Rethinking Sequential Relationships: Improving Sequential Recommenders with Inter-Sequence Data Augmentation

Yang Jiao, Fan Yang, Yetian Chen, Yan Gao, Yi Sun
{jaoyan,fnam,yetichen,yanngao,yisun}@amazon.com
Amazon
Seattle, WA, USA

Jia Liu
The Ohio State University
Department of Electrical and Computer Engineering
Columbus, OH, USA
liu.1736@osu.edu

ABSTRACT

Predicting customer preferences for each item is a prerequisite module for most recommender systems in e-commerce. However, the sparsity of behavioral data is often a challenge to learn accurate prediction models. Given millions of items, each customer may only be able to interact with a small subset of them over time. This sparse behavioral data is insufficient to represent item-customer and item-item relations for a machine learning model to digest, resulting in limited prediction accuracy that hinders recommendation performance. To mitigate this issue, this study introduces an inter-sequence data augmentation method, SDA_{inter}, that enhances data density by leveraging cross-customer behavioral patterns to enrich item relations. Tested on three public and one proprietary e-commerce dataset, SDA_{inter} significantly increases data density, leading to notable improvements in both evaluation and business metrics. Our findings demonstrate SDA_{inter}'s effectiveness and its potential to complement existing data augmentation strategies in recommender systems. See https://github.com/ML-apollo/SDA_inter.

KEYWORDS

Sequential recommendation, personalization, transformer, data augmentation, cold-start.

ACM Reference Format:
Yang Jiao, Fan Yang, Yetian Chen, Yan Gao, Yi Sun and Jia Liu. 2024. Rethinking Sequential Relationships: Improving Sequential Recommenders with Inter-Sequence Data Augmentation. In *Companion Proceedings of the ACM Web Conference 2024 (WWW '24 Companion), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3589335.3651552

1 INTRODUCTION

Recommender systems play a pivotal role in guiding users to their desired items across various platforms, including e-commerce, streaming services, and social networks. Among these, Sequential Recommenders (SR) stand out by adeptly predicting user preferences through analyzing patterns in item interactions over time. It has been proposed to mine frequent patterns to guide the recommendation [19]. This approach, which takes a user's interaction history to forecast future engagements, has evolved significantly with the advent of deep learning. Notably, Transformer models [2, 4, 10, 14, 16, 17] have enhanced SRs by adeptly handling complex, dynamic user-item relationships (e.g., higher-order and important) through mechanisms like self-attention [15], enabling a nuanced understanding of both sequential [4] and bidirectional [14] item interactions. Such technological advancements not only refine recommendation accuracy but also enrich the user experience, marking a significant leap in how digital platforms anticipate and meet user needs.

Developing and refining models to accurately predict item-item interaction transitions of each customer poses a significant challenge in the realm of recommender systems, primarily due to data sparsity in item-customer interactions. With e-commerce platforms offering hundreds of millions of items, customers typically engage with only a fraction, leading to extremely low interaction densities. For instance, the interaction density that measures the item-customer interaction in the 'Online' e-commerce dataset is a mere 0.04%, as shown in Table 1. Such sparsity hampers the model's ability to learn from limited data, affecting the accuracy of recommendations and the system's capacity to understand nuanced user preferences.

There have been works dedicate to mitigating the data sparisity problems from different perspectives. For example, a dual contrastive network (DCN) [6] is proposed to integrate auxiliary user sequence and capture the preference of different user types. MoHR and MT4SR[2, 5, 7] utilize heterogeneous item behavioral relationships. The second approach is to augment sequential data by introducing artificially generated interaction sequences that contain important but less frequent/unseen item-item transitions. Among the second approach, CL4SRec [18] proposed three random data augmentation methods (*cropping*, *masking* and *reordering*), CoSeRec [8] introduced two informative augmentation operators (*substitute* and *insert*) leveraging item correlation, and CCL [1] developed a model-based data generator with users' attribute information. ASRep [9] predicts the prior items of sequences to extend short sequences. DuoRec[13] proposed a model-level augmentation based on dropout

Table 1: Datasets Density after Augmentation. Beauty, Tool and Sports are Amazon review datasets, Online is real data from online e-commerce system.

