LLM-based Recommendation : From Basics to Beyond

Haeyoon Koo (구해윤)

Supervisor: Byungkook Oh

Graph & Language Intelligence Laboratory
Department of Computer Science and Engineering
Konkuk University

2025.07.24

CONTENTS

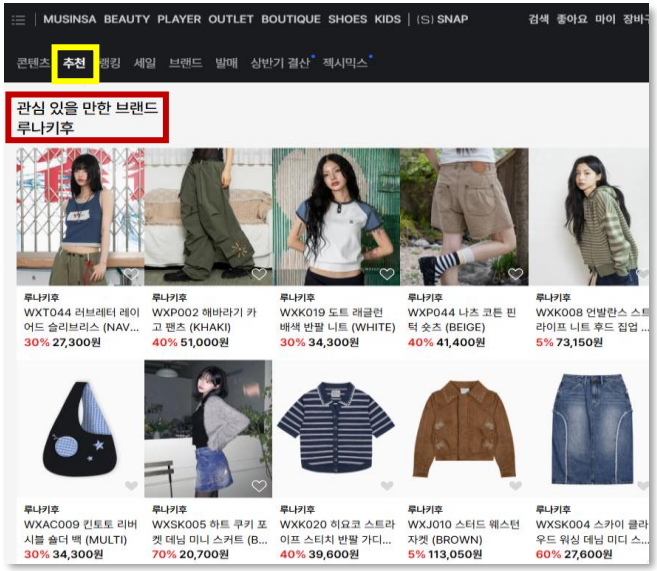
1. Background
   - 추천 시스템
   - LLM
   - LLM 기반 추천 시스템의 등장 배경
2. LLM-based Recommendation
3. 모델 비교 (LLM vs. w/o LLM)
4. 연구 방향

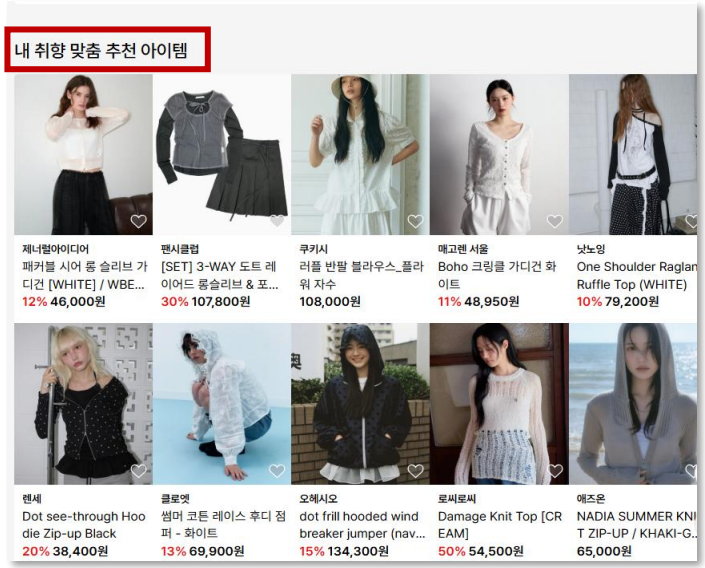
Background

추천 시스템이란?

- 사용자의 선호나 과거 행동 데이터를 기반으로, 개인 맞춤형 아이템을 추천하는 시스템
  - ✓ 유튜브, 뉴스 피드, 검색, 금융 상품 등

- 추천 시스템의 주요 장점 (예시: 무신사)
  - ✓ 사용자 관점 → 원하는 상품을 찾기 위한 시간 절감, 새로운 상품에 대한 접근 용이
  - ✓ 서비스 제공자 관점 → 사용자의 관심사에 맞춘 고품질 추천으로 매출 증대

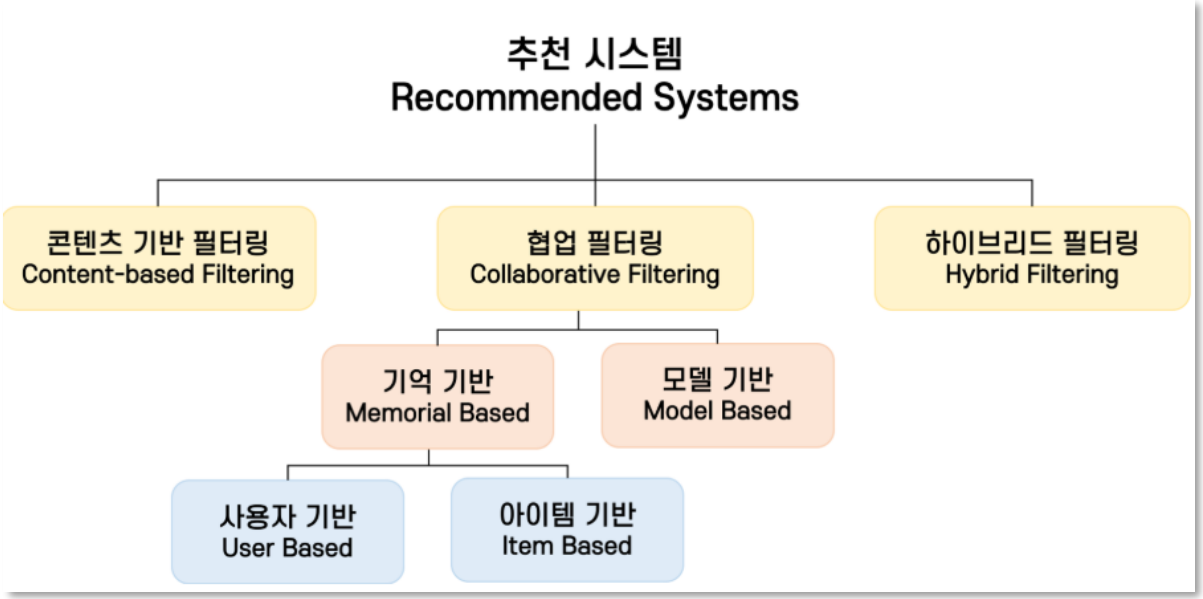




Background

추천 시스템의 방법론

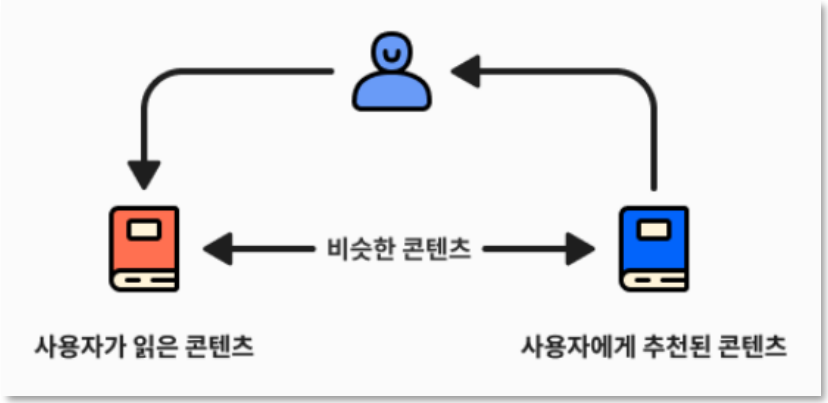
- Content-based Filtering(CBF) ☞ "네가 이전에 좋아한 컨텐츠와 비슷한 걸 추천해줄게!"
- Collaborative Filtering(CF) ☞ "너랑 비슷한 취향을 가진 사람들이 좋아한 걸 추천해줄게!"
- Knowledge-based filtering(KB)
- Hybrid Filtering



Background

추천 시스템의 방법론

- Content-based Filtering(CBF) ☞ "네가 이전에 좋아한 컨텐츠와 비슷한 걸 추천해줄게!"
  - ✓ 아이템의 세부 정보를 바탕으로 사용자가 과거에 소비한 컨텐츠와 유사한 컨텐츠를 추천
  - ✓ 즉, 사용자가 이전에 높게 평가했던 컨텐츠(빨간색)와 가장 유사한 컨텐츠(파란색)를 추천함!
    - Ex) 내가 구매한 스트라이프 니트와 유사한 색상·소재 · 핏의 옷을 추천



  - ✓ 'content'를 추천 시스템이 이해할 수 있는 형태로 전처리 하는 과정이 필요
    - 이 과정에서 데이터를 벡터화하고, 이를 embedding 형태로 변환하여
    - 시스템이 의미를 학습할 수 있도록 해야 함

Background

추천 시스템의 방법론

- Content-based Filtering(CBF) ☞ "네가 이전에 좋아한 컨텐츠와 비슷한 걸 추천해줄게!"
  - ✓ 'content'를 추천 시스템이 이해할 수 있는 형태로 전처리 하는 과정이 필요
    - 이 과정에서 데이터를 벡터화하고, 이를 embedding 형태로 변환하여
    - 시스템이 의미를 학습할 수 있도록 해야 함
  - ✓ Content의 구분 → 각 유형별로 벡터화에 적합한 방법이 다름
    - 이미지
      - Image to vector
        - » 딥러닝 기반 모델(CNN, ResNet, VGG 등)을 활용해 이미지를 고차원 임베딩 벡터로 변환
    - 텍스트
      - Text to vector
        - » 자연어 처리 기법(TF-IDF, Word2Vec 등)을 사용해 텍스트 데이터를 임베딩 형태로 변환
        - » 이를 통해 단어 간의 의미 & 문맥을 벡터로 학습함

Background

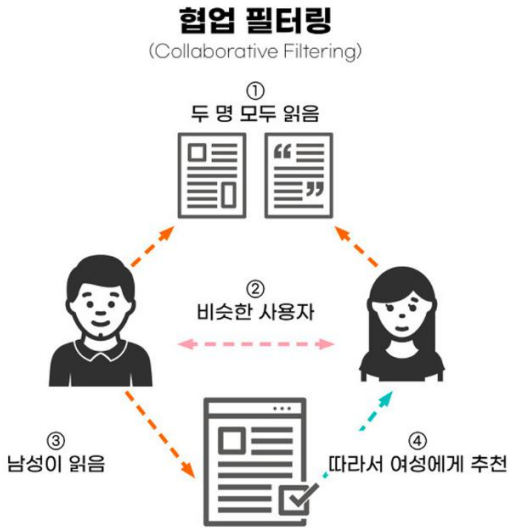
추천 시스템의 방법론

- Content-based Filtering(CBF) ☞ "네가 이전에 좋아한 컨텐츠와 비슷한 걸 추천해줄게!"
  - ✓ 장점
    - 신규 사용자에게도 추천이 가능
      - 다른 사용자의 데이터에 의존 x
      - 사용자가 제공한 기본 정보를 바탕으로 콘텐츠를 추천
    - 추천 근거를 제시 가능
      - 벡터화 된 컨텐츠 간의 유사성을 계산하여 추천하므로,
      - 사용자에게 명확한 추천의 근거를 제공할 수 있음
  - ✓ 단점
    - 초기 데이터 부족 문제
      - 사용자가 과거에 흥미를 보인 컨텐츠 정보가 없을 경우, 적절한 추천을 생성하기 어려움
    - 유사한 컨텐츠만 추천
      - 컨텐츠 간 유사성에만 초점을 맞추므로, 사용자가 이미 알고 있는 컨텐츠와 비슷한 것만 반복적으로 추천할 가능성이 있음

