

# 딥러닝 기반 추천시스템 성능 비교 연구

추천시스템

황서진

# 목차

## 1 추천시스템

## 2 협업 필터링

BE 모형, SVD 모형, BPR 모형, NCF 모형, BiVAE 모형

## 3 추천시스템 평가 기준

평점 기반 평가, 순위 목록 기반 평가

## 4 연구방법

사용 데이터, 데이터 분할

## 5 연구결과

추천시스템 성능 비교, 평점 기반 평가, 순위 목록 기반 평가

# 1. 추천시스템

## 1. 콘텐츠 기반 필터링

- a. 항목의 정보와 사용자 관심사를 기반으로 사용자에게 요소를 추천하는 것

## 2. 협업 필터링

- a. 구매, 소비한 제품에 대한 각 사용자의 평가를 받아서 평가 패턴이 비슷한 소비자 (사용자)를 한 집단으로 보고 그 집단에 속한 취향을 활용하여 추천하는 것

## 3. 지식 기반 필터링 접근 방식

- a. 해당 분야 전문가의 도움을 받아 전체적인 지식구조를 만들어 활용하여 추천하는 방식

## 4. 하이브리드 접근 방식

- a. 두 개 이상의 필터링을 조합해서 활용하는 방법
- b. 보통 협업 필터링과 콘텐츠 기반 필터링을 결합해서 사용

## 2. 협업 필터링

- 다양한 사용자들부터의 정보를 활용하여 유사한 패턴을 가진 또 다른 사용자의 구매 형태를 교차 추천하는 알고리즘
- 제품에 대한 특징을 알 수 없을 때도 사용할 수 있고 사용자들의 성격, 취향 등이 명확하게 구분되는 제품을 추천할 때 더욱 정확
- 콜드 스타트 문제 : 이전 구매이력이 없는 신규 사용자의 경우 사용자들 간의 유사한 패턴을 구할 수 없기 때문
  - ⇒ 소비 이력이 없는 신규 소비자를 대상으로 선호하는 취향 파악을 위한 설문조사 방법, 하이브리드 접근 방식을 사용하는 방법

## 2.1 BE 모형

일반적인 협업 필터링 데이터는 사용자와 항목 평가에서 사용자의 성향에 따라 다른 사용자보다 더 높은 등급을 주고 일부 항목은 다른 사용자가 평가한 등급에 비해 더 높은 등급을 받는 경향이 많음  
⇒ 기준 추정값(BE)를 구하여 이 효과를 제거해야함

$$r_{ui} = \mu + a_u + b_i + \epsilon_{ui},$$

BE 모형

## 2.2 SVD 모형

- MF (matrix factorization) 모형

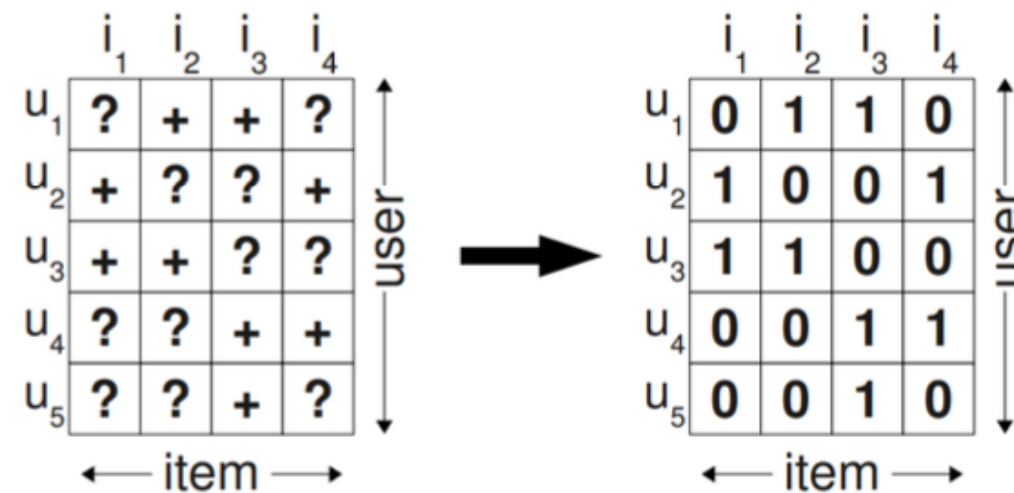
기존

- SVD를 명시적인 데이터에 적용할 경우 결측값이 많아져 분석하기 어렵고 모델이 과적합 되는  
단점
- 대체에 의존
- 데이터가 상당히 왜곡됨

