

Introduction to Sequential Recommender Systems

22.07.24

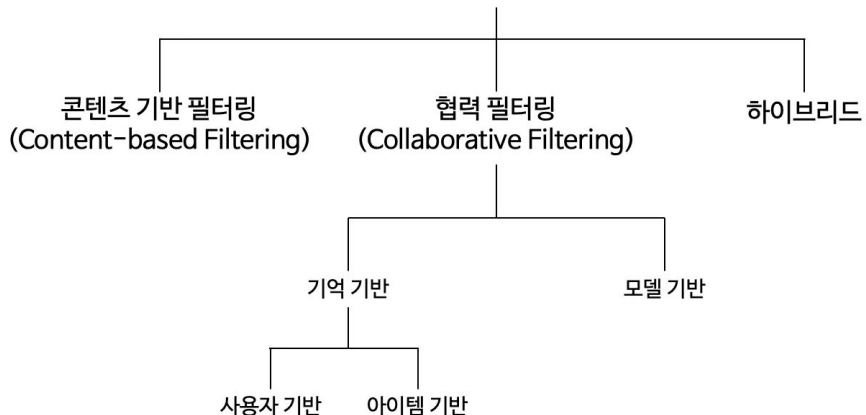
발표자 이지현

Introduction to Sequential Recommender Systems

- Introduction to Recommender System
- Deep Learning based Recommender System
 - Neural Collaborative Filtering
- Sequential Recommender System
 - Feature-level Deeper Self-Attention Network for Sequential Recommendation

Introduction to Recommender System

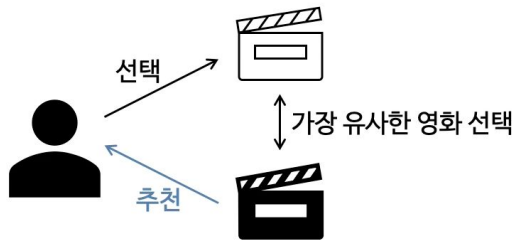
- 추천 시스템이란?
 - Youtube, 넷플릭스, 인스타그램 등등.
 - 인터넷이 발전됨에 따라 아이템 구매 및 선호에 대한 사용자의 피드백을 얻기 쉬워졌음. 이런 피드백을 바탕으로 과거의 사용자-아이템 간 데이터를 활용해 취향을 고려한 아이템을 추천하는 것.
- 추천 알고리즘 종류 (3가지)
 - Content-based Recommender Systems : 콘텐츠 기반 추천 시스템
 - Collaborative Filtering : 협업 필터링
 - 하이브리드



Introduction to Recommender System

Content-based Recommender Systems : 콘텐츠 기반 추천 시스템

- 콘텐츠 기반으로 분석하여 추천해주는 방식
- 사용자가 관심분야에 대해서 직접 입력한 정보나, 구매 내역, 평점 등을 기반으로 선호 아이템 파악
- 선호하는 아이템과 가장 유사한 다른 아이템 추천



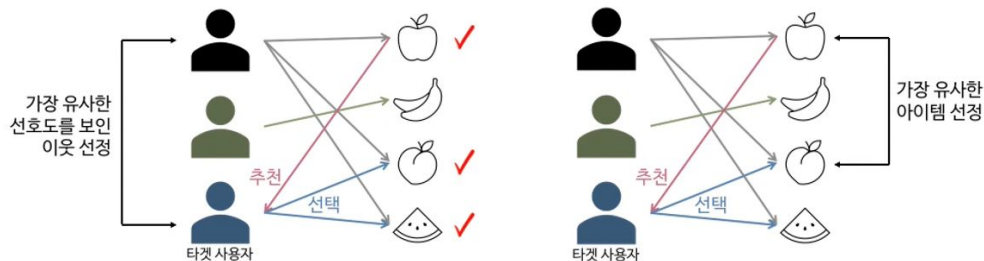
- 장점
 - 아이템의 속성 (콘텐츠)을 기반으로 추천하기 때문에 이전 선택 이력이 없는 새로운 아이템도 추천 가능
- 단점
 - 해당 고객에 대한 데이터가 부족한 경우 추천 성능 보장이 어려움
 - 과도한 특수화 (Over Specialization)
 - 이전에 구매 및 선택한 아이템과 비슷한 제품만 추천하는 경향

Introduction to Recommender System

Collaborative Filtering : 협업 필터링

‘특정 아이템에 대하여 선호도가 유사한 고객들은 다른 아이템에 대해서도 비슷한 선호도를 보일 것이다’

- [기억 기반] 사용자 기반 협력 필터링
 - 사용자 간의 유사도를 측정하여 유사도 높은 이웃이 선택한 아이템 중에서 추천
- [기억 기반] 아이템 기반 협력 필터링
 - 아이템 간 유사도를 측정하여 유사도 높은 아이템 추천



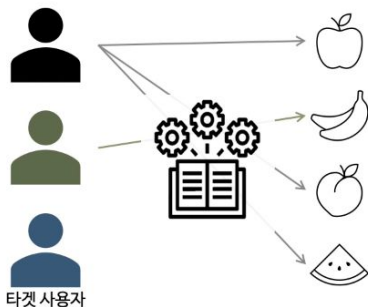
- 단점
 - **Cold Start Problem** : 한 사용자에게 대한 충분한 데이터가 부족한 경우 선호도 예측 불가능
 - **First Rater** : 새로운 아이템이 등장하여 평점 점수가 부족한 경우 추천 불가능
 - **Grey Sheep Problem** : 일관성이 없는 의견을 가진 사용자들의 데이터는 추천에 혼란을 줌
 - **Shilling Attack** : 악의적으로 평가 점수를 긍정/부정으로 입력하는 경우 추천에 방해가 됨