Background

추천 시스템의 방법론

- Collaborative Filtering(CF) ☞ "너랑 비슷한 취향을 가진 사람들이 좋아한 걸 추천해줄게!"
  - ✓ 다수의 사용자 행동 데이터를 바탕으로 아이템을 추천
  - ✓ 다음과 같은 가정을 사용
    - "다른 사용자로부터 얻은 취향 정보를 토대로, 나와 비슷한 취향을 가진 사람들이 선호하는 컨텐츠를 나도 좋아할 가능성이 크다" → 일종의 집단지성을 활용
  - ✓ Ex) A 브랜드 바지와 B 브랜드 셔츠를 구매했을 때, 나와 비슷한 구매 패턴을 가진 사람들이 자주 함께 구매한 신발을 추천



Background

추천 시스템의 방법론

- Collaborative Filtering(CF) ☞ "너랑 비슷한 취향을 가진 사람들이 좋아한 걸 추천해줄게!"
  - ✓ CF의 모델
    - Memory-based
      - 유사도를 바탕으로 동작하는 가장 전통적인 접근 방식
        - » 사용자 기반 CF (사용자 간의 유사도를 기반)
        - » 아이템 기반 CF (아이템 간의 유사도를 기반)
    - Model-based
      - 머신러닝을 통해 사용자 or 아이템의 숨겨진 feature 값을 계산하고 추천하는 방식
        - » Latent Factor (잠재 요인): 사용자가 평가하지 않은 항목들에 대한 평점까지 예측하여 추천하는 방법
        - » Matrix Factorization(MF, 행렬 분해): 사용자가 한 번도 보지 않은 컨텐츠를 추천 가능
  - ✓ 장점
    - 사용자가 이전에 보지 않았던 새로운 컨텐츠도 추천 가능
    - 컨텐츠의 세부 정보 없이도 사용자 행동 데이터만으로 추천이 가능
  - ✓ 단점
    - Cold start (신규 사용자 / 아이템에 대한 데이터가 부족할 때, 추천의 정확도가 떨어지는 문제)
    - Data sparsity
    - 계산 복잡성

Background

추천 시스템의 방법론

- Hybrid Filtering
  - ✓ 다양한 추천 시스템 방법론을 결합
  - ✓ (대표) Content-based Filtering + Collaborative Filtering
    - CF는 새로운 아이템에 대한 추천이 부족 → 이에 CBF 기법이 cold start 문제를 완화해줌
- Knowledge-based Filtering
  - ✓ 사용자가 명확히 제시한 조건이나 선호 정보, 도메인 지식(규칙, 제약조건 등)을 기반으로
  - ✓ 조건을 만족하는 아이템을 추론 및 추천
  - ✓ 사용자 행동/선호 데이터가 아예 없거나 매우 부족할 때 사용
  - ✓ 정확한 요구 충족이 중요한 도메인 (여행, 의료, 보험 등)

Background

Large Language Model(LLM)

- Language Model(LM)
  - ✓ 주어진 문맥에서 가장 적절한 단어나 문장을 예측하는 모델
  - ✓ 기계가 언어를 학습하는 방식 중 하나로, 텍스트에서 단어 or 문장 시퀀스의 확률을 계산하여 자연스러운 문장을 생성하는 데 활용

  - ✓ 대표 방식
    - 순방향 예측 모델: 이전 단어들을 기반으로 다음 단어의 확률을 예측하는 방식 (ex. GPT 시리즈)
    - 양방향 예측 모델: 문장의 앞 뒤 정보를 모두 활용하여 특정 단어를 예측하는 방식 (ex. BERT)

  - ✓ LM의 발전 과정
    - 통계적 언어 모델 (SLM, Statistical Language Model)
    - 신경 언어 모델 (NLM, Neural Language Model)
    - 사전 학습된 언어 모델 (PLM, Pre-trained Language Model)
    - 대형 언어 모델(LLM, Large Language Model)
      - PLM을 확장한 형태

Background

Large Language Model(LLM)

- Large Language Model(LLM)
  - ✓ Transformer (구글, 2017년) 모델을 기반으로 방대한 데이터를 학습한 대규모 NLP 모델
    - 왜 'Large' 인가?
      - 기존의 언어 모델보다 훨씬 더 많은 데이터를 통해 수십억 ~ 수천억 개의 파라미터를 활용
    - 특징
      - 방대한 양의 데이터를 학습했을 때, 다양한 NLP task들을 하나의 모델로 수행할 수 있음

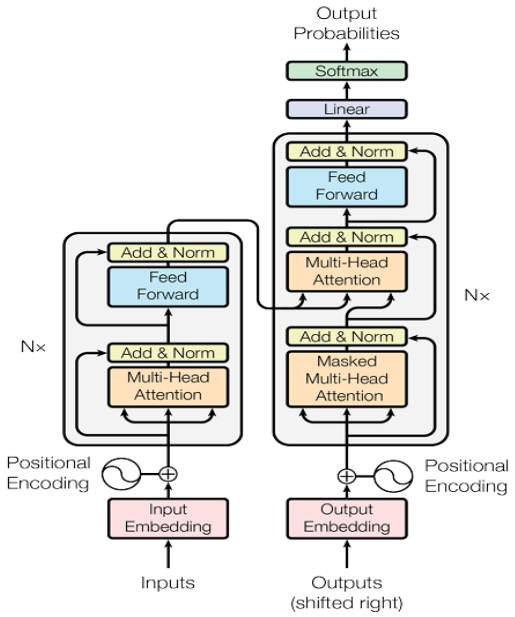
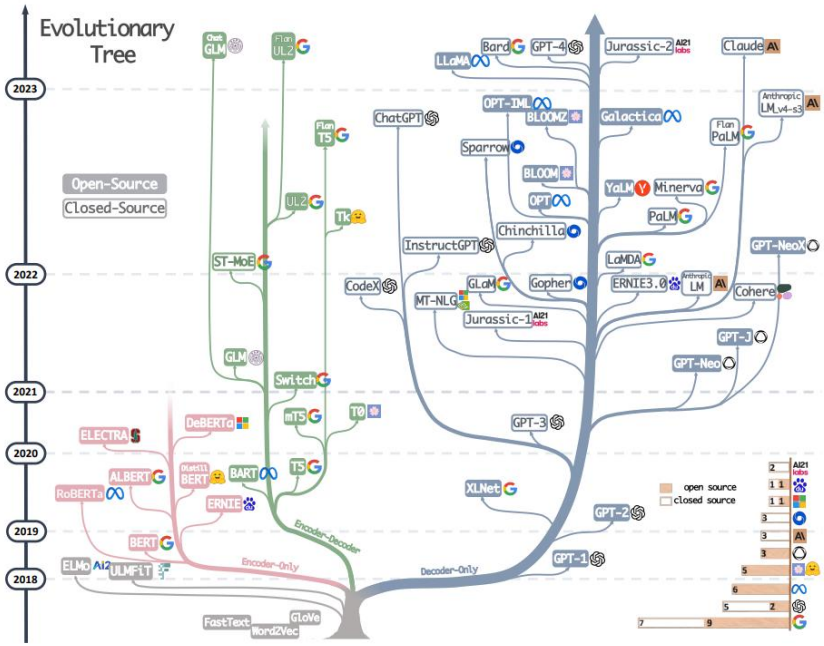


Figure 1: The Transformer - model architecture.



Background

Large Language Model(LLM)

- **구조별 LLM 모델 특징**
  - ✓ **Encoder-Decoder / Encoder-only (BERT-style)**
    - 문장에서 일부 단어를 masking한 후, 주변 문맥을 기반으로 이를 예측하도록 training
    - 문장의 양방향 문맥을 고려하여 학습이 가능
    - 대표 모델
      - BERT, RoBERTa, ALBERT, T5, GLM 등
  - ✓ **Decoder-only (GPT-style)** 🔥 🔥
    - Transformer의 decoder 부분만을 활용하여, 앞의 단어들을 기반으로 다음 단어를 예측하는 방식으로 학습
    - 즉, 문장의 흐름을 따라가며 점진적으로 새로운 단어를 생성
    - 대표 모델
      - GPT-3, GPT-4, PaLM, LLaMA 등

Background

LLM 기반 추천 시스템의 등장

- **전통적인 추천 시스템의 한계**
  - ✓ 대부분 ID 기반(item ID, user ID) embedding을 중심으로 학습
  - ✓ 새로운 아이템이나 사용자가 등장할 시, cold-start 문제가 발생
  - ✓ 상호작용 로그가 부족한 상황에서 일반화 능력이 떨어짐

- **텍스트 정보 활용에 대한 수요 증가**
  - ✓ 리뷰, 설명, 타이들 등의 텍스트 정보는 사용자 취향과 아이템 특성을 잘 담고 있음
  - ✓ But, 기존 모델은 텍스트를 잘 활용하지 못함

- **LLM의 자연어 처리 능력에 주목**
  - ✓ GPT 계열의 LLM은 다양한 텍스트 기반 질의 응답, 요약, 생성 task에서 뛰어난 성능을 보임
  - ✓ 텍스트 feature의 고품질 표현을 추출 가능
  - ✓ LLM이 가지고 있는 방대한 지식을 활용 가능