최근

적절한 정규화된 모델을 통해 과적합을 피하면서 관찰된 항목 값을 직접 모델 구축하는 것을  
목표

## 2.3 BPR 모형



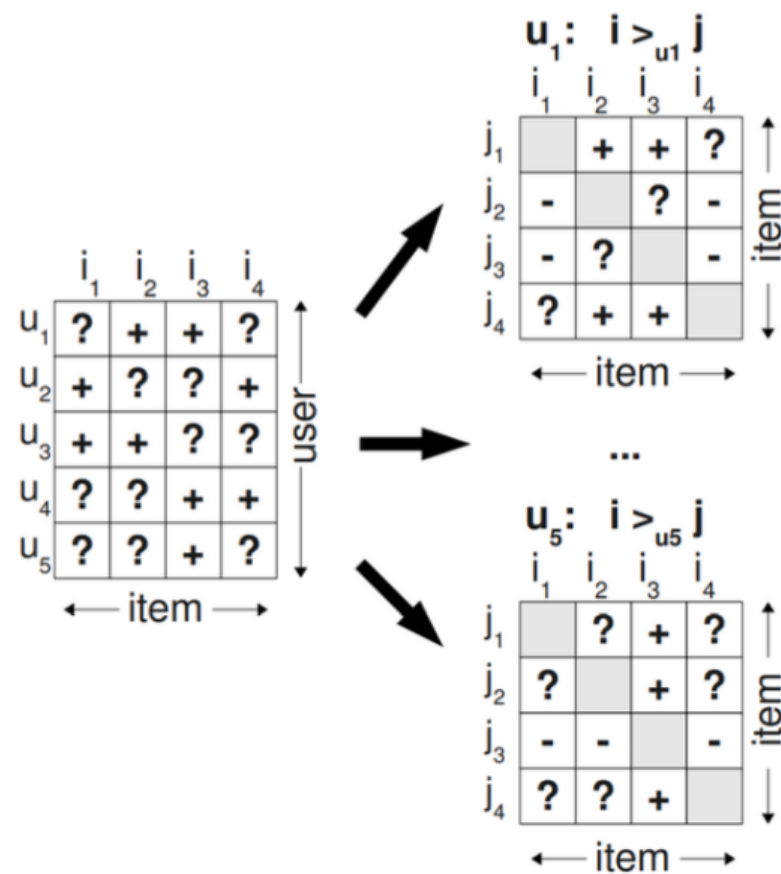
**Figure 2.1** Implicit feedback of usual collaborative filtering (Rendle *et al.*, 2012)