Introduction to Recommender System

Collaborative Filtering : 협업 필터링

‘특정 아이템에 대하여 선호도가 유사한 고객들은 다른 아이템에 대해서도 비슷한 선호도를 보일 것이다’

- [모델 기반] 사용자 기반 협력 필터링
 - 데이터에 내제 되어있는 복잡한 패턴을 발견하도록 다양한 모델을 활용한 기법
 - 실제 데이터에 적용했을 때 성능이 우수함



- Matrix Factorization
 - ‘사용자와 아이템 사이에는 사용자의 행동과 평점에 영향을 끼치는 잠재된 특성이 있을 것이다’
 - 크기가 크며 복잡한 데이터로도 쉽고 빠르게 분석 진행 가능
 - 특정 사용자의 특정 아이템에 대한 평점을 알고 싶을 때 사용
 - https://github.com/easy-note/Recommendation_System/blob/main/lecture_notes/section5.pdf

Deep Learning based Recommender System

- 데이터 내에 존재하는 여러 추상적이며 복잡한 내용을 학습하여 핵심을 알아내는 딥러닝의 발전
- 추천 시스템의 관점에서, 사용자의 복잡한 정보를 학습하여 핵심을 알아낼 수 있다면?
- 현실의 데이터가 더 많은 사용자, 더 많은 아이템에 대한 정보를 담게 되면서, 기존의 기본적인 방식보다 데이터를 더 잘 이해하고 이에 더 좋은 추천해주는 알고리즘의 필요성 대두
- 기존 데이터 뿐만 아니라 이미지, 텍스트 등 다양한 데이터도 함께 활용

Neural Collaborative Filtering

Xiangnan He
National University of
Singapore, Singapore
xiangnanhe@gmail.com

Liqiang Nie
Shandong University
China
nieliqiang@gmail.com

Lizi Liao
National University of
Singapore, Singapore
liaolizi.llz@gmail.com

Xia Hu
Texas A&M University
USA
hu@cse.tamu.edu

Hanwang Zhang
Columbia University
USA
hanwangzhang@gmail.com

Tat-Seng Chua
National University of
Singapore, Singapore
dcscts@nus.edu.sg

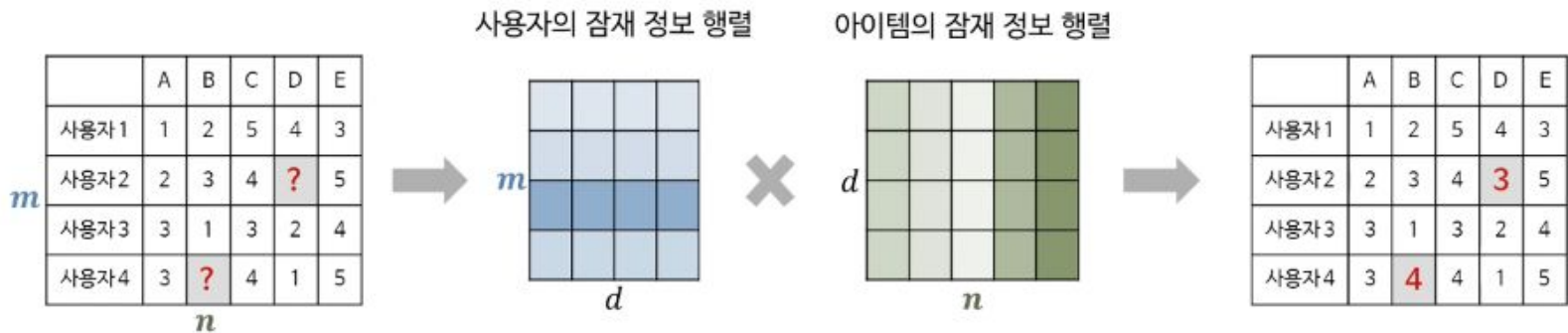
- 2017, International World Wide Web Conference (IWWW)
- Collaborative Filtering 의 대표적인 방법중 하나인 Matrix Factorization (MF) 의 한계를 지적. MF 의 문제점과 Neural Collaborative Filtering 이 어떻게 이런 문제를 해결하는지 소개함.

<https://arxiv.org/pdf/1708.05031.pdf>

Deep Learning based Recommender System

- Matrix Factorization 의 단점

- 사용자와 아이템에 대한 잠재 정보 행렬을 만든다는 것은 특정 2차원 공간으로 투영시켜 표현하는 것을 의미함
- 하지만, 새로운 사용자가 등장했을 때, 2차원 상에서 이를 적합하게 표현하는 것의 어려움이 있음.



*예시 : 4명의 사용자가 5개의 아이템에 대한 평점을 매긴 경우

Deep Learning based Recommender System

- Matrix Factorization 의 단점

- 사용자와 아이템에 대한 잠재 정보 행렬을 만든다는 것은 특정 2차원 공간으로 투영시켜 표현하는 것을 의미함
- 하지만, 새로운 사용자가 등장했을 때, 2차원 상에서 이를 적합하게 표현하는 것의 어려움이 있음.