2. LLM-based recommendation

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- LLM 기반 추천 시스템의 taxonomy

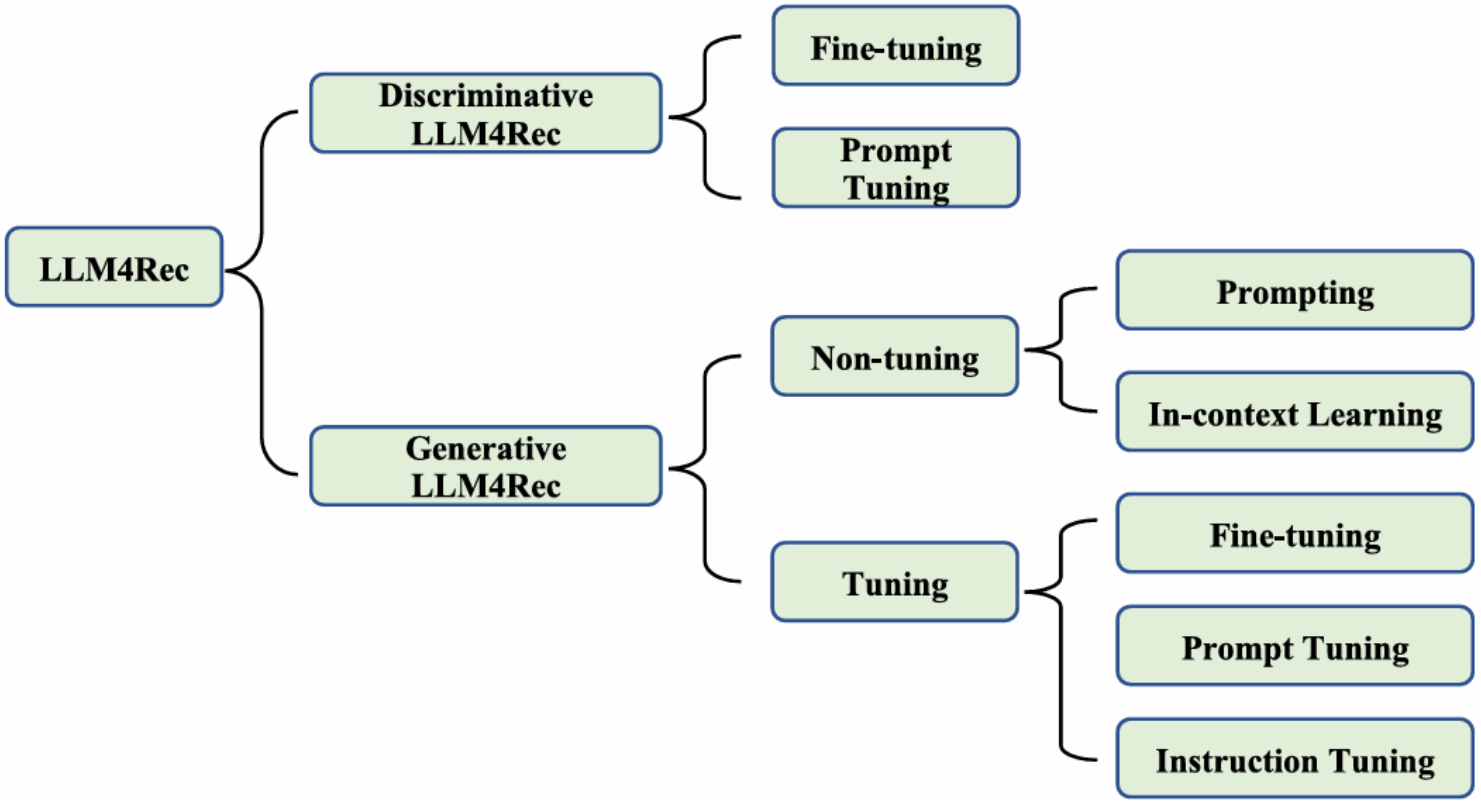


Figure 3 A taxonomy of the research for large language models on recommendation systems

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- LLM을 사용하는 추천시스템의 두 가지 메커니즘
  - ✓ 판별 모델 (Discriminative Models)
    - 주로 데이터를 분류하거나 예측하는 데 중점
    - 추천 시스템에서 discriminative models은 다음과 같은 과정을 통해 동작
      - LLM을 텍스트 표현의 '기초 작업 도구'로 사용하고, 추천 작업은 별도의 모델에서 처리
      - 즉, LLM은 데이터 전처리나 임베딩 생성 역할에 그침
  - ✓ 생성 모델(Generative Models)
    - 직접적인 추천 결과를 자연어 형태로 생성할 수 있는 능력을 가짐
    - 이를 통해 추천 작업 자체를 자연어 처리 작업으로 변환
      - **1) 추천 작업을 NLP 작업으로 변환**
        - » 사용자 입력 → LLM 입력 → LLM 출력
      - **2) 생성 결과를 직접 출력**
        - » In-context learning: LLM이 입력된 데이터(context)만으로 작업을 수행
        - » Prompt tuning: 특정 추천 작업에 적합한 지시문(prompt)을 설계해 LLM을 출력을 최적화
        - » Instruction tuning: 다양한 추천 작업 유형에 대해 학습하여, zero-/few-shot 능력을 강화

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- LLM을 추천 시스템에 통합하는 방식 (3가지)

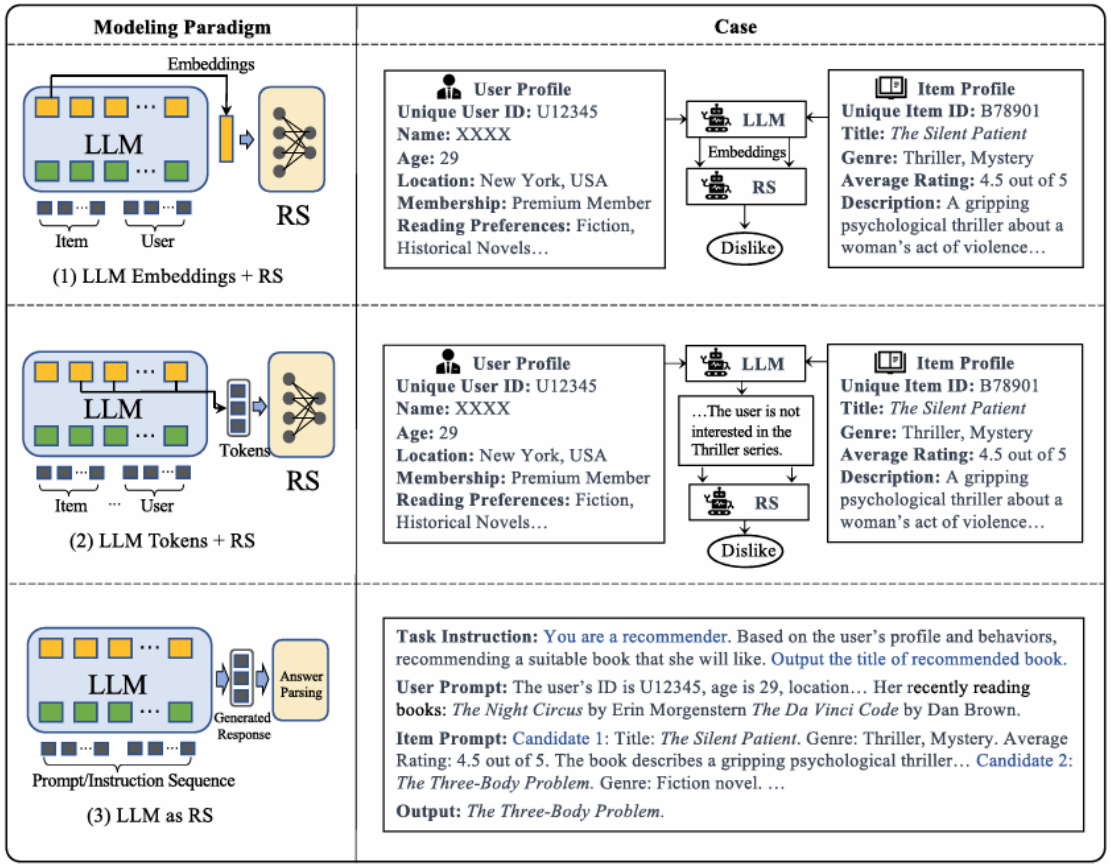
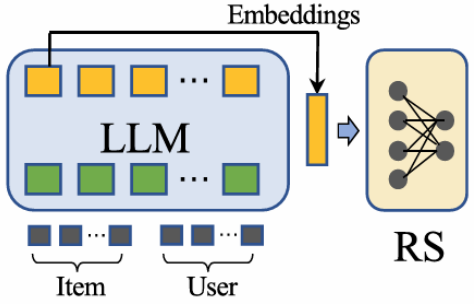


Figure 2 Three representative modeling paradigms of the research for large language models on recommendation systems

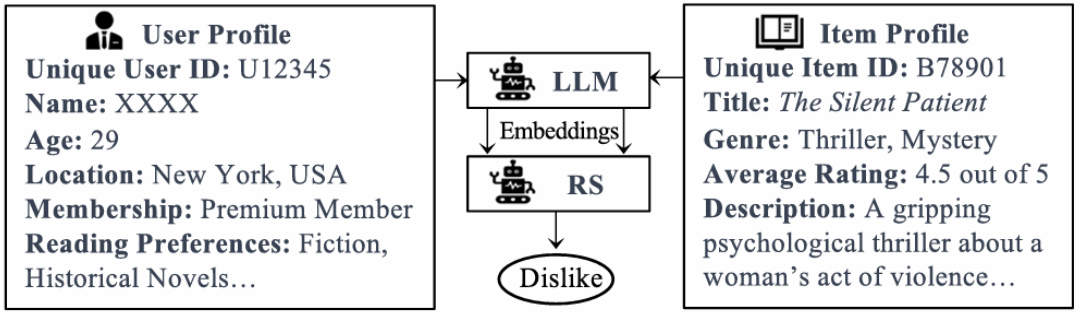
LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- LLM을 추천 시스템에 통합하는 방식 (3가지)
  - ✓ (1) LLM Embeddings + RS
    - LLM을 feature 추출기로 활용
    - 사용자와 아이템의 feature를 입력 받고, 이를 바탕으로 embedding을 생성함
    - 단, LLM은 추천 시스템을 보조하는 역할만! 핵심 추천 로직은 별도의 모델에서 처리됨
    - 지식 기반 embedding을 통해 다양한 추천 작업에 활용 가능
      - 즉, LLM을 단순 임베딩 모델로 활용
      - - 아이템과 사용자 input을 LLM에 입력하여 임베딩을 생성하는 구조
      - 이렇게 생성된 지식 기반 임베딩은 전통적인 추천 시스템 모델에서 활용하는 것이 목적!