Dataset	Beauty	Tool	Sports	Online
# user	22, 363	16, 638	35, 598	$9.76M$
# item	12, 101	10, 217	18, 357	48, 642
# interaction w.o. SDA_{inter}	$0.20M$	$0.13M$	$0.30M$	$185M$
density w.o. SDA_{inter}	0.07%	0.08%	0.05%	0.04%
# interaction w. SDA_{inter}	$1.87M$	$0.37M$	$1.05M$	$888M$
density w. SDA_{inter}	0.69%	0.22%	0.16%	0.19%

WWW '24 Companion, May 13–17, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-0172-6/24/05.
https://doi.org/10.1145/3589335.3651552

to enable better semantic representations. However, most of these existing works focused on intra-sequence augmentations, such as masking [1, 8], reordering [8], cropping [8], and reversing [3, 9]. However, intra-sequence data augmentation only leverages item relationships of the same customer, while the relationships across customers are not utilized. It can only increases item-item transition diversity but cannot increase data density.

In this work, we explore a novel interchangeability rule among sequences from different customers and introduce an inter-sequence data augmentation method, SDA_{inter}. SDA_{inter} searches for subsequence pairs that meet the interchangeability rule in a set of customers' historical interactions. By exchanging the subsequences within a pair, SDA_{inter} builds pseudo item-customer interaction sequences from original sequences. This approach significantly enhances the density of item-customer interaction, facilitating the learning process for sequential transformers, as evidenced by our experimental results shown in Table 1.

In summary, our major contributions are as follows:

- We propose a innovative data augmentation method called SDA_{inter}, which leverages cross-user sequential relations to effectively improve data density in sequential recommendation systems
- Our experiments on four datasets show that SDA_{inter} markedly improves key performance indicators, including NDCG, Recall, Price-weighted purchases, and the view-to-purchase conversion rate. Our findings highlight SDA_{inter}'s effectiveness in refining the sequential recommendations through enriched data.
- Our ablation studies show that SDA_{inter} has ability to complement existing strategies for mitigating data sparsity.

2 METHOD

2.1 Problem Formulation

Consider a set of user sequences $\{S_u\}_{u=1}^{U}$, where U is the total number of users and $S_u = [i_u^0, i_u^1, ..., i_u^{N_u}]$ is a sequence of items of length N_u. Let $\mathcal{I}$ be the set of all items. Our goal is to predict the distribution of next interacted item $p(i^{N_u+1} = i|S_{N_u})$, where item $i \in \mathcal{I}$. The idea of sequential recommendation is to model the next interaction distribution:

$$p(i^{N_u+1}|S_{N_u}) = f(S_{N_u}, \theta),$$

where f is a vector-valued function that is implemented by different machine learning methods with parameters θ.

In next section, we present an inter-sequence augmentation method to enrich S_{N_u} to improve the data sparsity problem and produce more accurate next-interaction predictions.

2.2 Inter-sequence Data Augmentation

In this subsection, we introduce the inter-sequence data augmentation operation that can discover more item interactions and relations via the interchangeability between sequences. For two user sequences $S_a = [i_a^0, i_a^1, ..., i_a^{N_a}]$ and $S_b = [i_b^0, i_b^1, ..., i_b^{N_b}]$ for users a and b, $S_a, S_b \in \{S_u\}_{u=1}^{U}$. Considering two pairs of anchor items $(i_a^{n_1}, i_a^{n_2})$ and $(i_b^{m_1}, i_b^{m_2})$ from S_a and S_b respectively, $i_a^{n_1}, i_a^{n_2} \in S_a$, $i_b^{m_1}, i_b^{m_2} \in S_{b2}$, $n_2 > n_1$, and $m_2 > m_1$. Let the anchor items be the starting and ending transitions for two subsequences between the anchor items in S_a and S_b, i.e., $S_{a_s} = [i_a^{n_1}, ..., i_a^{n_2}]$ and $S_{b_s} = [i_b^{m_1}, ..., i_b^{m_2}]$. If the anchor items $i_a^{n_1} = i_b^{m_1}$ and $i_a^{n_2} = i_b^{m_2}$, we say that there exists interchangeability between S_{a_s} and S_{b_s}. Here, we set an intersection-over-union (IoU) threshold T_c to ensure the interchangeability confidence. The confidence $C = (S_{a_s} \cap S_{b_s})/(S_{a_s} \cup S_{b_s})$. When $C >= T_c$, the interchangeability is established. T_c is set to 0.2 in our experiments. By exchanging the two subsequences with interchangeability in S_a and S_b, we generate two new pseudo user sequences as follows:

$$S_{a'} = [i_a^0, i_a^1, ..., i_a^{n_1-1}, i_b^{m_1}, ..., i_b^{m_2}, i_a^{n_2+1}, ..., i_a^{N_a}],$$
$$S_{b'} = [i_b^0, i_b^1, ..., i_b^{m_1-1}, i_a^{n_1}, ..., i_a^{n_2}, i_b^{m_2+1}, ..., i_b^{N_b}].$$

