論文読み会：*“A Systematic Review and Replicability Study of BERT4Rec for Sequential Recommendation”*, Recsys 2022

宮本 隆志

ナビプラス株式会社

2023/06/28

紹介する論文

SASRec と BERT4Rec

過去の論文の調査

実験

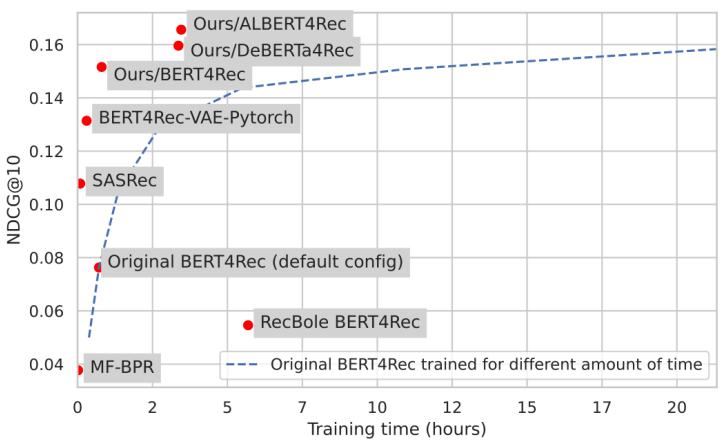
結果

まとめ

紹介する論文

"A Systematic Review and Replicability Study of BERT4Rec for Sequential Recommendation" (Recsys'22)

- Glasgow 大学の論文
- Sequential recommendation の話
- BERT4Rec は sequential recommendation の SOTA でベースラインとして使われてる。しかしいろんな論文で性能に整合性が無い。そこで BERT4Rec について調査した。
  - BERT4 Rec の 4 実装、ベースラインに SASRec, MF-BPR を MovieLens-1M で比較した。
  - 青い点線はオリジナルの BERT4Rec を訓練時間を変えて評価したもの。
- 我々の実装は性能高い。また DeBERTa, ALBERT ベースの方が性能高い。