(1) LLM Embeddings + RS



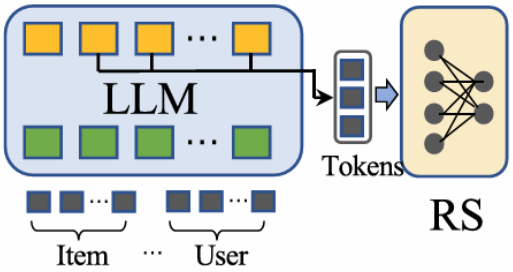
LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

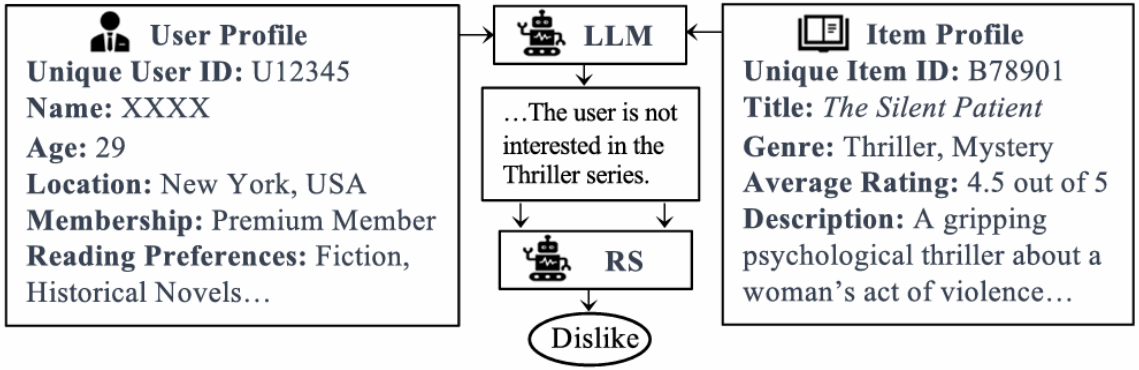
- LLM을 추천 시스템에 통합하는 방식 (3가지)

✓ (2) LLM Tokens + RS

- 사용자와 아이템의 feature를 바탕으로 token(단어, 문장 등)을 생성
- 이 token은 사용자 선호를 분석하거나 의사 결정에 사용됨
- 텍스트 데이터를 사용하여 사용자의 잠재적인 선호도를 더 잘 반영함
- LLM이 텍스트 표현을 생성하지만, 최종 추천 작업은 여전히 외부의 추천 시스템이 수행
  - (1)과 유사한 패러다임이나, 임베딩 벡터를 추출하는 것이 X → token 생성이 목적!
  - 생성된 token은 semantic mining을 비롯한 추천 시스템의 의사결정 과정에 사용



(2) LLM Tokens + RS



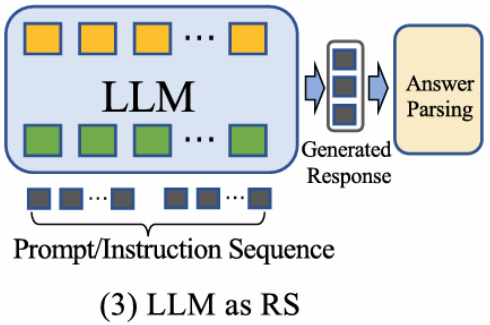
LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- LLM을 추천 시스템에 통합하는 방식 (3가지)

✓ (3) LLM as RS

- LLM이 단독적으로 추천 시스템의 역할을 수행
- 별도의 추천 모델이 필요 X
- Input은 profile description, behavior prompt, task instruction으로 구성
- Output은 합리적인 추천 결과



(3) LLM as RS

Task Instruction: You are a recommender. Based on the user's profile and behaviors, recommending a suitable book that she will like. Output the title of recommended book.

User Prompt: The user's ID is U12345, age is 29, location… Her recently reading books: *The Night Circus* by Erin Morgenstern *The Da Vinci Code* by Dan Brown.

Item Prompt: Candidate 1: Title: *The Silent Patient*. Genre: Thriller, Mystery. Average Rating: 4.5 out of 5. The book describes a gripping psychological thriller… Candidate 2: *The Three-Body Problem*. Genre: Fiction novel. …

Output: *The Three-Body Problem*.

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- LLM 기반 추천 시스템에서 사용되는 domain adaption 방법

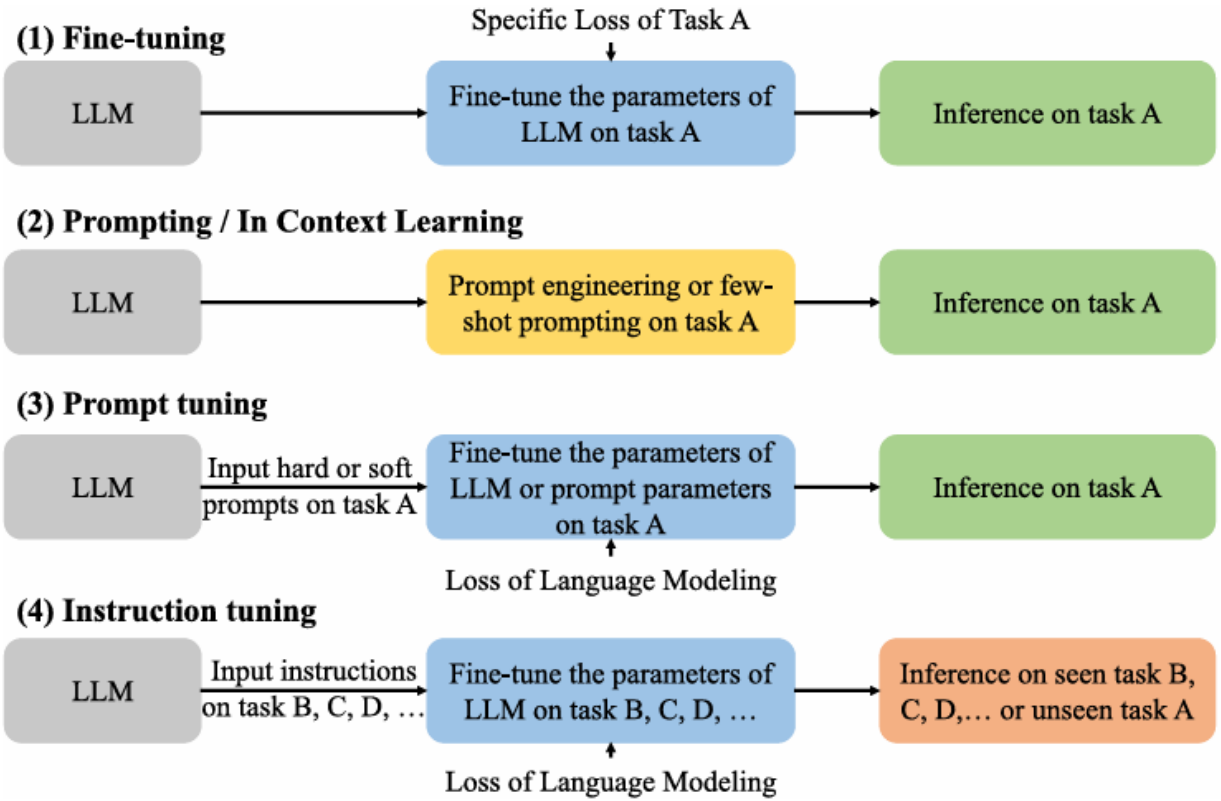


Figure 4 Detailed explanation of five different training (domain adaption) manners for LLM-based recommendations

LLM-based recommendation

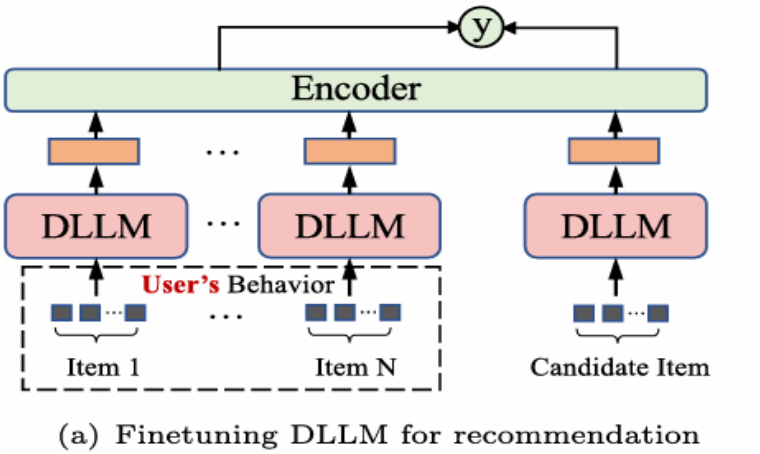
[2024][WWW] A survey on large language models for recommendation

- Discriminative LLMs for Recommendation
  - ✓ 추천 시스템에서 Discriminative LLM(DLLM)이란?
    - 주로 BERT 계열 모델을 의미
    - 다양한 분야에서 downstream task로 활용
      - 주로 임베딩 모델의 backbone으로 활용됨 (추천 시스템에서도 마찬가지)
  - ✓ 대부분의 기존 연구는 BERT와 같은 사전학습 모델의 표현을 도메인별 데이터와 정렬하기 위해 "Fine-Tuning" 전략을 사용

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- Discriminative LLMs for Recommendation
  - ✓ Fine-tuning
    - 핵심 아이디어
      - 대규모 텍스트 데이터에서 풍부한 언어적 표현을 학습한 언어 모델을 특정 작업이나 도메인에 맞게 적응시키는 것
      - 모델을 해당 작업에 특화된 데이터로 추가 학습을 수행!
  - ✓ 즉, BERT 기반의 Fine-tuning을 추천 시스템에 접목하면?
    - 추천의 정확도가 올라갈 뿐만 아니라, cold-start 문제에 robust한 추천 모델이 됨

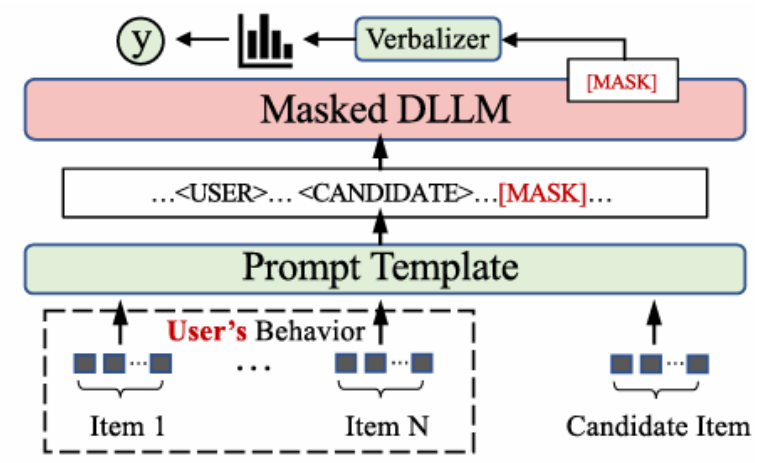


(a) Finetuning DLLM for recommendation

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- Discriminative LLMs for Recommendation
  - ✓ Prompt-Tuning
    - 추천 튜닝 목표를 사전 학습된 loss와 정렬시키는 것을 목표
      - Hard prompt
      - Soft prompt
    - 즉, LLM 모델 자체의 파라미터는 건드리지 않고, 프롬프트를 베이스 모델에 맞게 학습시킴
    - Fine-tuning에 비해 비교적 간단한 작업



(b) Prompt tuning DLLM for recommendation

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- Generative LLMs for Recommendation
  - ✓ DLLM 기반 접근법
    - LLM이 학습한 표현을 추천 도메인에 정렬 시키는 데 초점
  - ✓ GLLM 기반 접근법
    - 추천 작업을 자연어 작업으로 변환한 뒤, In-context learning, prompt tuning, instruction tuning 등의 기술을 적용하여 LLM이 직접 추천 결과를 생성하도록 유도함