기존 협업 필터링의 암묵적 피드백

왼쪽 : 관찰된 데이터 행렬  $S$

사용자가 응답한 값만 관찰되기 때문에  $S$ 를 직접 학습하는 것은 불가능하기 때문에 응답하지 않은 값은 0으로 채워 행렬 생성

## 2.3 BPR 모형



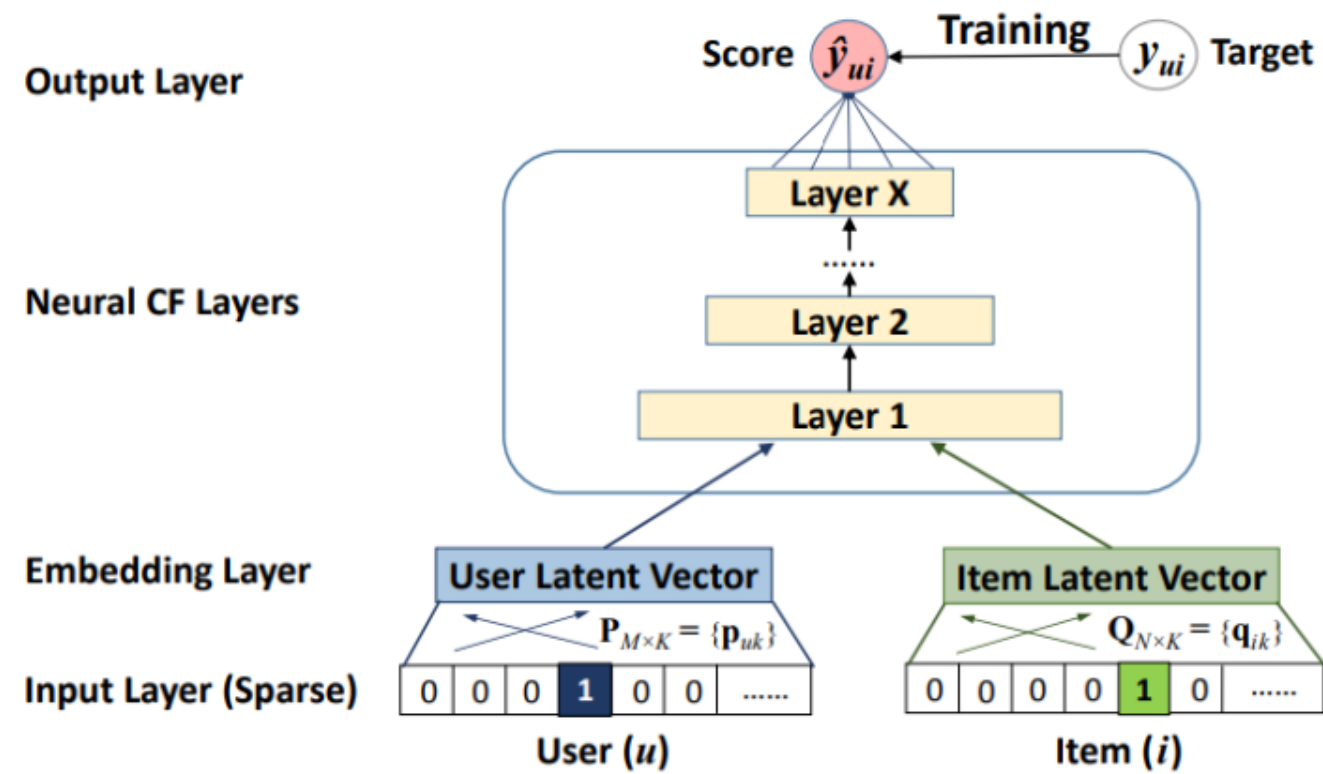
1. 사용자의 선호도를 반영하여 항목에 대한 개인화된 점수를 예측하는 것
2. 해당 점수에 따라 항목을 정렬하여 순위를 매김
3. 순위 최적화를 위하여 항목  $i$ 와  $j$ 의 쌍  $(i,j)$ 를 이용하여 사용자  $u$ 의 선호도를 학습
4. 사용자-항목의 차원으로 구성된 평점 행렬  $S$ 를 재구성

⇒ 항목  $i$ 와 항목  $j$ 의 항목 쌍을 이용하여, 항목  $i$ 가 더 상위 순위에 나타나는지를 확인하는 모형

**Figure 2.2** Pairwise preference method of Bayesian personalized ranking (Rendle *et al.*, 2012)



## 2.4. NCF 모형



NCF = MF의 선형성 + MLP의 비선형

**Figure 2.3** Structure of neural collaborative filtering (He *et al.*, 2017)

## 2.5 BiVAE 모형

- 사용자와 항목을 대칭적으로 처리
- 양방향 혹은 이원형 데이터에 더 적합
- 쌍방향 데이터의 양쪽에서 불확실성을 포착할 수 있어 기존의 VAE에 비해 희소 선호도 데이터에 대한 성능을 향상 가능

# 3.1 추천시스템 평가 기준

## 평점 기반 평가

사용자가 항목에 평가한 예측 등급에 대한 정확도를 평가하는 것

추천시스템으로부터 예측값에서 상위  $k$ 의 항목을 선별하여 1위부터  $k$ 위까지의 순위 목록을 구성하고, 이를 검증 자료에 존재하는 상호작용과 대조하여 평가하는 방법

## 순위 목록 기반 평가

- (1) 추천시스템에서 사용자별로 모든 항목에 대한 평점 예측값이나 순위 예측값을 반환받는다.
- (2) 이미 상호작용이 이루어진 (학습 자료에 포함된) 항목의 예측값을 제외한다.
- (3) 평점 (등급)을 반환하는 모형의 경우 사용자별 평점 예측값  $\hat{r}_{ui}$  상위  $k$ 개 항목을 추출하고, 순위를 반환하는 모형의 경우 사용자별 항목들의 순위 예측값 상위  $k$ 개 항목을 추출한다.
- (4) 사용자별 순위 목록을 검증 자료와 대조하여 평가를 진행하고, 전체 사용자에게 대한 평균 지표를 반환한다.