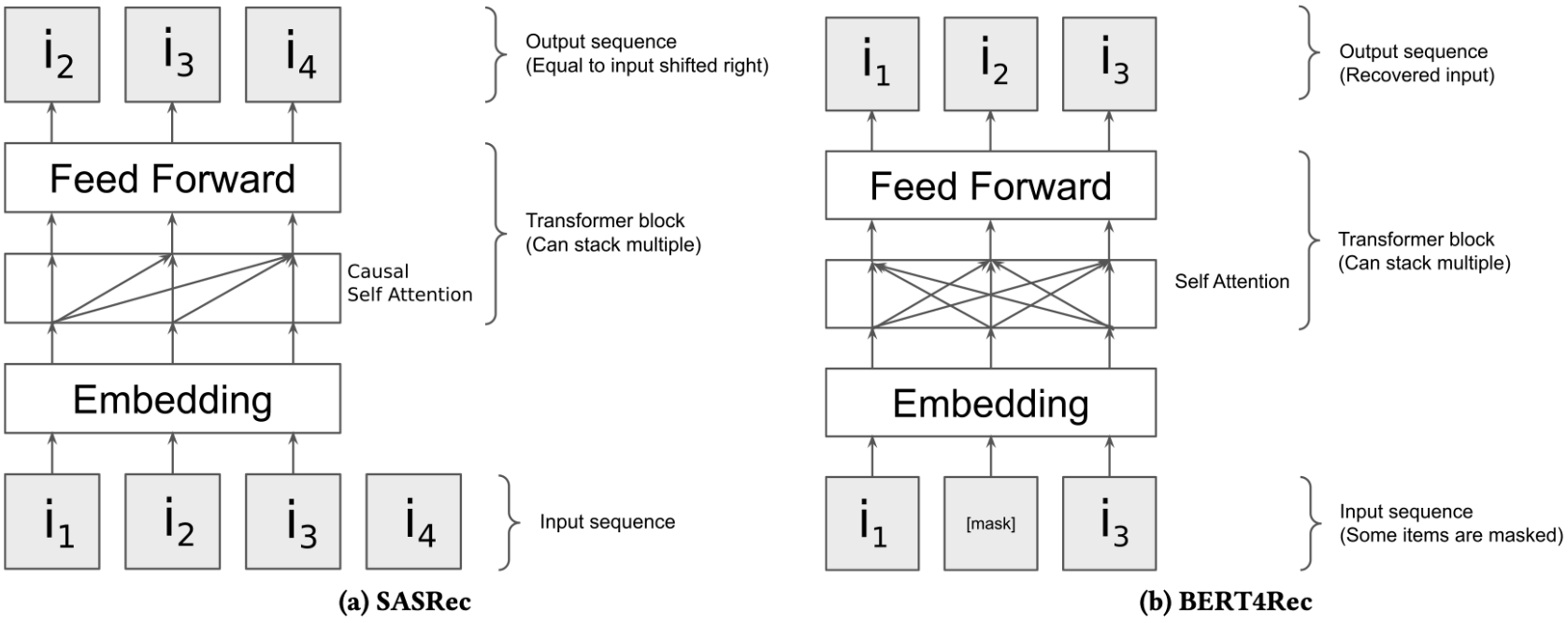


SASRec と BERT4Rec

- どちらも Transformer ベースだが、違いは Self Attention のところ。

SASRec causal (unidirectional). 次のアイテムを予測することが目的。

BERT4Rec regular (bidirectional). mask されたアイテムを当てることが目的。



過去の論文の調査

- "**H1** Overall, BERT4Rec is not systematically better than SASRec in the published literature."
- BERT4Rec に言及した論文 370 本 → SASRec と比較してるものは 58 → peer-review されてる論文が 40。
- 46 dataset で、134 の比較。
- 優劣に再現性がない。実装とかパラメタ調整とかの差なのか？

dataset	total	BERT4Rec wins	SASRec wins	Ties	BERT4Rec wins papers	SASRec wins papers	Ties papers
Beauty* [16]	19	**12 (63%)**	5 (26%)	2 (11%)	[52], [40], [68], [38], [29], [7], [64], [55], [62], [18], [60], [53]	[69], [32], [65], [58], [47]	[3], [9]
ML-1M* [14]	18	**13 (72%)**	3 (17%)	2 (11%)	[52], [40], [68], [29], [7], [23], [55], [18], [60], [51], [42], [46], [28]	[12], [62], [47]	[64], [9]
Yelp [2]	10	**6 (60%)**	4 (40%)	0 (0%)	[69], [1], [3], [58], [47], [42]	[12], [32], [65], [33]	
Steam* [43]	8	**7 (88%)**	1 (12%)	0 (0%)	[52], [40], [68], [29], [64], [9], [60]	[62]	
ML-20M* [14]	8	**7 (88%)**	0 (0%)	1 (12%)	[52], [40], [66], [29], [7], [60], [46]		[9]
Sports [16]	6	1 (17%)	**4 (67%)**	1 (17%)	[28]	[69], [32], [65], [47]	[3]
LastFM [5]	6	**4 (67%)**	2 (33%)	0 (0%)	[69], [23], [55], [28]	[1], [65]	
Toys [16]	5	0 (0%)	**5 (100%)**	0 (0%)		[69], [13], [32], [65], [3]	
Total	134	**86 (64%)**	32 (23 %)	16 (12 %)			

実験

実装

MF-BPR matrix factorization w/ pairwise BPR loss. LightFM library, #latent factor=128.

SASRec `https://github.com/kang205/SASRec` . seq.length=50, embed.size=50, #transformer block=2

BERT4Rec 表を参照。我々の実装は Hugging Face Transformers library 使用。ハイパーパラメタは Table 4 参照。

評価指標 Leave-One-Out で、NDCG と Recall@K。

sampled metrics positive item 1 つにつき、100 個の negative item を選んで評価。negative sampleing は popularity-based。問題の多い評価方法だが元論文との比較のため。

unsampled metrics 全アイテムが各ユーザ毎に比較される。

Implementation	GitHub URL	Framework	GitHub stars	Example papers
Original	FeiSun/BERT4Rec	Tensorflow v1	390	[21, 22, 37]
RecBole	RUCAIBox/RecBole	PyTorch	1,800	[3, 12, 31]
BERT4Rec-VAE	jaywonchung/BERT4Rec-VAE-Pytorch	PyTorch	183	[48, 56, 63]
Ours/Hugging Face	asash/bert4rec_repro	Tensorflow v2	N/A	N/A