	아이템 1	아이템 2	아이템 3	아이템 4	아이템 5
사용자 1	1	1	1	0	1
사용자 2	0	1	1	0	0
사용자 3	0	1	1	1	0

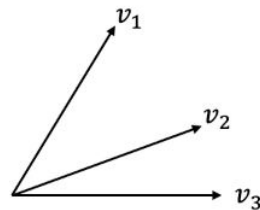


$$s_{12} = \frac{2}{4} = 0.5$$

$$s_{13} = \frac{2}{5} = 0.4$$



$$s_{23} = \frac{2}{3} = 0.666$$



* 예시 : 아이템 구매 여부를 0과 1로 표현한 데이터

- Jaccard Coefficient 를 활용하여 유사도 계산

- 사용자 2, 3 간의 유사도가 가장 높고
- 사용자 1, 3 간의 유사도가 가장 낮음.
- 이런 정보를 반영하여 2차원 벡터로 표현하면 오른쪽 그림과 같이 표현.

Deep Learning based Recommender System

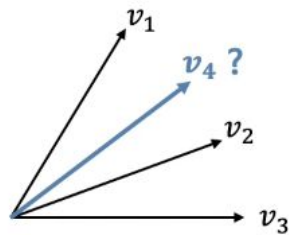
- Matrix Factorization 의 단점
 - 새로운 사용자가 등장했을 때, 다른 사용자들과의 유사도 정도를 2차원으로 표현하는데 한계가 있음.
 - 사용자 4, 1 이 가장 가깝게,
 - 사용자 4, 2 가 가장 멀게 표현하는 것에 어려움이 있음.

	아이템 1	아이템 2	아이템 3	아이템 4	아이템 5
사용자 1	1	1	1	0	1
사용자 2	0	1	1	0	0
사용자 3	0	1	1	1	0
사용자 4	1	0	1	1	1

$$s_{41} = \frac{3}{5} = 0.6$$

$$s_{42} = \frac{1}{5} = 0.2$$

$$s_{43} = \frac{2}{5} = 0.4$$

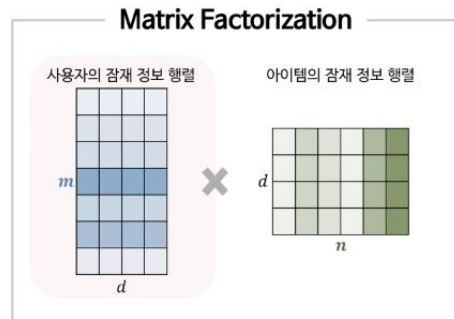
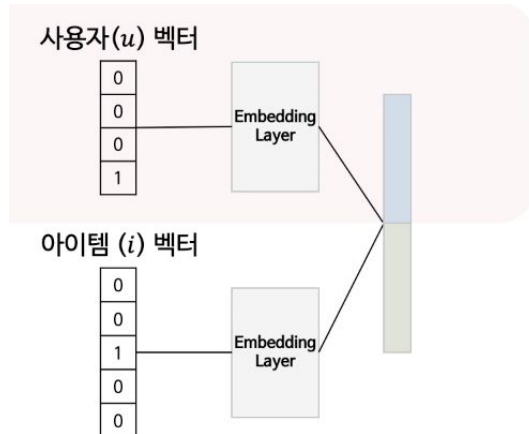
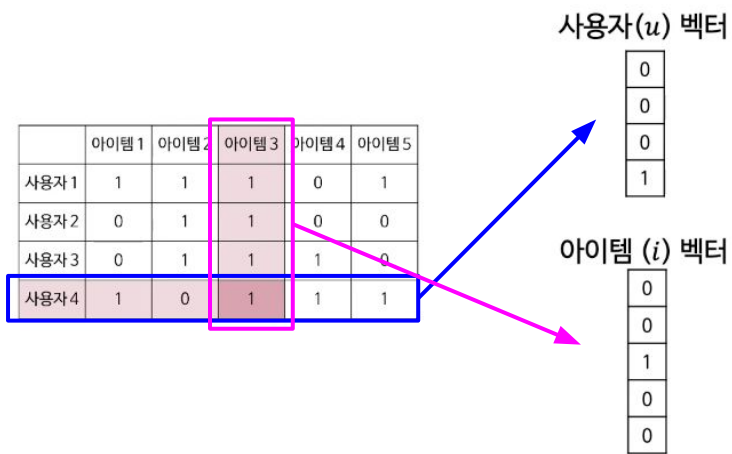


* 예시 : 아이템 구매 여부를 0과 1로 표현한 데이터

Deep Learning based Recommender System

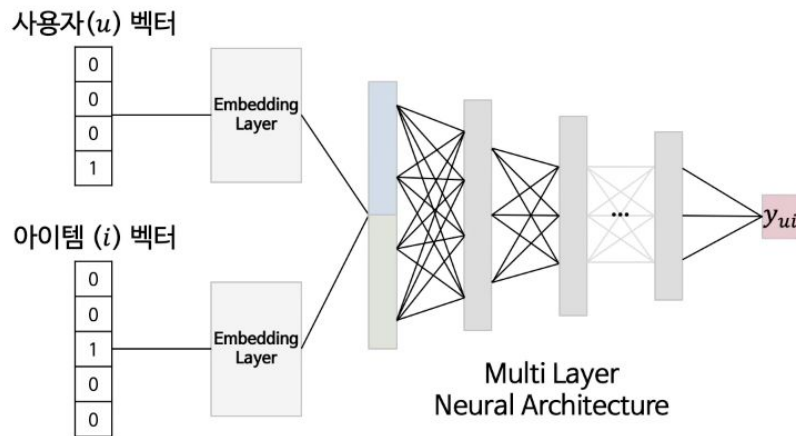
- Neural Collaborative Filtering

- 입력 데이터 : 사용자와 아이템에 대한 인풋 벡터
- 협력 필터링을 기반으로 하는 모델이기 때문에, 사용자와 아이템에 대한 정보만 인풋으로 활용