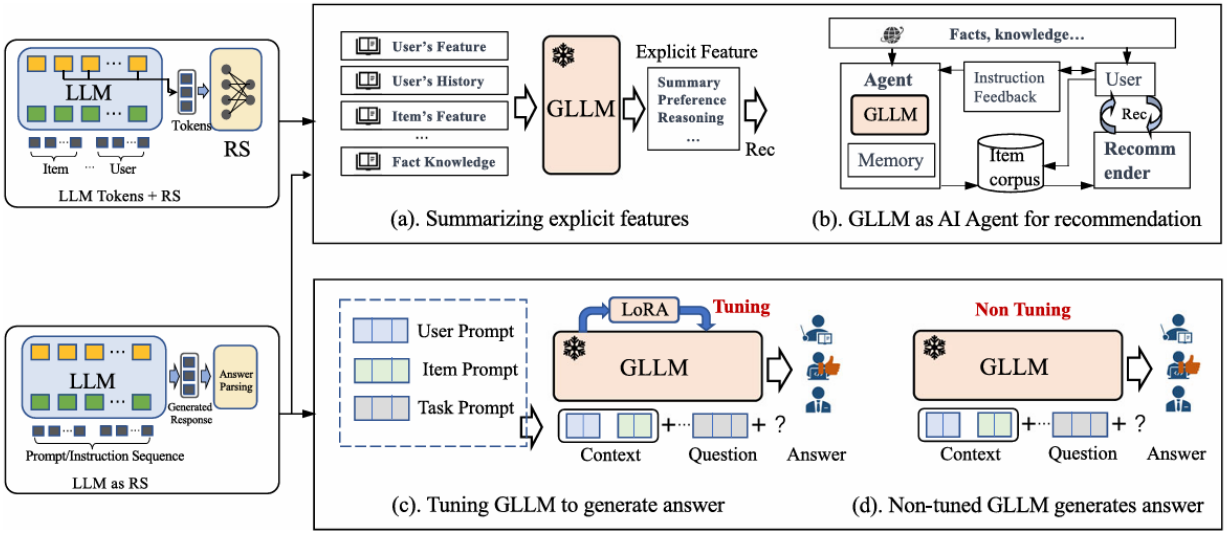


Figure 6 Generative LLMs for recommendation

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- Generative LLMs for Recommendation
  - ✓ 두 개의 패러다임으로 구분
    - Non-tuning 패러다임
      - Prompting, In-context learning
    - Tuning 패러다임
      - Tuning, Prompt tuning, Instruction tuning

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- 데이터셋

Table 2 A list of common datasets used in existing LLM-based recommendation methods

Name	Scene	Tasks	Information	URL
Amazon review [118]	Commerce	Seq Rec / CF Rec	This is a large crawl of product reviews from Amazon. Ratings: 82.83 million, Users:20.98 million, Items: 9.35 million, Timespan: May 1996 - July 2014	http://jmcauley.ucsd.edu/data/amazon/
Amazon-M2 [87]	Commerce	Seq Rec / CF Rec	A large dataset of anonymized user sessions with their interacted products collected from multiple language sources at Amazon. It includes 3,606,249 train sessions, 361,659 test sessions, and 1,410,675 products.	https://arxiv.org/abs/2307.09688
Amazon review 2023 [119]	Commerce	Seq Rec / CF Rec	The dataset comprises over 570 million reviews and 48 million items from 33 categories.	https://amazon-reviews-2023.github.io
Steam [120]	Game	Seq Rec / CF Rec	Reviews represent a great opportunity to break down the satisfaction and dissatisfaction factors around games. Reviews: 7,793,069, Users: 2,567,538, Items: 15,474, Bundles: 615	https://cseweb.ucsd.edu/~jmcauley/datasets.html#steam_data
MovieLens	Movie	General	The dataset consists of 4 sub-datasets, which describe users' ratings to moives and free-text tagging activities from MovieLens, a movie recommendation service.	https://grouplens.org/datasets/movielens/

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- 데이터셋

Table 2 continued

Name	Scene	Tasks	Information	URL
Yelp	Commerce	General	There are 6,990,280 reviews, 150,346 businesses, 200,100 pictures, 11 metropolitan areas, 908,915 tips by 1,987,897 users. Over 1.2 million business attributes like hours, parking, availability, etc.	https://www.yelp.com/dataset
Douban [121]	Movie, Music, Book	Seq Rec / CF Rec	This dataset includes three domains, i.e., movie, music, and book, and different kinds of raw information, i.e., ratings, reviews, item details, user profiles, tags (labels), and date.	https://github.com/ MarkWuNLP/ MultiTurnResponse Selection
MIND [122]	News	General	MIND contains about 160k English news articles and more than 15 million impression logs generated by 1 million users. Every news contains textual content including title, abstract, body, category, and entities.	https://msnews.github.io/assets/doc/ACL2020_MIND.pdf
U-NEED [123]	Commerce	Conversa -tion Rec	U-NEED consists of 7,698 fine-grained annotated pre-sales dialogues, 333,879 user behaviors, and 332,148 product knowledge tuples.	https://github.com/LeeeeoLiu/U-NEED

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- 데이터셋

Table 2 continued

Name	Scene	Tasks	Information	URL
KuaiSAR [124]	Video	Search and Rec	KuaiSAR contains genuine search and recommendation behaviors of 25,877 users, 6,890,707 items, 453,667 queries, and 19,664,885 actions within a span of 19 days on the Kuaishou app.	https://kuaisar.github.io/
Tenrec [125]	Video, Article	General	Tenrec is a large-scale benchmark dataset for recommendation systems. It contains around 5 million users and 140 million interactions.	https://tenrec0.github.io/
PixelRec [126]	Video	Seq Rec / CF Rec	PixelRec is a massive image-centric recommendation dataset that includes approximately 200 million user-image interactions, 30 million users, and 400,000 cover images. The texts and other aggregated attributes of videos are also included.	https://github.com/westlake-repl/PixelRec

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- 데이터셋
  - ✓ 대부분의 연구가 MovieLens, Amazon Books 등과 같은 벤치마크 데이터셋을 사용해 LLM의 추천 결과를 평가

  - ✓ 이러한 벤치마크 데이터셋은 규모가 적을 뿐더러, 도메인에서 문제가 O
    - Why? 영화나 책은 이미 LLM의 사전학습 단계에서 웹 문서에 많이 등장한 내용에 포함된 경우가 많음
    - 즉, 새로운 도메인의 추천 결과와 많이 다를 것임
    - 따라서 조금 더 포괄적인 평가가 가능한 데이터셋이 LLM 기반 추천시스템에서 필요

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- LLM 기반 추천시스템의 주요 강점 & challenges
  - ✓ 강점
    - Explainability (설명 가능성)
      - LLM은 자연어로 reasoning을 생성할 수 있어, 추천 이유에 대해 사람이 이해할 수 있는 형태로 설명이 가능함
    - Zero-/Few-shot Learning
      - 훈련되지 않은 task라도, 간단한 예시 or 프롬프트만으로도 새로운 추천 task 수행이 가능

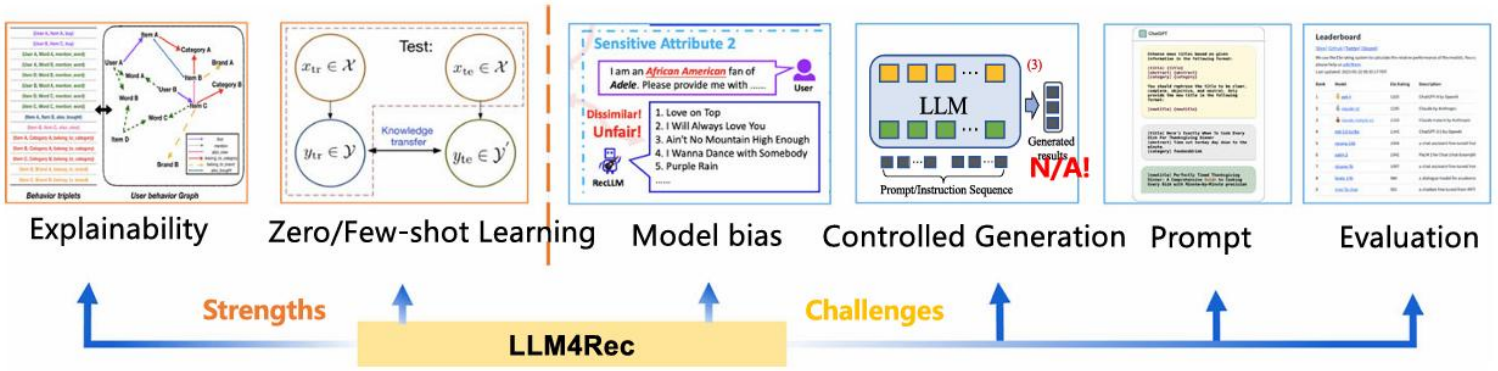


Figure 7 The major strengths and technical challenges of LLM4Rec

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- LLM 기반 추천시스템의 주요 강점 & challenges
  - ✓ Challenges
    - Model Bias (편향)
      - 사용자 특성(ex. 인종, 성별 등)에 따라 편향된 추천이 발생할 수 있음
    - Controlled Generation (제어된 생성)
      - LLM은 출력 제어가 어려워, 특정한 유형/포맷의 추천 결과를 안정적으로 생성하기 어려움
    - Prompt Design (프롬프트 설계)
      - 적절한 프롬프트를 설계하는 것이 매우 중요하고, 그 품질에 따라 성능이 좌우됨
    - Evaluation (평가)
      - 기존의 평가 지표(NDCG, Recall 등)로는 LLM의 추천 품질을 충분히 설명하기 어려움
      - 새로운 평가 방법이 필요

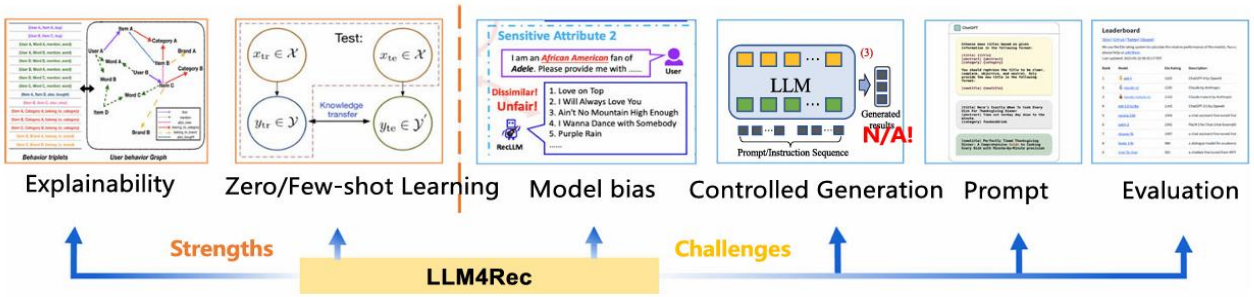


Figure 7 The major strengths and technical challenges of LLM4Rec

LLM-based recommendation

[2024][WWW] A survey on large language models for recommendation

- **앞으로 해결해야 할 기술적 과제**
  - ✓ **효율성 문제**
    - 대규모 모델의 높은 연산 비용과 메모리 요구 사항
    - 실시간 추천 시스템에 적용하기 위한 최적화 필요
  - ✓ **일반화 능력**
    - 특정 도메인 또는 작업에 과도하게 특화되지 않으면서도 보편적인 추천 성능을 유지
    - 다양한 사용자와 아이템 데이터를 다룰 수 있는 더 나은 적응성 필요
  - ✓ **데이터 제한**
    - Cold-start 문제
    - 사용자-아이템 상호작용 데이터가 부족한 환경에서의 모델 성능
  - ✓ **해석 가능성**
    - LLM의 추천 이유를 이해하고 설명할 수 있는 능력이 부족
    - 사용자 신뢰를 향상시키기 위한 투명한 추천 프로세스 필요