Interchangeability could also exist in multiple subsequences in a user sequence. A user sequence S_a could have p_a anchor item pairs in total that is denoted as a set Ω_a, where each pair is in the form of $(i_a^{n_1}, i_a^{n_2})$, $0 < n_1 < n_2 < N_a$. Each item pair in Ω_a is associated with q_a candidate sequences denoted as a set Υ_a, where each candidate sequence shares an anchor item pair with S_a and has the interchangeability with S_a. Hence, the total interchangeable candidates for user sequence S_a is $p_a \times q_a$. Algorithm 1 summarizes the inter-sequence data augmentation operation.

Algorithm 1 Inter-sequence Data Augmentation.

Input: user sequence S_a, candidate anchor pair set Ω_a, candidate sequence set Υ_a, IoU threshold T_c
Output: pseudo user sequences set Φ_a
for (i^{o_1}, i^{o_2}) in Ω_a **do**
  for S_b in Υ_a **do**
    Sequence $S_a = [i_a^0, i_a^1, ..., i_a^{N_a}]$ and the candidate sequence $S_b = [i_b^0, i_b^1, ..., i_b^{N_b}]$
    if i^{o_1} and i^{o_2} in S_a and S_b
      Given anchors in S_a and S_b:
      S_a: $i^{o_1} = i_a^{n_1}$ and $i^{o_2} = i_a^{n_2}$, and S_b: $i^{o_1} = i_b^{m_1}$ and $i^{o_2} = i_b^{m_2}$
      Extract the interchangeable subsequences:
      $S_{a_s} = [i_a^{n_1}, ..., i_a^{n_2}]$, $S_{b_s} = [i_b^{m_1}, ..., i_b^{m_2}]$
      The interchangibity confidence:
      $C = (S_{a_s} \cap S_{b_s})/(S_{a_s} \cup S_{b_s})$
      if $C >= T_c$
        Construct pseudo user sequences S_a:
        $S'_a = [i_a^0, i_a^1, ..., i_a^{n_1-1}, i_b^{m_1}, ..., i_b^{m_2}, i_a^{n_2+1}, ..., i_a^{N_a}]$
        Pseudo user sequence S_b:
        $S'_b = [i_b^0, i_b^1, ..., i_b^{m_1-1}, i_a^{n_1}, ..., i_a^{n_2}, i_b^{m_2+1}, ..., i_b^{N_b}]$
        Add S'_a and S'_b to Φ_a
      end if
    end if
  end for
end for

2.3 Flip-flop Training of Sequential Transformer

Given a user sequence $S_u = [i_u^0, i_u^1, ..., i_u^{N_u}]$, a popular approach [2, 18] for training a sequential transformer uses $[i_u^0, i_u^1, ..., i_u^{N_u-2}]$ to train, item $i_u^{N_u-1}$ as the validation target, and item $i_u^{N_u}$ as the test target. Since S is sequential, if the data distribution is time-varying (due to multiple reasons such as item availability and trend changing), a sequential transformer trained on the forward sequence $[i_u^0, i_u^1, ..., i_u^{N_u-2}]$ could have difficulty predicting items arriving in future. While some works [14] propose bi-directional sequential

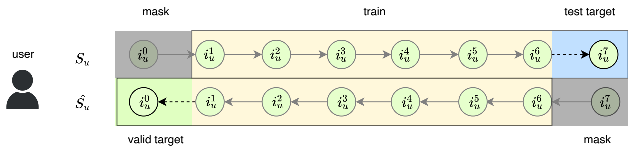

Figure 1: Flip-flop training of sequential transformer.

modeling, it is infeasible to use bi-directional information to predict the next item. To address the limitations of one-directional learning, in this section, we propose to train sequential transformers in a *flip-flop* fashion so that sequential transformers are able to learn various types of data distribution trends. Figure 1 illustrates an example of flip-flop training. For user sequence S_u, we flip the sequence as $\hat{S_u} = [i_u^{N_u}, i_u^{N_u-1}, ..., i_u^0]$. With S_u and $\hat{S_u}$, we use the subsequences $[i_u^1, ..., i_u^{N_u-1}]$ and $[i_u^{N_u-1}, ..., i_u^1]$ to train, $i_u^0 \in \hat{S_u}$ to validate, and $i_u^{N_u} \in S_u$ to test. To avoid data leakage, we mask $i_u^0 \in S_u$ and $i_u^{N_u} \in \hat{S_u}$.