Deep Learning based Recommender System

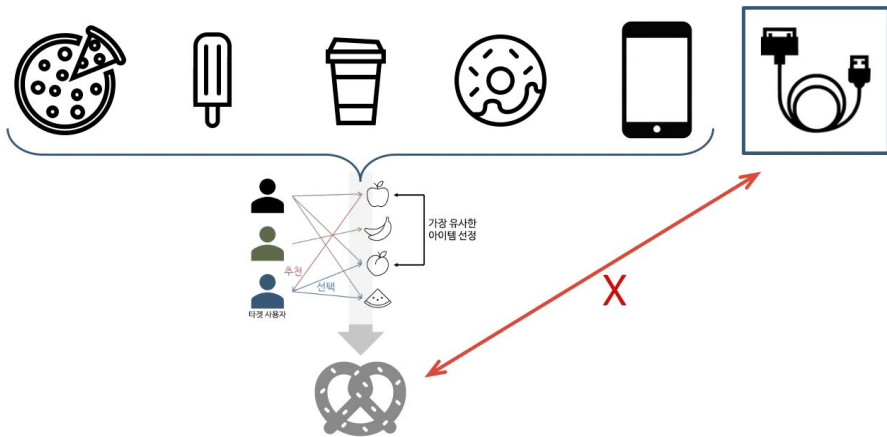
- Neural Collaborative Filtering
 - 입력 데이터 : 사용자와 아이템에 대한 인풋 벡터; 사용자 4와 아이템 3에 대한 input vector
 - 출력 데이터 : 사용자 4가 아이템 3을 선택한 여부 (확률값 출력)



- 딥러닝을 추천 시스템에 활용한 대표적인 사례
 - Multi Layer Perceptron + Collaborative Filtering
- 간단한 모델의 아이디어이지만, 이전 기본적인 모델을 활용하던 방법과 비교하였을 때 높은 성능을 보임.

Sequential Recommender System

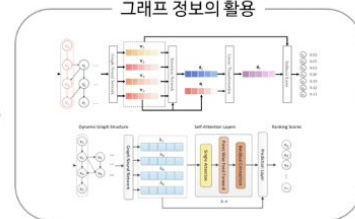
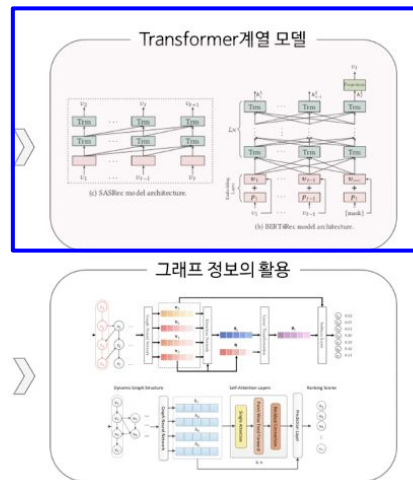
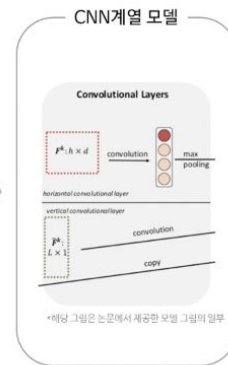
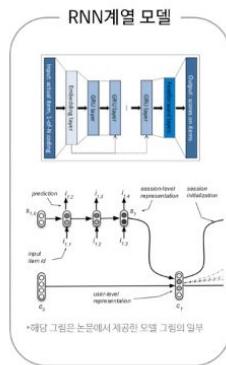
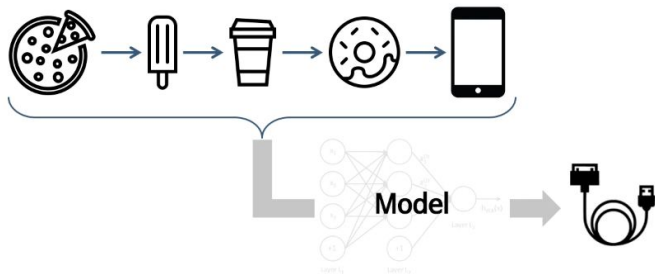
- 사용자의 과거 아이템 선택의 정보가 동일하게 중요하다는 기본 가정에서 출발한 추천 시스템 알고리즘
- 실제로 사용자가 선택을 할 때 과거 구매 정보가 동일하게 중요할까?



- 주로 선택하던 아이템과 다른 아이템이 등장할 경우 이에 대한 충분한 설명이 부족
- 협력 필터링 등 보편적으로 사용되는 추천시스템 모델에서는 사용자가 주로 선택하던 아이템을 추천하게 됨.