3. 모델 비교 (LLM vs. w/o LLM)

모델 비교 (LLM vs. w/o LLM)

[2018][IEEE][SASRec] Self-Attentive Sequential Recommendation

- SASRec ☞ Transformer 기반의 sequential recommendation
  - ✓ 최신 LLM-based recommendation 논문들의 베이스라인으로 자주 언급
  - ✓ 기존의 sequential recommender인 Markov Chains(MC)과 RNN 계열의 단점을 보완
  - ✓ 당시 NLP task에서 SOTA였던 Transformer 모델을 추천 시스템에 처음 도입한 모델

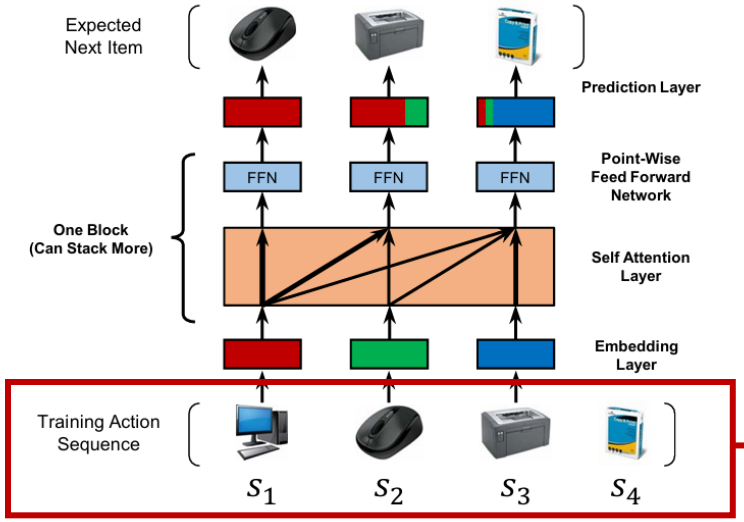


Figure 1: A simplified diagram showing the training process of SASRec. At each time step, the model considers all previous items, and uses attention to 'focus on' items relevant to the next action.

- SASRec이 Transformer의 self-attention 메커니즘을 이용해 seq rec을 수행하는 과정
- 각 시간 단계에서 이전의 모든 아이템을 고려
- 다음 행동과 관련된 아이템에 'focus on' 하고자 attention을 사용

사용자의 과거 행동 시퀀스
(순서대로 클릭하거나 구매한 아이템)

모델 비교 (LLM vs. w/o LLM)

[2018][IEEE][SASRec] Self-Attentive Sequential Recommendation

- SASRec ☞ Transformer 기반의 sequential recommendation
  - ✓ 최신 LLM-based recommendation 논문들의 베이스라인으로 자주 언급
  - ✓ 기존의 sequential recommender인 Markov Chains(MC)과 RNN 계열의 단점을 보완
  - ✓ 당시 NLP task에서 SOTA였던 Transformer 모델을 추천 시스템에 처음 도입한 모델

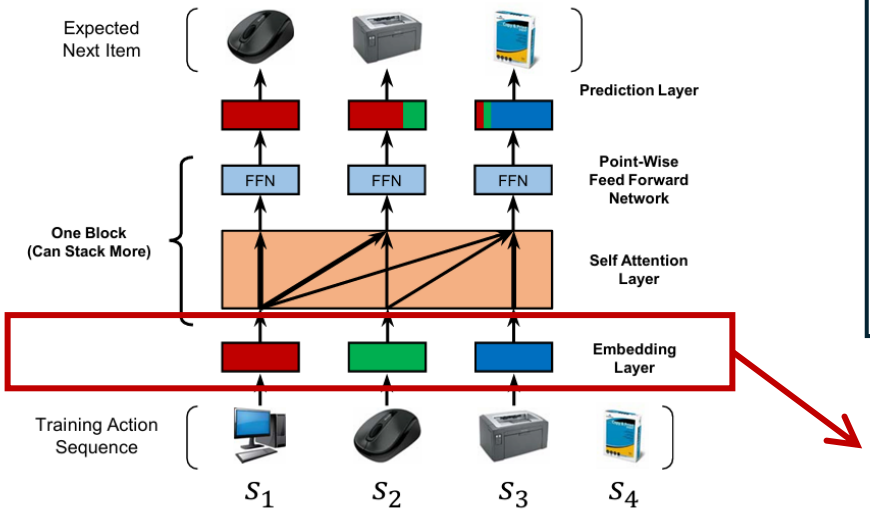


Figure 1: A simplified diagram showing the training process of SASRec. At each time step, the model considers all previous items, and uses attention to 'focus on' items relevant to the next action.

- SASRec이 Transformer의 self-attention 메커니즘을 이용해 seq rec을 수행하는 과정
- 각 시간 단계에서 이전의 모든 아이템을 고려
- 다음 행동과 관련된 아이템에 'focus on' 하고자 attention을 사용

각 아이템에 대해
고유한 아이템 임베딩 벡터를 가져옴

위치 정보를 반영하고자
positional embedding을 더함

모델 비교 (LLM vs. w/o LLM)

[2018][IEEE][SASRec] Self-Attentive Sequential Recommendation

- SASRec ☞ Transformer 기반의 sequential recommendation
  - ✓ 최신 LLM-based recommendation 논문들의 베이스라인으로 자주 언급
  - ✓ 기존의 sequential recommender인 Markov Chains(MC)과 RNN 계열의 단점을 보완
  - ✓ 당시 NLP task에서 SOTA였던 Transformer 모델을 추천 시스템에 처음 도입한 모델

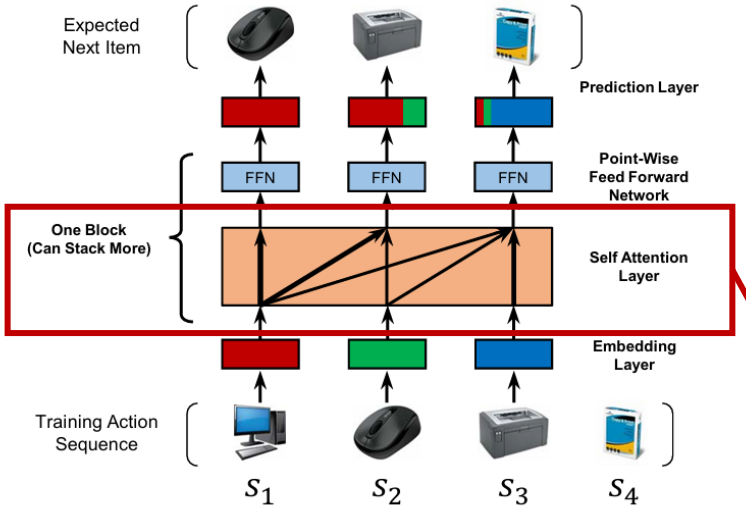


Figure 1: A simplified diagram showing the training process of SASRec. At each time step, the model considers all previous items, and uses attention to 'focus on' items relevant to the next action.

- SASRec이 Transformer의 self-attention 메커니즘을 이용해 seq rec을 수행하는 과정
- 각 시간 단계에서 이전의 모든 아이템을 고려
- 다음 행동과 관련된 아이템에 'focus on' 하고자 attention을 사용

과거 시점의 아이템 중
어떤 것이 다음 아이템 예측에 중요한지 attention

현재 시점(t)보다 이후 시점을 보지 않도록
causality mask가 적용됨

모델 비교 (LLM vs. w/o LLM)

[2018][IEEE][SASRec] Self-Attentive Sequential Recommendation

- SASRec ☞ Transformer 기반의 sequential recommendation
  - ✓ 최신 LLM-based recommendation 논문들의 베이스라인으로 자주 언급
  - ✓ 기존의 sequential recommender인 Markov Chains(MC)과 RNN 계열의 단점을 보완
  - ✓ 당시 NLP task에서 SOTA였던 Transformer 모델을 추천 시스템에 처음 도입한 모델

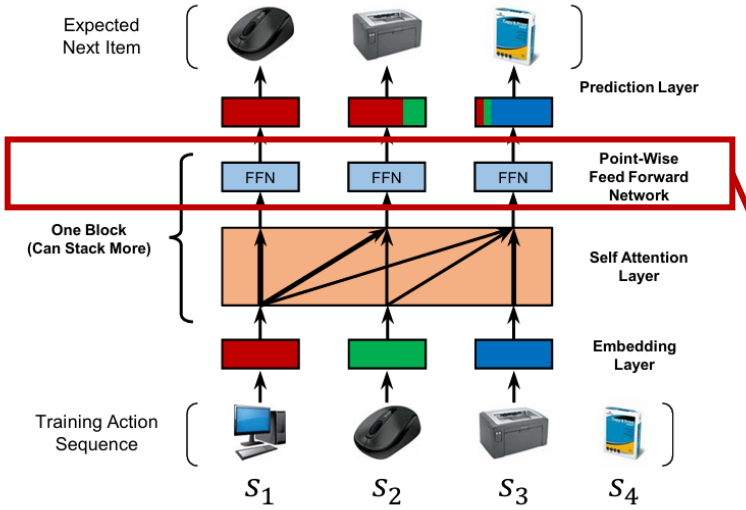


Figure 1: A simplified diagram showing the training process of SASRec. At each time step, the model considers all previous items, and uses attention to 'focus on' items relevant to the next action.