結果

- 一部のみ掲載。シーケンス長は、ML-1M が 165.5、Steam は 12.4 とかなり違う。
- 我々の実装が優れてる。(なお計算時間は A6000 使って評価してる)
- 元の BERT4Rec は報告されてる訓練時間では報告値の性能が出ない。

(a) ML-1M Dataset

	Model	Popularity-sampled		Unsampled		Training Time
		Recall@10	NDCG@10	Recall@10	NDCG@10	
Baselines	MF-BPR	0.5134 (-26.34%)†	0.2736 (-43.21%)†	0.0740†	0.0377†	58
	SASRec	0.6370 (-8.61%)†	0.4033 (-16.29%)†	0.1993†	0.1078†	316
BERT4Rec versions	Original	0.5215 (-25,18%)†	0.3042 (-36.86%)†	0.1518†	0.0806†	2,665
	RecBole	0.4562 (-34.55%)†	0.2589† (-46.26%)†	0.1061†	0.0546†	20,499
	BERT4Rec-VAE	**0.6698 (-3.90%)†**	0.4533 (-5.29%)†	0.2394†	0.1314†	1,085
	Ours	**0.6865 (-1.51%)**	**0.4602 (-4.48%)**	0.2584	0.1392	3,679
	Ours (longer seq)	**0.6975 (+0.07%)**	**0.4751 (-1.39%)**	0.2821	0.1516	2,889
Reported [52]	BERT4Rec	0.6970	0.4818	N/A	N/A	N/A

(b) Steam Dataset

	Model	Popularity-sampled		Unsampled		Training Time
		Recall@10	NDCG@10	Recall@10	NDCG@10	
Baselines	MF-BPR	0.3466 (-13.63%)†	0.1842 (-18.53%)†	0.0398†	0.0207†	162
	SASRec	0.3744 (-6.70%)†	0.2052 (-9.24%)†	0.1198†	0.0482†	3,614
BERT4Rec versions	Original	0.2148 (-46.47%)†	0.1064 (-52.94%)†	0.0737†	0.0375†	4,847
	RecBole	0.2325 (-42.06%)†	0.1177 (-47.94)†	0.0744†	0.0377†	83,816
	BERT4Rec-VAE	0.3520 (-12.29%)†	0.1941 (-14.15%)†	0.1237†	0.0526†	65,303
	Ours	**0.3978 (-0.87%)**	**0.2219 (-1.86%)**	0.1361	0.0734	117,651
Reported [52]	BERT4Rec	0.4013	0.2261	N/A	N/A	N/A

Model	Recall@10	NDCG@10	Training Time
BERT4Rec	0.282	0.151	2,889
DeBERTa4Rec	0.290 (+3.0%)	0.159 (+2.3%)	12,114
ALBERT4Rec	**0.300 (+6.4%)†**	**0.165 (+9.2%)†**	12,453

Publication	BERT4Rec Recall@10	Best model	Best model Recall@10
This paper	**0.282 (+0.0%)**	ALBERT4Rec	**0.300 (+6.4%)**
Dallmann et al. [9]	0.160 (-43.2%)	GRU4Rec+	0.224 (-20.5%)
Qiu et al. [47]	0.132 (-53.1%)	DuoRec	0.294 (+4.4%)
Fan et al. [12]	0.221 (-21.6%)	LightSANs	0.228 (-19.0%)
Liu et al. [35]	0.252 (-10.5%)	NOVA-BERT	0.286 (+1.5%)

まとめ

- SASRec vs BERT4Rec 論争が存在するが、SASRec より BERT 系の方が性能良いと分かった。
- 我々の実装が性能良い。RecBole 遅い。データセット間で計算時間の大小が違いすぎるので優劣はなんとも。
- 感想
  - SASRec は性能悪いんだけど、速くて実案件では使いやすそうな気もする・・・
  - 一回のカートアイテム数が多い EC ショップの場合、userid の無い顧客でも、sequence と見て推薦出来るので、cold-start 問題に対応出来る可能性はある。
  - ただ、それなら sequental recommendation じゃなく、バスケット分析で良いのでは、という気も。