Sequential Recommender System

- 사용자의 선호도, 관심은 끊임없이 변화하고 발전한다는 아이디어에서 시작
- 조금 더 실제 사용자의 관심사를 잘 반영하는, 변화의 패턴까지도 잡아낼 수 있는 모델을 만들어보자
- 과거의 행동에서 유의한 순차적인 패턴을 찾고, 최근 아이টে에 더 집중하여 이를 기반으로 가장 선호할 다음 아이টে 추천
- 다양한 딥러닝 모델들을 활용한 추천시스템 방법론들이 제안됨
 - RNN 계열 모델
 - CNN 계열 모델
 - Transformer 계열 모델
 - 그래프 정보의 활용



Sequential Recommender System

- Transformer 계열의 Sequential Recommender System
- 2019, International Joint Conference on Artificial Intelligence

Feature-level Deeper Self-Attention Network for Sequential Recommendation

**Tingting Zhang^{1,2}, Pengpeng Zhao^{1,2*}, Yanchi Liu³, Victor S. Sheng⁴,
Jiajie Xu¹, Deqing Wang⁵, Guanfeng Liu⁶ and Xiaofang Zhou^{7,2}**

¹Institute of AI, School of Computer Science and Technology, Soochow University, China

²Zhejiang Lab, China

³Rutgers University, New Jersey, USA

⁴University of Central Arkansas, Conway, USA

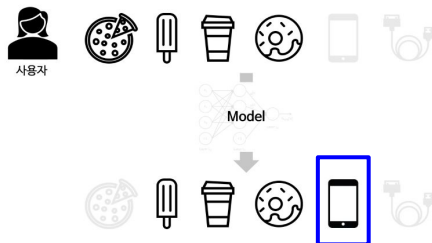
⁵School of Computer, Beihang University, Beijing, China

⁶Department of Computing, Macquarie University, Sydney, Australia

⁷The University of Queensland, Brisbane, Australia

<https://www.ijcai.org/proceedings/2019/0600.pdf>

Sequential Recommender System



- Input : 한 사용자의 순차적인 아이템 목록 중 앞 L 개의 아이템
- Output : 한 칸 미룬 후의 L 개 아이템
- 모델 아키텍처

Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19)

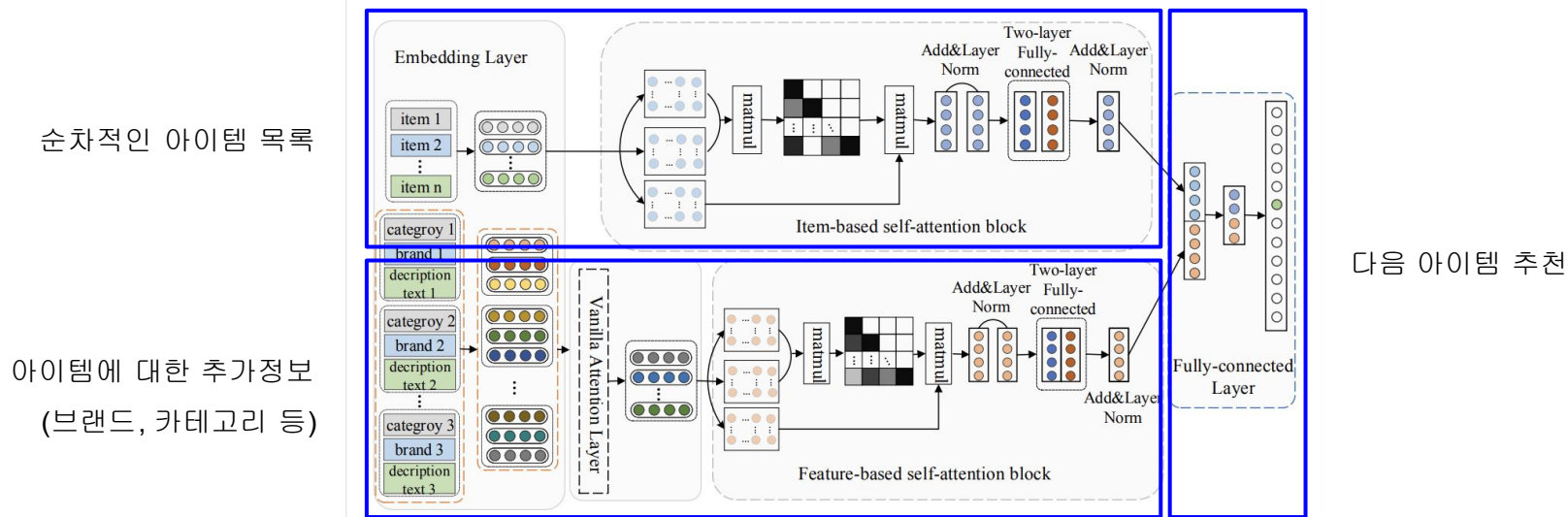
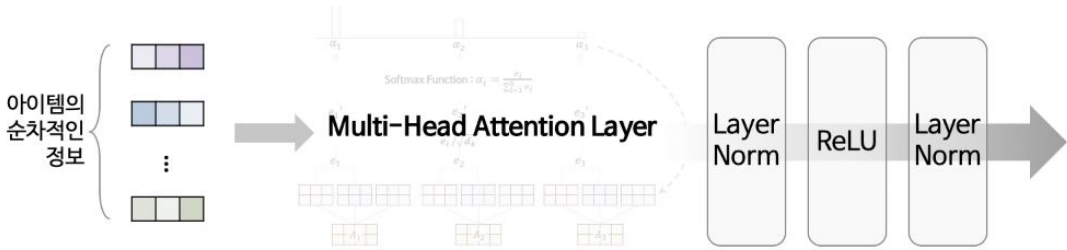
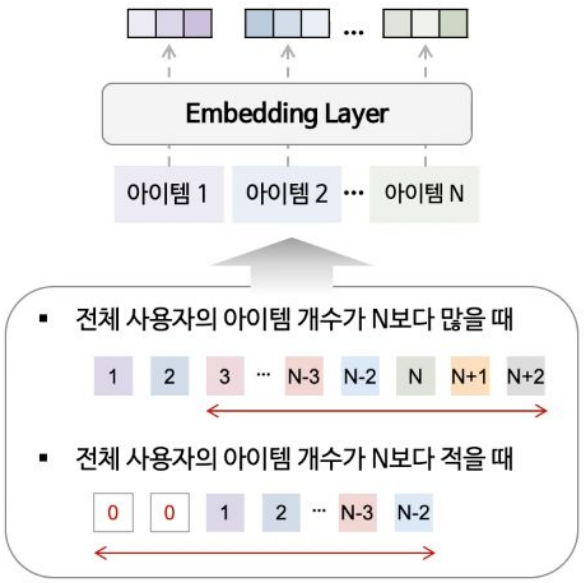
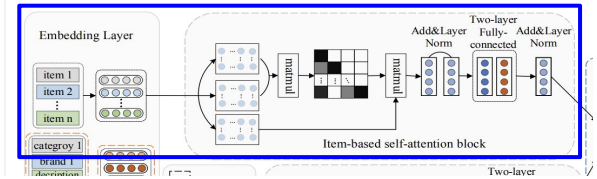