- SASRec이 Transformer의 self-attention 메커니즘을 이용해 seq rec을 수행하는 과정
- 각 시간 단계에서 이전의 모든 아이템을 고려
- 다음 행동과 관련된 아이템에 'focus on' 하고자 attention을 사용

각 위치의 attention output에 대해 개별적으로 FFN을 적용

모델 비교 (LLM vs. w/o LLM)

[2018][IEEE][SASRec] Self-Attentive Sequential Recommendation

- SASRec ☞ Transformer 기반의 sequential recommendation
  - ✓ 최신 LLM-based recommendation 논문들의 베이스라인으로 자주 언급
  - ✓ 기존의 sequential recommender인 Markov Chains(MC)과 RNN 계열의 단점을 보완
  - ✓ 당시 NLP task에서 SOTA였던 Transformer 모델을 추천 시스템에 처음 도입한 모델

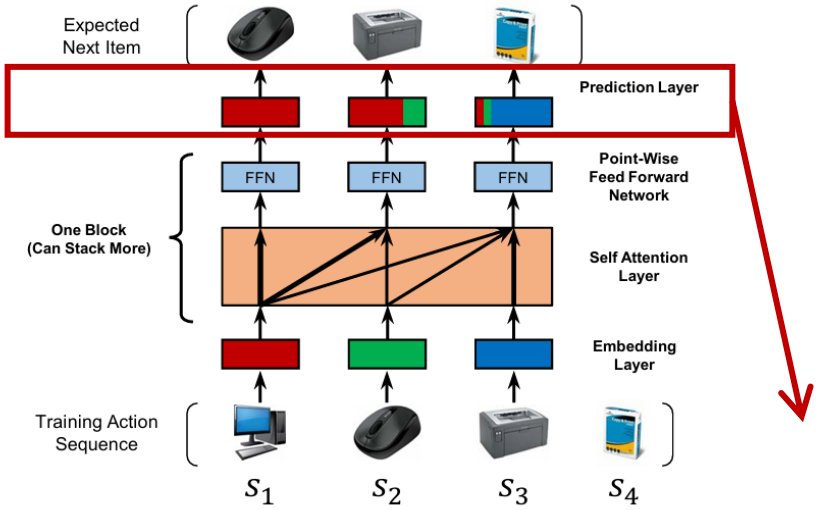


Figure 1: A simplified diagram showing the training process of SASRec. At each time step, the model considers all previous items, and uses attention to 'focus on' items relevant to the next action.

- SASRec이 Transformer의 self-attention 메커니즘을 이용해 seq rec을 수행하는 과정
- 각 시간 단계에서 이전의 모든 아이템을 고려
- 다음 행동과 관련된 아이템에 'focus on' 하고자 attention을 사용

예측 목표: 시점 t에서 다음에 올 아이템을 맞히는 것

이를 위해 현재 시점까지의 output과 전체 아이템 임베딩 간의 내적을 계산

이 점수가 높을수록, 해당 아이템이 다음에 올 확률이 높다!

모델 비교 (LLM vs. w/o LLM)

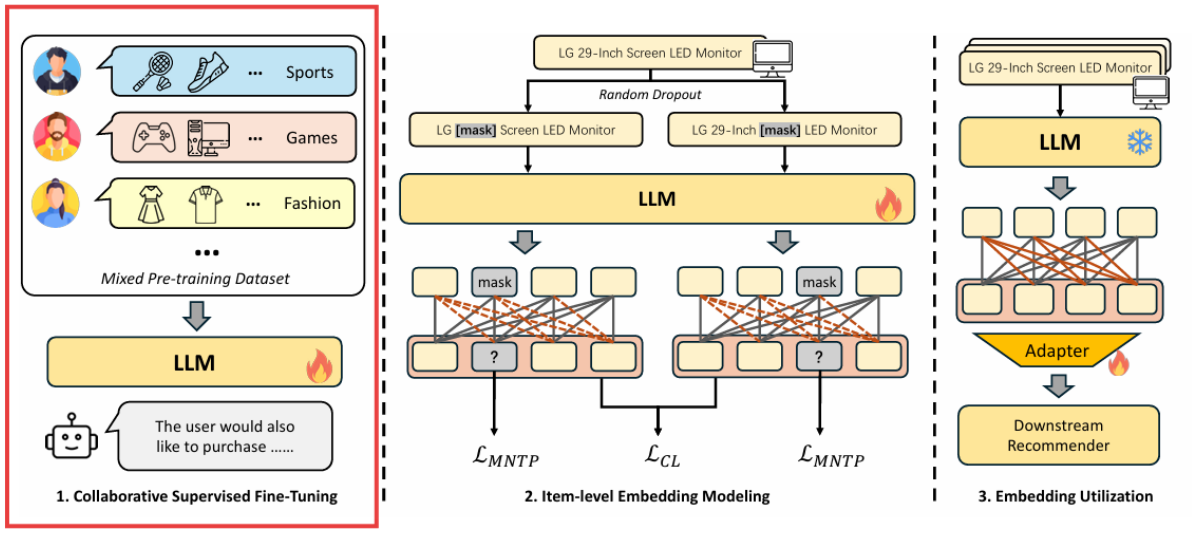
[2025][KDD][LLM2Rec] Large Language Models Are Powerful Embedding Models for Sequential Recommendation

- LLM2Rec
  - ✓ 기존의 Sequential recommendation에서 단순 아이템 ID 중심 임베딩만 수행
    - 한계점: 아이템 간의 의미적 유사성을 반영하기 어려움 (SASRec도 해당)
  - ✓ LLM을 활용해 텍스트 의미 정보 + CF 신호를 통합해 추천 품질과 일반화 능력을 크게 향상
    - 도메인 일반화 능력 & cold-start robustness가 크게 향상됨
  - ✓ 즉, seq rec에서의 의미 기반 추천을 가능하게 함
  - ✓ **pre-training이 완료된 후 LLM을 freeze(동결) 상태로 유지한 채,** downstream sequential recommendation을 수행
    - 다시 말해, **추가적인 fine-tuning 없이,** LLM이 생성한 **item-level embedding만 활용하여** 기존 sequential recommender(GRU4Rec, SASRec 등)에 **plug-in**하는 방식
  - ✓ **모델 구조**
    - 1) Collaborative Supervised Fine-Tuning (CSFT)
    - 2) Item-level Embedding Modeling (IEM)
    - 3) Embedding Utilization

모델 비교 (LLM vs. w/o LLM)

[2025][KDD][LLM2Rec] Large Language Models Are Powerful Embedding Models for Sequential Recommendation

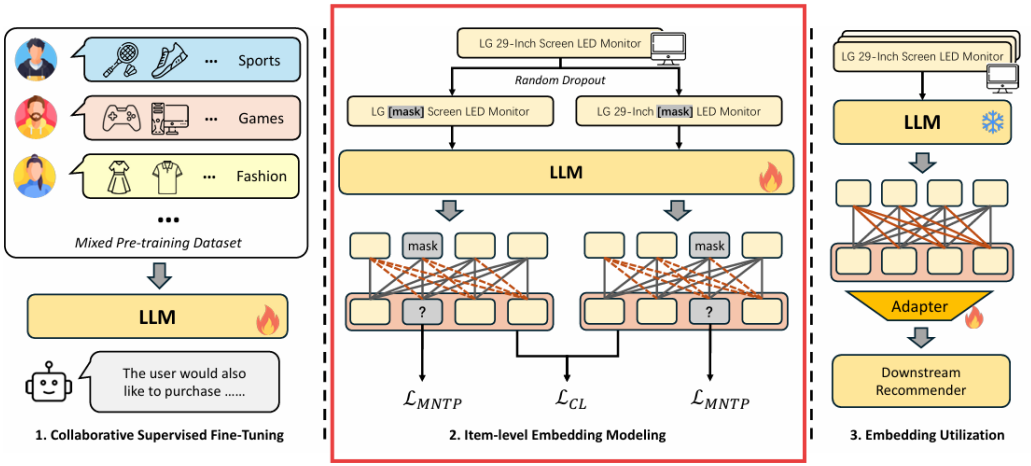
- 모델 구조
  - ✓ 1) Collaborative Supervised Fine-Tuning (CSFT)
    - 사용자 행동 시퀀스를 자연어 형태의 instruction으로 구성해 LLM에 입력함
    - 그 시퀀스 다음에 등장할 아이템의 title을 autoregressive 방식으로 예측하도록 학습시킴
      - 입력 시퀀스는 단순히 item titles만으로 구성되며, 일부 구분자(예: 쉼표)를 제외하고는 불필요한 문맥을 제거
    - LLM이 사용자-아이템 상호작용으로부터 CF 정보를 학습할 수 있도록 함



모델 비교 (LLM vs. w/o LLM)

[2025][KDD][LLM2Rec] Large Language Models Are Powerful Embedding Models for Sequential Recommendation

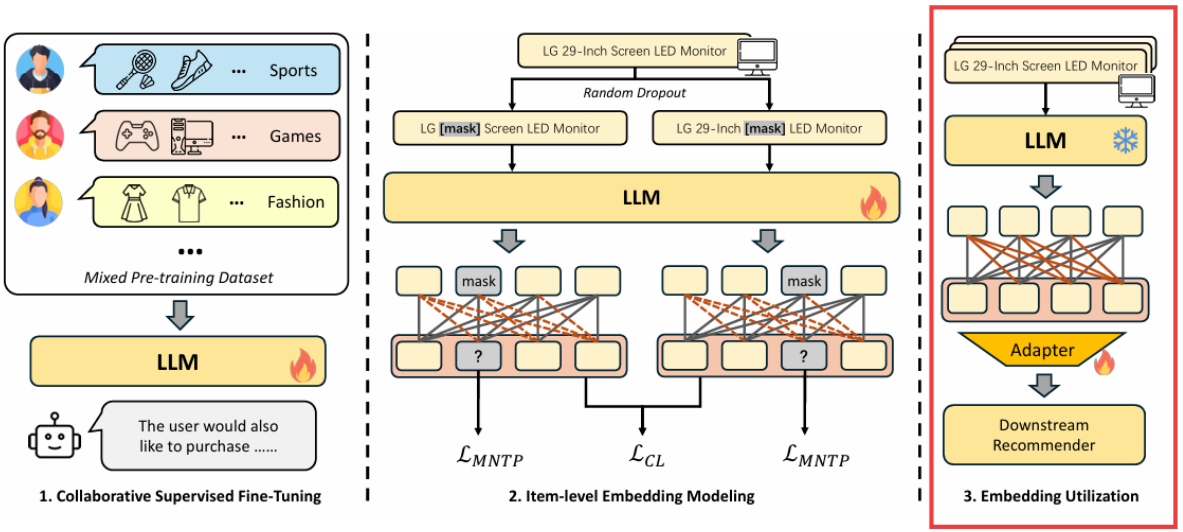
- 모델 구조
  - ✓ 2) Item-level Embedding Modeling (IEM)
    - CSFT로 학습된 LLM을 기반으로 고품질의 item-level embedding을 생성하기 위해 아키텍처를 수정하고, 추가 pre-trainin을 수행
    - 기존 decoder-only LLM의 **causal attention mask**를 제거하여 **bidirectional attention**을 적용, 아이템 title 내 전후 문맥 정보를 모두 활용
    - 이후 **두 개의 masking된 view**를 생성하고, Item-level Contrastive Learning (CL)을 수행해 동일 아이템 간 embedding 유사도를 높임
    - 최종 item embedding은 LLM의 출력 hidden state들을 **평균 풀링(avg pooling)** 하여 구성



