를 가지고 있다 [1]. CPU 메모리는 GPU 메모리에 비해 확장이 용이하나 임베딩 테이블 사이즈가 지속적으로 증가함에 따라 CPU 메모리에 임베딩 테이블을 모두 적재하는 것이 어려워지고 있다. 그렇기 때문에, 본 연구에서는 CPU 메모리 계층에서의 miss-penalty가 임베딩 캐시 모델 성능에 미치는 영향을 분석하였다.

**그림 3**의 파란색 그래프는 CPU 메모리의 cache-ratio를 75%에서 10%로 감소시키는 과정에서 CPU 메모리의 hit-rate 변화를 나타낸 그래프이다. 이 실험에서 CPU 메모리의 hit-rate는 cache ratio가 75% 일 때도 99%로 높은 수준을 유지하는 것을 확인하였다. 이러한 결과는 일부 임베딩 엔트리가 사용 빈도에 비해 CPU 메모리를 불필요하게 점유하고 있음을 시사한다. **그림 3**의 붉은색 그래프는 cache ratio가 100%인 상황 대비 모델의 실행 시간 저하 정도를 나타낸 그래프이다. 이 실험에서 모델의 실행 시간은 hit-rate가 99%인 상황에서도 3배가량 느려지는 것을 확인하였다.

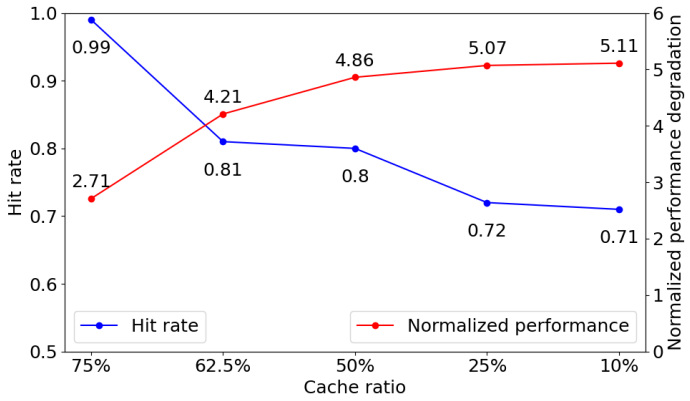


**그림 3. CPU 캐시 hit-rate 및 성능 비교**

**그림 4**는 같은 실험에서 임베딩 캐시 모델의 실행 시간 중 SSD에서 GPU로 데이터를 전달하는 데 소요된 시간이 차지하는 비율을 보여준다. Cache ratio가 75%인 환경에서 전체 실행 시간 중 SSD-GPU 간 데이터 이동 시간이 차지하는 비율은 67%에 달했으며 cache ratio가 감소함에 따라 그 비율은 최대 86%까지 증가하는 것을 확인하였다. 이러한 결과는 CPU 메모리에서의 miss-penalty로 인해 발생하는 SSD에서 GPU로 데이터를 전달하는 과정이 임베딩 캐시 모델의 주요한 병목점이 될 수 있음을 시사한다.

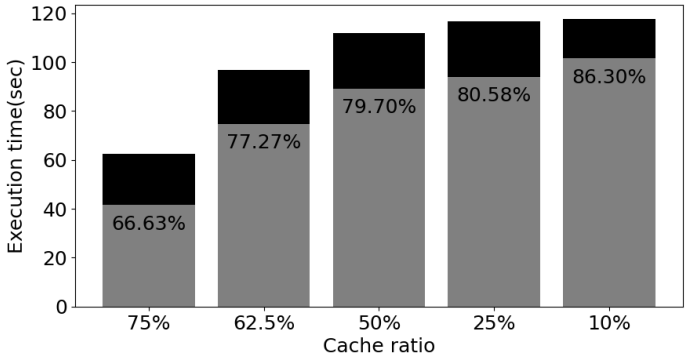


**그림 4. 전체 실행 시간 중 SSD-GPU 데이터 이동 시간 비율**

## 4. 결 론

본 연구에서는 딥러닝 추천 모델의 임베딩 레이어를 효율적으로 실행하기 위한 임베딩 캐시 모델의 성능 분석을 진행하였다. 이를 위해 임베딩 캐시 모델을 구현하여 GPU 메모리 계층과 CPU 메모리 계층에서의 성능 분석을 진행하였다. 이를 통해 GPU 메모리 계층에서는 cache ratio를 증가시키는 것이 일정 비율 이상부터는 메모리 사용량 증가 대비 성능 향상에 미치는 영향이 줄어드는 것을 밝혀냈다. 이어서 CPU 메모리 계층에서는 상당한 양의 임베딩 벡터가 접근 빈도에 비해 CPU 메모리를 불필요하게 점유하고 있음을 확인하였다. 이에 반해 CPU 메모리 계층에서의 캐시 미스로 인해 발생하는 SSD에서 GPU로의 데이터 이동이 임베딩 캐시 모델의 성능의 주요한 병목점으로 작용할 수 있음을 밝혀냈다.

## 5. 참 고 문 헌

[1] Y. Wei *et al.*, "A GPU-specialized Inference Parameter Server for Large-Scale Deep Recommendation Models," in *RecSys 2022 - Proceedings of the 16th ACM Conference on Recommender Systems*, Association for Computing Machinery, Inc, Sep. 2022, pp. 408-419. doi: 10.1145/3523227.3546765.

[2] K. Balasubramanian, A. Alshabanah, J. D. Choe, and M. Annavaram, "CDLRM: Look ahead caching for scalable training of recommendation models," in *RecSys 2021 - 15th ACM Conference on Recommender Systems*, Association for Computing Machinery, Inc, Sep. 2021, pp. 263-272. doi: 10.1145/3460231.3474246.

[3] D. Mudigere *et al.*, "Sofware-Hardware Co-design for Fast and Scalable Training of Deep Learning Recommendation Models," in *Proceedings - International Symposium on Computer Architecture*, Institute of Electrical and Electronics Engineers Inc., Jun. 2022, pp. 993-1011. doi: 10.1145/3470496.3533727.

[4] NAUMOV, Maxim, et al. Deep learning recommendation model for personalization and recommendation systems. *arXiv preprint arXiv:1906.00091*, 2019.

[5] J. Fang *et al.*, "A Frequency-aware Software Cache for Large Recommendation System Embeddings," Aug. 2022, [Online]. Available: http://arxiv.org/abs/2208.05321