Figure 1: The Network Architecture of FDSA.

Sequential Recommender System

- 인풋으로 사용하기 위하여 순차적인 아이템 목록을 각각 위치정보를 반영한 벡터로



Sequential Recommender System

- 기존 모델에서는 주로 순차적인 아이템 정보만을 사용하여 추천
- 모델의 추가적인 정보를 제공함으로써 추천 성능을 높일 수 있는 구조 제안
- 사용자가 선택한 아이템의 부가적인 정보 (브랜드, 카테고리 등) 을 추가적으로 요약하여 활용
- 카테고리 정보, 브랜드 정보, 선택 아이템에 대한 텍스트 정보등을 한번에 결합하는 방법 제시
- 인풋으로 사용하기 위하여 모든 정보를 Dense Vector Representation 의 형태로 변환

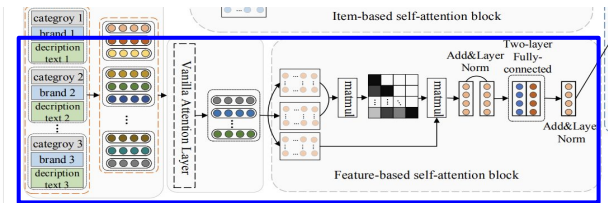
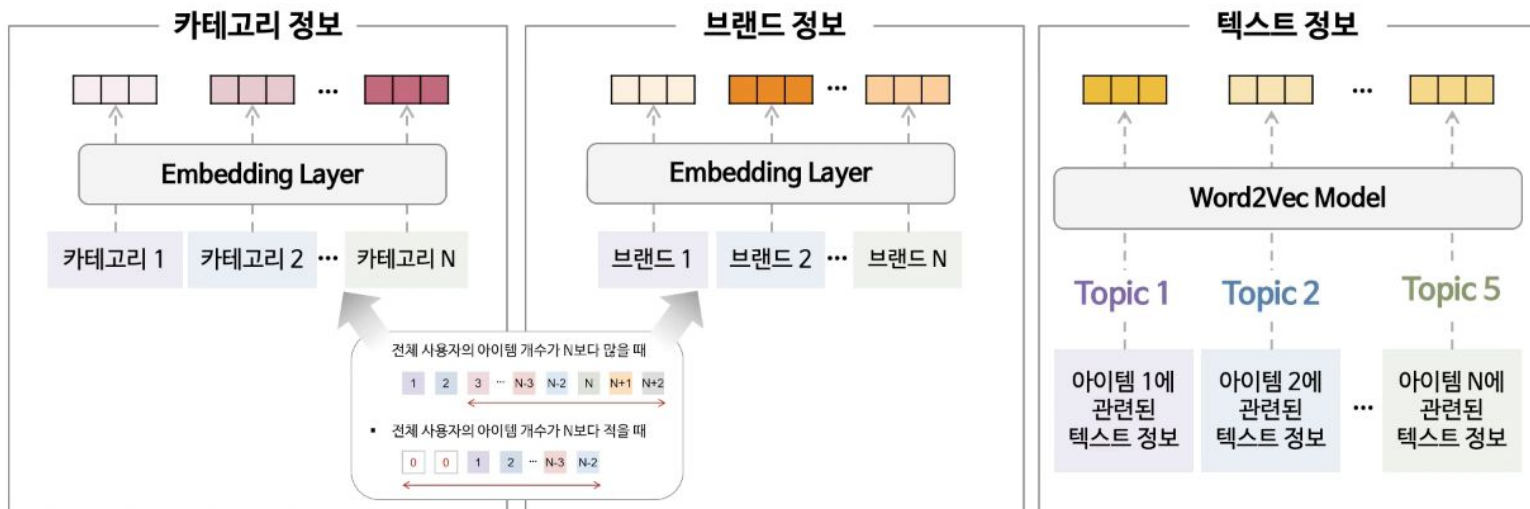
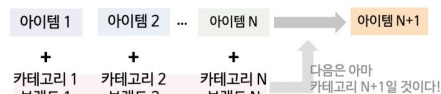


Figure 1: The Network Architecture of FDSA.