모델 비교 (LLM vs. w/o LLM)

[2025][KDD][LLM2Rec] Large Language Models Are Powerful Embedding Models for Sequential Recommendation

- 모델 구조
  - ✓ 3) Embedding Utilization
    - 학습된 item-level embedding을 **동일한 downstream sequential recommender**에 적용하여 성능을 평가
    - LLM의 출력에 **경량화된 adapter 모듈**을 추가하여 downstream 모델과의 연결을 용이하게 하고, task-specific fine-tuning 없이도 활용 가능
    - 기존 sequential 모델 구조 (예: SASRec, GRU4Rec)에 통합하여 **추천 정확도(Recall@K, NDCG@K)** 측면에서 embedding 품질을 검증



모델 비교 (LLM vs. w/o LLM)

	SASRec (2018, w/o LLM)	LLM2Rec (2025, LLM)
input	ID 기반 user/item 시퀀스	ID + 텍스트 시퀀스 (item titles 등)
기반 모델	Transformer 기반 self-attention	Pre-trained LLM 기반 임베딩 모델
추천 방식	Item ID 시퀀스를 학습 -> 다음 item 예측	Item 시퀀스를 생성해 다음 item 예측
장점	- 연산 효율성 높음 - Data sparse 환경에 유연	- 텍스트 의미 해석 + CF 통합 -> 일반화 능력 우수 - 다양한 도메인에 적용 가능 - Text 기반의 의미 이해로 cold-start 대응 가능
한계점	- ID 기반 모델 -> cold start 문제 존재 - 텍스트 의미를 반영하지 못함 - 새로운 아이템에 대한 대응 불가 (cold start)	- LLM 학습 / 추론 비용 큼

모델 비교 (LLM vs. w/o LLM)

[2018][IEEE][SASRec] Self-Attentive Sequential Recommendation

- 실험 결과
  - ✓ in-domain 데이터셋
    - 학습 시 사용한 도메인과 동일
      - 모델이 기존 방식으로도 잘 작동하는 익숙한 환경
        - » 베이스라인은 일반 임베딩 모델 / 추천 특화 임베딩 모델

In-Domain Datasets												
Models		Games				Arts				Movies			
		R@10	N@10	R@20	N@20	R@10	N@10	R@20	N@20	R@10	N@10	R@20	N@20
GRU4Rec	BERT	0.0365	0.0184	0.0573	0.0236	0.0363	0.0191	0.0559	0.0240	0.0243	0.0126	0.0383	0.0160
	GTE	0.0540	0.0290	0.0792	0.0353	0.0348	0.0185	0.0569	0.0240	0.0396	0.0195	0.0583	0.0242
	BGE	0.0491	0.0261	0.0760	0.0329	0.0413	0.0221	0.0632	0.0276	0.0379	0.0187	0.0587	0.0239
	LLM2Vec	0.0540	0.0286	0.0784	0.0348	0.0473	0.0274	0.0678	0.0325	0.0370	0.0187	0.0557	0.0234
	BLAIR	0.0455	0.0245	0.0713	0.0309	0.0416	0.0233	0.0639	0.0289	0.0379	0.0188	0.0583	0.0239
	EasyRec	0.0450	0.0235	0.0700	0.0298	0.0436	0.0232	0.0643	0.0284	0.0356	0.0180	0.0551	0.0229
	LLMEmb	0.0544	0.0298	0.0775	0.0357	0.0480	0.0277	0.0673	0.0325	0.0377	0.0196	0.0538	0.0236
	LLM2Rec	**0.0624**	**0.0344**	**0.0874**	**0.0408**	**0.0590**	**0.0366**	**0.0802**	**0.0419**	**0.0419**	**0.0214**	**0.0595**	**0.0258**
	%Improv.	14.76%	15.46%	10.35%	14.31%	22.83%	32.32%	18.16%	29.03%	5.92%	9.46%	1.46%	6.77%
SASRec	BERT	0.0585	0.0311	0.0863	0.0381	0.0650	0.0405	0.0869	0.0460	0.0447	0.0240	0.0646	0.0290
	GTE	0.0641	0.0349	0.0911	0.0418	0.0644	0.0394	0.0880	0.0454	0.0570	0.0300	0.0817	0.0363
	BGE	0.0733	0.0410	0.1022	0.0483	0.0748	0.0475	0.1006	0.0540	0.0626	0.0350	0.0847	0.0406
	LLM2Vec	0.0740	0.0407	0.1029	0.0480	0.0770	0.0506	0.1007	0.0566	0.0662	0.0384	0.0874	0.0438
	BLAIR	0.0654	0.0361	0.0954	0.0437	0.0648	0.0379	0.0906	0.0444	0.0581	0.0315	0.0801	0.0370
	EasyRec	0.0647	0.0357	0.0926	0.0428	0.0658	0.0395	0.0929	0.0463	0.0528	0.0278	0.0739	0.0331
	LLMEmb	0.0813	0.0487	0.1085	0.0555	0.0865	0.0601	0.1086	0.0657	0.0659	0.0390	0.0837	0.0435
	LLM2Rec	**0.0865**	**0.0521**	**0.1157**	**0.0595**	**0.0925**	**0.0637**	**0.1142**	**0.0692**	**0.0705**	**0.0429**	**0.0895**	**0.0477**
	%Improv.	6.42%	6.92%	6.70%	7.04%	6.95%	5.97%	5.16%	5.29%	6.49%	10.01%	2.40%	9.13%

모델 비교 (LLM vs. w/o LLM)

[2018][IEEE][SASRec] Self-Attentive Sequential Recommendation

- 실험 결과
  - ✓ out-of-domain 데이터셋
    - ▪ 학습에 사용되지 않은 새로운 도메인
      - – 기존 모델은 성능 저하, LLM2Rec의 도메인 일반화 능력을 입증
        - » 즉, 다양한 환경에서 fine-tuning 없이도 reasonable한 성능을 낼 수 있음

Out-Of-Domain Datasets

Models		Sports				Baby				Goodreads			
		R@10	N@10	R@20	N@20	R@10	N@10	R@20	N@20	R@10	N@10	R@20	N@20
GRU4Rec	BERT	0.0335	0.0183	0.0499	0.0224	0.0111	0.0050	0.0252	0.0086	0.0851	0.0412	0.1226	0.0506
	GTE	0.0295	0.0147	0.0459	0.0188	0.0226	0.0116	0.0340	0.0145	0.1169	0.0599	0.1701	0.0733
	BGE	0.0489	0.0281	0.0685	0.0330	0.0252	0.0131	0.0364	0.0158	0.1072	0.0585	0.1517	0.0697
	LLM2Vec	0.0663	0.0464	0.0810	0.0501	0.0254	0.0138	0.0362	0.0165	0.1174	0.0655	0.1643	0.0773
	BLAIR	0.0537	0.0316	0.0735	0.0366	0.0207	0.0099	0.0316	0.0127	0.0939	0.0496	0.1339	0.0596
	EasyRec	0.0492	0.0270	0.0674	0.0315	0.0207	0.0105	0.0275	0.0122	0.0951	0.0477	0.1364	0.0581
	LLMEmb	0.0705	0.0482	0.0861	0.0521	0.0252	0.0136	0.0378	0.0168	0.1219	0.0701	0.1667	0.0814
	LLM2Rec	**0.0828**	**0.0632**	**0.0948**	**0.0662**	**0.0327**	**0.0181**	**0.0463**	**0.0216**	**0.1299**	**0.0761**	**0.1738**	**0.0872**
	%Improv.	17.50%	31.18%	10.07%	27.06%	28.55%	31.61%	22.32%	28.51%	6.58%	8.68%	2.17%	7.15%
SASRec	BERT	0.0860	0.0649	0.1017	0.0689	0.0114	0.0050	0.0232	0.0080	0.1479	0.0858	0.1929	0.0972
	GTE	0.0823	0.0584	0.1001	0.0629	0.0264	0.0142	0.0387	0.0173	0.1488	0.0851	0.1944	0.0967
	BGE	0.0974	0.0736	0.1141	0.0778	0.0428	0.0250	0.0569	0.0286	0.1445	0.0813	0.1972	0.0945
	LLM2Vec	0.1079	0.0854	0.1234	0.0893	0.0561	0.0339	0.0722	0.0379	0.1424	0.0790	0.1906	0.0911
	BLAIR	0.0893	0.0614	0.1091	0.0664	0.0332	0.0180	0.0484	0.0218	0.1508	0.0860	0.2000	0.0984
	EasyRec	0.0887	0.0627	0.1061	0.0671	0.0271	0.0154	0.0381	0.0182	0.1445	0.0825	0.1908	0.0941
	LLMEmb	0.1131	0.0936	0.1257	0.0969	0.0659	0.0439	0.0807	0.0476	0.1374	0.0778	0.1838	0.0895
	LLM2Rec	**0.1170**	**0.0976**	**0.1289**	**0.1006**	**0.0708**	**0.0503**	**0.0850**	**0.0539**	**0.1530**	**0.0897**	**0.2017**	**0.1020**
	%Improv.	3.51%	4.26%	2.56%	3.89%	7.39%	14.56%	5.32%	13.04%	1.45%	4.37%	0.83%	3.73%

모델 비교 (LLM vs. w/o LLM)

결론

- **추천시스템에 LLM을 적용하면 뭐가 좋을까?**
  - ✓ 기존 seq rec에 **LLM 임베딩만 교체해도,**
    최신 임베딩 기법이나 전통적 방식(ID 임베딩 기반)을 뛰어넘는 성능을 달성할 수 있음
  - ✓ **별도의 모델 구조 변경 없이도 성능 향상** 가능 → 구조는 그대로, 임베딩만 바꿔도 효과 충분
  - ✓ **실험 결과, LLM 임베딩을 적용한 SASRec**이
    - 과거 방식(ID 임베딩 기반)
    - 최근 등장한 다양한 임베딩 기법들보다 일관되고 우수한 성능을 보임
  - ✓ 즉, **SASRec과 같은 전통적인 sequential recommendation 구조도 LLM 임베딩만 잘 적용한다면?**
    - 강력한 최신 seq rec 모델로 재탄생할 수 있음!
  - ✓ **LLM을 추천에 도입한다는 것 자체가** 단순한 임베딩 교체를 넘어
    추천 시스템의 효율성과 설계 방식에 근본적인 변화를 가져오는 전략

4. 연구 방향

연구 방향

LLM-based sequential recommendation

- LLM2Rec (베이스라인)

- ELMRec + ELCRec …. (방법론)

- 사용자의 latent한 intent를 단순한 relation이나 item 영역 구분이 아닌,
- 의미 흐름을 반영한 가상의 노드로 모델링