# 4. 연구방법

## 4.1 사용 데이터

캐글에서 제공된 IMDbb 자료

- 평점을 1부터 5까지 정수 단위로 분석에 사용
- 사용자가 항목에 대해 10개 이상 평가된 경우만 분석에 포함

## 4.2 데이터 분할

6:2:2 비율로 훈련 데이터, 검증 데이터, 시험 데이터로 구성

데이터 분할 방법

- 무작위분할 방법
- 훈련, 검증, 시험을 시간순서로 분할하는 시간고려분할방법

Table 4.1 Characteristics of the datasets	
Count	100836
features name	userID, itemID, rating, timestamp
Rating scale (mean, sd)	1-5 (3.65, 1.05)
unique of user	610
unique of item	9724
Count of missing value	0

# 5. 연구결과

## 5.1 추천 시스템 성능 비교

Table 5.2 Tuning parameters selected from two sampling methods		
Partition method	Algorithm	Tuning parameters
Random	BE	$\gamma$ : 0.005, $\lambda$ : 0.026, n_epoch: 20
	SVD	$F$ : 163, $\gamma$ : 0.021, $\lambda$ : 0.031, n_epoch: 19
	BPR	$F$ : 250, $\alpha$ : 0.059, $\lambda_\theta$ : 0.001, max_iter: 287
	NCF	Model type: NeuMF, n_factors: 9, layer_sizes: [16, 8, 4], batch_size: 1024, learning_rate: 0.004, n_epochs: 15, verbose: 10
	BiVAE	$F$ : 100, $\alpha$ : 0.002, encoder_structure: 200, act_fn: tanh, likelihood: pois, batch_size: 256, n_epochs: 529
Time-dependent	BE	$\gamma$ : 0.005, $\lambda$ : 0.030, n_epoch: 20
	SVD	$F$ : 211, $\gamma$ : 0.066, $\lambda$ : 0.030, n_epoch: 37
	BPR	$F$ : 350, $\alpha$ : 0.090, $\lambda_\theta$ : 0.001, max_iter: 236
	NCF	Model type: NeuMF, n_factors: 4, layer_sizes: [16, 8, 4], batch_size: 1024, learning_rate: 0.003, n_epochs: 27, verbose: 21
	BiVAE	$F$ : 100, $\alpha$ : 0.001, encoder_structure: 200, act_fn: tanh, likelihood: pois, batch_size: 512, n_epochs: 466

- 두 가지 추출 방법 각각에 대한 Optuna에 의해 선정된 튜닝 파라미터

# 5. 연구결과

## 평점 기반 평가

Table 5.3 Results of rating-based evaluation

Partition method	Algorithm	<i>RMSE</i>	<i>MAE</i>	$R^2$	<i>EVS</i>
Random	BE	0.830	0.648	0.373	0.373
	SVD	0.886	0.687	0.285	0.285
Time-dependent	BE	0.792	0.609	0.454	0.454
	SVD	0.880	0.679	0.326	0.326

두 가지 분할 모형에서  
모두 BE 모형이 SVD 모형보다 성능이 더 좋게 나옴

# 5. 연구결과

## 순위 목록 기반 평가

Table 5.4 Results of ranking-based evaluation

Partition method	<i>k</i>	Algorithm	<i>RECALL@k</i>	<i>PRECISION@k</i>	<i>MAP@k</i>	<i>NDCG@k</i>
Random	5	BE	0.016	0.089	0.010	0.096
		SVD	0.010	0.054	0.006	0.061
		NCF	0.052	0.206	0.032	0.207
		BPR	0.080	0.274	0.054	0.288
		BiVAE	0.098	0.340	0.068	0.355
	10	BE	0.033	0.080	0.013	0.091
		SVD	0.017	0.045	0.007	0.053
		NCF	0.086	0.165	0.042	0.185
		BPR	0.131	0.234	0.073	0.271
		BiVAE	0.159	0.287	0.095	0.330
Time-dependent	5	BE	0.017	0.067	0.009	0.071
		SVD	0.003	0.013	0.002	0.014
		NCF	0.027	0.089	0.015	0.095
		BPR	0.030	0.101	0.016	0.104
		BiVAE	0.035	0.118	0.019	0.118
	10	BE	0.033	0.058	0.012	0.068
		SVD	0.007	0.013	0.003	0.015
		NCF	0.050	0.084	0.022	0.094
		BPR	0.061	0.099	0.023	0.109
		BiVAE	0.067	0.109	0.028	0.120

### 무작위분할 방법

- 전체적으로 BPR 모형과 BiVAE 모형이 우수한 성능
- BE 모형, SVD 모형이 성능이 좋지 않은 이유는 선호 순위에 대한 예측을 목표로하는 모형이 아니기 때문
- NCF, BPR, BiVAE 모형의 경우 순위나 상호작용의 발생 가능성을 예측하는 것을 목표로 하는 모형이기 때문에 순위 예측에 좀 더 유리

### 시간고려분할방법

- 무작위분할 방법과 비슷한 순서
- 하지만 NCF, BPR, BIVAE 모형의 성능은 무작위분할 방법보다 전체적으로 많이 떨어짐