Sequential Recommender System

- 어떤 정보가 사용자의 선택에 영향을 미쳤는지 알아보기 위해, Vanilla Attention Layer
- 순차적인 정보는 보존한 채, 각 아이템 정보 간 추가적인 관계(정보)를 파악할 수 있는 Self-Attention Blocks 활용

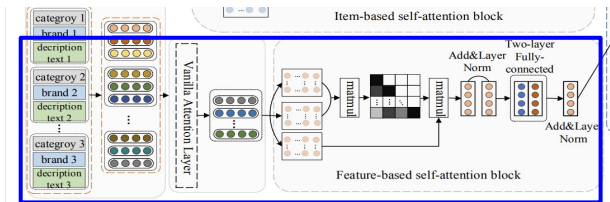
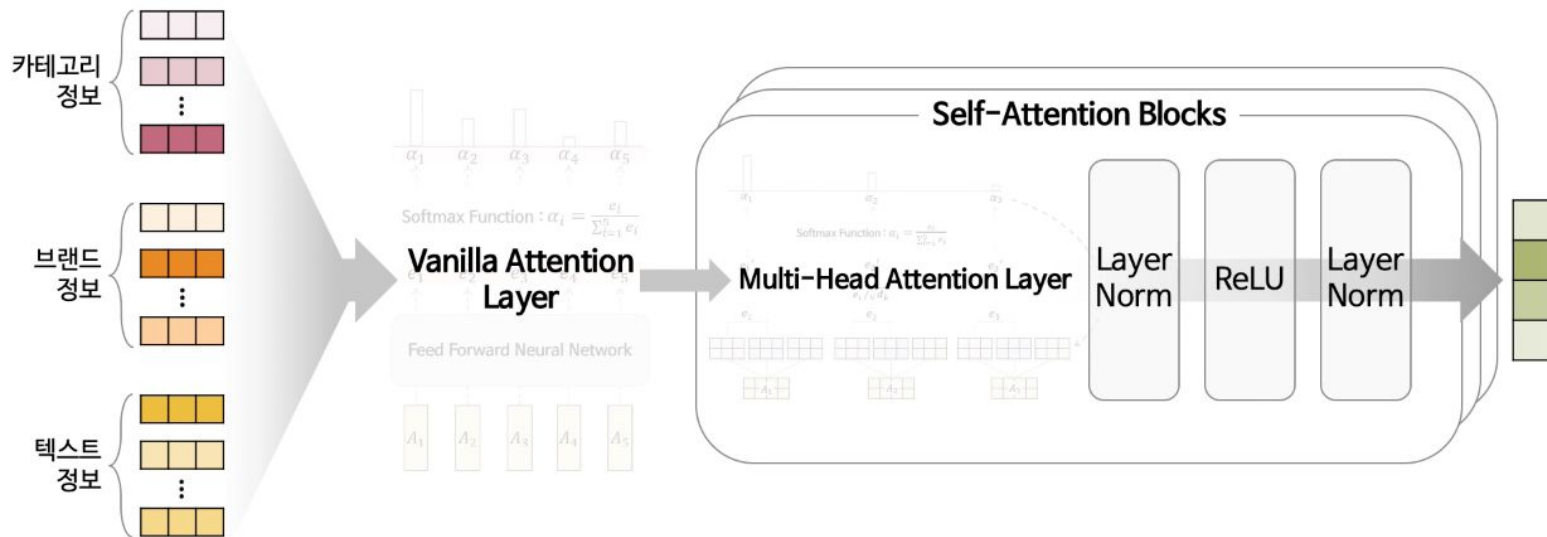


Figure 1: The Network Architecture of FDSA.



Sequential Recommender System

- 최종 결과값으로 인풋으로 들어온 다음 아이템 예측

Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19)

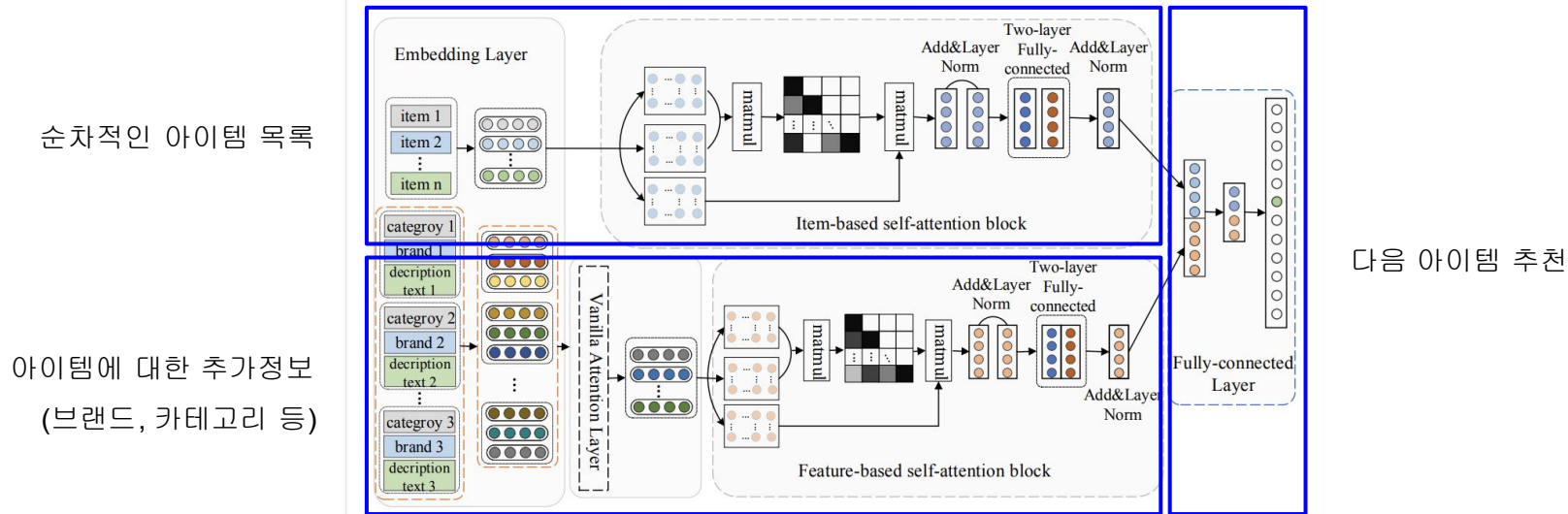


Figure 1: The Network Architecture of FDSA.