3 EXPERIMENTS

In this section, we conduct extensive experiments and ablations studies to answer the following questions:

- Q1: What is the performance of sequential transformers with SDA_{inter} compared to the state-of-the-art baselines in sequential recommendation task?
- Q2: How does inter-sequence data augmentation complement with intra-sequence augmentation?
- Q3: what is the contribution of each component?
- Q4: What is the impact of different item popularity on target?

3.1 Experiment Settings

1) Datasets: To demonstrate the efficacy of our method, we use three popular public datasets [11, 12], including *Amazon Beauty*, *Amazon Sports and Outdoors*, and *Amazon Tools and Home Improvement*. The dataset details are shown in Table 1. We follow the data preprocessing approach in [2], and use the five-core settings by filtering out users with less than five interactions. Additionally, We construct an online e-commerce dataset to compare the performance of all models on real industrial application. The large-scale dataset "Online" (in Table 1) collects a year of customer purchase records from an online shopping platform. The target is to predict the latest purchase of each customer based on historical purchases.

2) Baselines: We compare our work with the following baselines on the sequential recommendation task. SASRec [4] and BERT4Rec [14] are classic sequential transformers for sequential recommendation tasks. CL4REC [18] and DuoRec [13] are the state-of-the-art works using data augmentation to boost the sequential recommendation. MoHR [5] and MT4SR [2] are the state-of-the-art works that model the auxiliary item relationship with sequential transitions.

3) Evaluation: We follow the evaluation task in [2] and rank all items for model performance comparisons. We use popular evaluation metrics NDCG@N and Recall@N. Recall@N evaluates the existence of ground truth positive in top-N ranked items, and NDCG@N measures the ranking position of the positive in top-N ranked items. Higher metric values mean better model performances. We report $N = 5, 10$, which is the same as in [2, 4, 14]. For fair comparisons, we only use the augmented pseudo user sequences

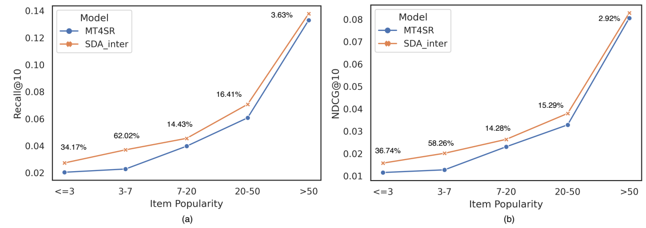

Figure 2: Recall@10 and NDCG@10 performance on differenct item popularity. Dataset: Sports.

in training, but keep the test samples the same as in previous works [2, 13, 18].

4) Implementation: To demonstrate the benefits of SDA_{inter}, we implement SDA_{inter} with the best-performing baseline on each dataset. To show the complementarity of the inter- and intra-sequence data augmentation, we implement the intra-sequence (masking, cropping and reordering), denoted as SDA_{intra}, and apply two augmentations simultaneously, denoted as hybrid sequential data augmentation SDA_{hybrid}.

3.2 Results

3.2.1 *Overall performance comparison (Q1, Q2).*

In Table 2, we find the best-performing baseline model ($MT4SR$) and apply the proposed data augmentations. SDA_{inter} boosts performance on all three public datasets when applying SDA_{inter} on $MT4SR$. These results show that the pseudo user sequences constructed by SDA_{inter} improve the sequential modeling with transformers. To confirm the data distribution change, we analyze the existence of test 2-grams in training sequences. To answer Q2 and demonstrate the complementarity between inter-sequence data augmentation and other sparsity solutions, we compare using SDA_{inter} only and jointly with intra-sequence data augmentation. SDA_{hybrid} consistently achieves the best performance. Take the Beauty dataset as an example, compared to SDA_{inter}, on NDCG@5, SDA_{hybrid} achieves additional 4.89% improvement that shows the complementarity of intra- and inter-sequence augmentations.

3.2.2 *Key component ablation study (Q3).*

In Table 3, we reveal the contribution from each component of the propose method. Specifically, we experiment the efficacy of each component by adding it to the baseline model. Taking Beauty dataset as an example, each component contributes independently. The best performer SDA_{hybrid} composes all three components.