Sequential Recommender System

Experiment

- 이전에 제안된 다양한 순차적 추천시스템들보다 더 높은 성능을 보

Dataset	Method	@5		@10	
		Hit	NDCG	Hit	NDCG
Tmall	PopRec	0.1532	0.0988	0.2397	0.1267
	BPR	0.1749	0.1129	0.2647	0.1418
	FPMC	0.2731	0.2034	0.3680	0.2339
	TransRec	0.2652	0.1854	0.3773	0.2214
	GRU4Rec	0.1674	0.1217	0.2446	0.1465
	CSAN	0.3481	0.2440	0.4787	0.2863
	SASRec	0.3572	0.2531	0.4840	0.2940
	SASRec+	0.3427	0.2415	0.4714	0.2829
	SASRec++	0.3550	0.2534	0.4785	0.2932
	CFSA	0.3836	0.2724	0.5152	0.3149
	FDSA	0.3940	0.2820	0.5197	0.3226
Toys and Games	PopRec	0.1952	0.1287	0.3058	0.1643
	BPR	0.2096	0.1394	0.3219	0.1756
	FPMC	0.2983	0.2261	0.3833	0.2535
	TransRec	0.3135	0.2255	0.4206	0.2600
	GRU4Rec	0.2039	0.1359	0.3118	0.1705
	CSAN	0.2327	0.1601	0.3404	0.1947
	SASRec	0.3292	0.2334	0.4441	0.2705
	SASRec+	0.3367	0.2410	0.4510	0.2776
	SASRec++	0.3394	0.2428	0.4544	0.2799
	CFSA	0.3391	0.2411	0.4538	0.2782
	FDSA	0.3571	0.2572	0.4738	0.2949

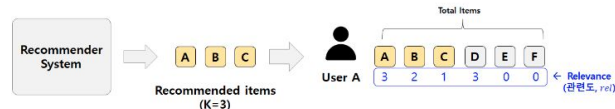
Table 2: Experimental results of FDSA and baselines. The best performance of each column (the larger is the better) is in bold.

1) Relevance란?

Relevance란, 사용자가 특정 아이템과 얼마나 관련이 있는지를 나타내는 값이다. Relevance값은 정해진 것이 아니고 추천의 상황에 맞게 정해야 한다. 예를 들어, 신발 추천인 경우 사용자가 해당 신발을 얼마나 클릭했는지, 혹은 클릭 여부(Binary) 등 다양한 방법으로 선정할 수 있다. [그림4]의 예시에서는 전체 아이템에 대한 Relevance를 파란색으로 표시해두었다.

3. NDCG@K (Normalized Discounted Cumulative Gain)

NDCG는 원래 검색 분야에서 등장한 지표이나 추천 시스템에도 많이 사용되고 있다. 위의 두 평가 지표와 마찬가지로 Top K개 아이템을 추천하는 경우, 추천 순서에 가중치를 두어 평가한다. **NDCG@K 값은 1에 가까울수록 좋다.** MAP는 사용자가 선호한 아이템이 추천 리스트 중 어떤 순서에 포함되었는지 여부에 대해서 1 or 0으로만 구분하지만, **NDCG@K는 순서별로 가중치 값(관련도, relevance)을 다르게 적용하여 계산한다.** [그림4]의 예시를 통해 하나씩 살펴보자.



$$CG_3 = \sum_{i=1}^K rel_i = rel_1 + rel_2 + rel_3 = 3 + 2 + 1 = 6$$

$$DCG_3 = \sum_{i=1}^K \frac{rel_i}{\log_2(i+1)} = \frac{3}{\log_2(1+1)} + \frac{2}{\log_2(2+1)} + \frac{1}{\log_2(3+1)} = \frac{3}{1} + \frac{2}{1.58} + \frac{1}{2} = 4.78$$

$$IDCG_3 = \sum_{i=1}^K \frac{rel_i^{opt}}{\log_2(i+1)} = \frac{3}{\log_2(1+1)} + \frac{3}{\log_2(2+1)} + \frac{2}{\log_2(3+1)} = \frac{3}{1} + \frac{3}{1.58} + \frac{2}{2} = 5.89$$

$$NDCG_3 = \frac{DCG}{IDCG} = \frac{4.78}{5.89} = 0.81$$

[그림5] NDCG@K 예시

Sequential Recommender System

- 결론
 - 기본적인 추천 알고리즘부터 순차적인 정보를 고려하는 추천 시스템 알고리즘까지 다양한 방법으로 발전
 - 다양한 머신러닝 및 딥러닝 알고리즘들이 발전되면서 해당 모델을 추천 시스템에 적용시키는 사례가 다양
 - **Feature-level Deeper SElf-Attention Network for Sequential Recommendation**
 - 사용자가 선택한 아이템의 순차적 정보와 그 아이템에 대한 세부적인 정보까지 반영한 방법론 제안
 - **Multi-Head Attention** 등의 기법을 활용하여 각 정보들을 요약하고, 이를 활용

Thank you :)