3.2.3 *Improvement on cold-start items (Q4).*

To study the benefit of augmentation on cold-start items, we group items into five groups based on popularity. In Figure 2, we observe SDA_{inter} outperforms the best baseline in all popularity groups. For low-popularity groups, SDA_{inter} demonstrates large improvements. This result shows that SDA_{inter} can address the problem of cold-start items and improve the recommendation for cold-start items.

3.2.4 *Performance on online platform.*

To study the method performance on real industrial system, we build new ranking features between customer and product using the proposed method. We report product ranking NDCG by measuring the rank of true purchases in a list products. The rank cutoffs is at 10. We also examine the business impact of the new feature with Price Weighted Purchases (PWP) and view-to-purchase conversion rate (CR) with the

Table 2: Overall performance comparison. The best results are bold, and the best baseline is underlined. The improvement is against the best baseline.

Model	Beauty				Tools				Sports			
	Recall@5	Recall@10	NDCG@5	NDCG@10	Recall@5	Recall@10	NDCG@5	NDCG@10	Recall@5	Recall@10	NDCG@5	NDCG@10
SASRec	0.0416	0.0633	0.0274	0.0343	0.0284	0.0427	0.0194	0.0240	0.0218	0.0336	0.0127	0.0169
BERT4Rec	0.0396	0.0595	0.0257	0.0321	0.0189	0.0319	0.0123	0.0165	0.0176	0.0326	0.0105	0.0153
CL4REC	0.0401	0.0683	0.0223	0.0317	0.0229	0.0381	0.0143	0.0192	0.0227	0.0374	0.0129	0.0184
DuoRec	0.0546	0.0845	0.0352	0.0443	0.0318	0.0490	0.0197	0.0252	0.0326	0.0498	0.0208	0.0262
MoHR	0.0529	0.0829	0.0349	0.0445	0.0481	0.0697	0.0340	0.0409	0.0321	0.0516	0.0206	0.0256
MT4SR	<u>0.0579</u>	<u>0.0859</u>	<u>0.0390</u>	<u>0.0480</u>	<u>0.0536</u>	<u>0.0751</u>	<u>0.0379</u>	<u>0.0449</u>	<u>0.0368</u>	<u>0.0518</u>	<u>0.0252</u>	<u>0.0300</u>
+SDA_{inter}	0.0634	0.0915	0.0437	0.0527	0.0557	0.0771	0.0394	0.0462	0.0395	0.0571	0.0267	0.0324
Improvement	9.51%	6.51%	11.96%	9.84%	3.95%	2.68%	3.84%	2.94%	7.49%	10.08%	6.22%	7.90%
+SDA_{hybrid}	**0.0663**	**0.0945**	**0.0456**	**0.0546**	**0.0570**	**0.0780**	**0.0406**	**0.0473**	**0.0399**	**0.0575**	**0.0272**	**0.0329**
Improvement	14.53%	10.05%	16.85%	13.81%	6.30%	3.80%	7.06%	5.33%	8.48%	10.95%	8.06%	9.45%

same cutoffs (i.e., PWP@10 and CR@10). Compared with the feature built with SASRec, the features built with various augmentation contribute more to online ranking system. From Table 4, we observe both SDA_{intra} and SDA_{inter} contribute independently on all business metrics. The hybrid augmentation additionally boosts the business metrics.

4 CONCLUSION

In this paper, we introduced an inter-sequence data augmentation method called SDA_{inter} to address the data sparsity problem for sequential recommendations. SDA_{inter} leverages the interchangeability between sequences and improves the item transitions density. Our experimental results on three public datasets and one online e-commerce dataset showed that SDA_{inter} achieved the state-of-the-art performance on sequential recommendation with a significant improvement. Our ablation study of intra- and inter-sequence data augmentations showed that inter-sequence data augmentation complements existing data sparsity solutions. An analysis further demonstrated that the sequential recommendation of cold-start items is largely improved by SDA_{inter}.

Table 3: Ablation study of inter-sequence data augmentation and flip-flop training. Dataset: Amazon Beauty

	Component	Recall@5	Recall@10	NDCG@5	NDCG@10
0	MT4SR	0.0579	0.0859	0.0390	0.0480
0+1	$+ flip-flop$	0.0614	0.0879	0.0420	0.0505
0+2	$+ inter-seq$	0.0593	0.0872	0.0405	0.0495
0+3	$+ intra-seq$	0.0603	0.0879	0.0403	0.0492
0+1+2	+ SDA_{inter}	0.0634	0.0915	0.0437	0.0527
0+1+2+3	+ SDA_{hybrid}	0.0663	0.0945	0.0456	0.0546

Table 4: Performance with feature built using SASRec with different augmentations. Metric: relative feature contribution. Dataset: Online.

Model	NDCG@10	PWP@10	CR@10
SASRec	+0.53%	+0.38%	+1.14%
+ SDA_{intra}	+0.65%	+0.56%	+1.28%
+ SDA_{inter}	+0.63%	+0.54%	+1.24%
+ SDA_{hybrid}	**+0.97%**	**+0.90%**	**+1.36%**

REFERENCES

[1] Shuqing Bian, Wayne Xin Zhao, Kun Zhou, Jing Cai, Yancheng He, Cunxiang Yin, and Ji-Rong Wen. 2021. Contrastive curriculum learning for sequential user behavior modeling via data augmentation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 3737–3746.

[2] Ziwei Fan, Zhiwei Liu, Chen Wang, Peijie Huang, Hao Peng, and S Yu Philip. 2022. Sequential Recommendation with Auxiliary Item Relationships via Multi-Relational Transformer. In *2022 IEEE International Conference on Big Data (Big Data)*. IEEE, 525–534.

[3] Juyong Jiang, Yingtao Luo, Jae Boum Kim, Kai Zhang, and Sunghun Kim. 2021. Sequential recommendation with bidirectional chronological augmentation of transformer. *arXiv preprint arXiv:2112.06460* (2021).

[4] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*. IEEE, 197–206.

[5] Wang-Cheng Kang, Mengting Wan, and Julian McAuley. 2018. Recommendation through mixtures of heterogeneous item relationships. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 1143–1152.

[6] Guanyu Lin, Chen Gao, Yinfeng Li, Yu Zheng, Zhiheng Li, Depeng Jin, and Yong Li. 2022. Dual contrastive network for sequential recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. 2686–2691.

[7] Feng Liu, Weiwen Liu, Xutao Li, and Yunming Ye. 2020. Inter-sequence enhanced framework for personalized sequential recommendation. *arXiv preprint arXiv:2004.12118* (2020).

[8] Zhiwei Liu, Yongjun Chen, Jia Li, Philip S Yu, Julian McAuley, and Caiming Xiong. 2021. Contrastive self-supervised sequential recommendation with robust augmentation. *arXiv preprint arXiv:2108.06479* (2021).

[9] Zhiwei Liu, Ziwei Fan, Yu Wang, and Philip S Yu. 2021. Augmenting sequential recommendation with pseudo-prior items via reversely pre-training transformer. In *Proceedings of the 44th international ACM SIGIR conference on Research and development in information retrieval*. 1608–1612.

[10] Jianxin Ma, Chang Zhou, Hongxia Yang, Peng Cui, Xin Wang, and Wenwu Zhu. 2020. Disentangled self-supervision in sequential recommenders. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 483–491.

[11] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*. 43–52.

[12] Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing*. 188–197.

[13] Ruihong Qiu, Zi Huang, Hongzhi Yin, and Zijian Wang. 2022. Contrastive learning for representation degeneration problem in sequential recommendation. In *Proceedings of the fifteenth ACM international conference on web search and data mining*. 813–823.

[14] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1441–1450.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[16] Jibang Wu, Renqin Cai, and Hongning Wang. 2020. Déjà vu: A contextualized temporal attention mechanism for sequential recommendation. In *Proceedings of The Web Conference 2020*. 2199–2209.

[17] Liwei Wu, Shuqing Li, Cho-Jui Hsieh, and James Sharpnack. 2020. SSE-PT: Sequential recommendation via personalized transformer. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 328–337.

[18] Xu Xie, Fei Sun, Zhaoyang Liu, Shiwen Wu, Jinyang Gao, Jiandong Zhang, Bolin Ding, and Bin Cui. 2022. Contrastive learning for sequential recommendation. In *2022 IEEE 38th international conference on data engineering*. IEEE, 1259–1273.

[19] Ghim-Eng Yap, Xiao-Li Li, and Philip S Yu. 2012. Effective next-items recommendation via personalized sequential pattern mining. In *Database Systems for Advanced Applications: 17th International Conference, DASFAA 2012, Busan, South Korea, April 15-19, 2012, Proceedings, Part II 17*. Springer, 48–64.