A Generic Behavior-Aware Data Augmentation Framework for Sequential Recommendation

Jing Xiao, Weike Pan* and Zhong Ming

xiaojing2022@email.szu.edu.cn, {panweike, mingz}@szu.edu.cn

College of Computer Science and Software Engineering,
Shenzhen University, Shenzhen, China

Problem Definition

Multi-Behavior Sequential Recommendation (MBSR)

- Input: Each user $u \in \mathcal{U}$ is associated with an interaction sequence of (item, behavior) pairs $\mathcal{S}_u = \{(v_u^1, b_u^1), \cdots, (v_u^\ell, b_u^\ell), \cdots, (v_u^L, b_u^L)\}$, where $v_u^\ell \in \mathcal{V}$ denotes the ℓ-th item interacted by user u with behavior $b_u^\ell \in \mathcal{B}$.
- Goal: Predict the most likely-to-purchase item of a user u from $\mathcal{V}$ at time step $L+1$, i.e., v_u^{L+1}, which can be formulated as follows:

$$\arg\max_{v_u^i \in \mathcal{V}} P\left(v_u^{L+1} = v_u^i \mid \mathcal{S}_u\right) \tag{1}$$

Related Work

- Data Augmentation for Recommendation
  - CL4SRec [Xie et al., 2022] proposes three augmentation operations, i.e., crop, mask and reorder. It treats the augmented sequences constructed from the same sequence as positive pairs.
  - DuoRec [Qiu et al., 2022] constructs positive and negative samples through both unsupervised and supervised approaches.
  - CASR [Wang et al., 2021] first pre-trains a sampling model and an anchor model, and then constructs some counterfactual sequences to re-optimize the anchor model via the sampling model.
  - ASRep [Liu et al., 2021], BiCAT [Jiang et al., 2021] rely on Transformer to reversely generate new items to address the issue of short sequences.
  - RSS [Petrov and Macdonald, 2022] considers that every item in a sequence should be possible as the target item. It designs a recency-based sequence sampling strategy to reconstruct the training samples, representing a completely model-agnostic approach.

Limitations of Existing Works (1/2)

- Existing methods only focus on a single type of behavior, neglecting the correlations between multi-type behaviors in user-item interactions, which has twofold drawbacks.
  - Models that extend the original sequences through pre-training (e.g., CASR and ASRep) tend to generate items deviating from user preferences.
  - Failure to differentiate behavior types when constructing new samples may introduce excessive noise (e.g., CL4SRec).

Limitations of Existing Works (2/2)

- Enhancing training samples with some pre-trained models often relies on a specific network architecture, such as Transformer or GNN (e.g., BiCAT and GraphDA), which makes it challenging to seamlessly transfer to other network structures.
- Models like S^3-Rec rely on some additional auxiliary information while limited data fail to provide sufficient support.

Overall of Our Solution

- We propose a novel multi-behavior data augmentation model for sequential recommendation, namely MBASR for short.
- We propose three behavior-aware data augmentation methods, including order perturbation, redundancy reduction, and pairwise swapping, to generate new sequences. These methods can be seamlessly integrated into various existing MBSR models to boost their performance.
- We design a position-based sampling strategy that can maximally preserve the smooth sequentiality between neighboring items in the original sequences.

MBASR

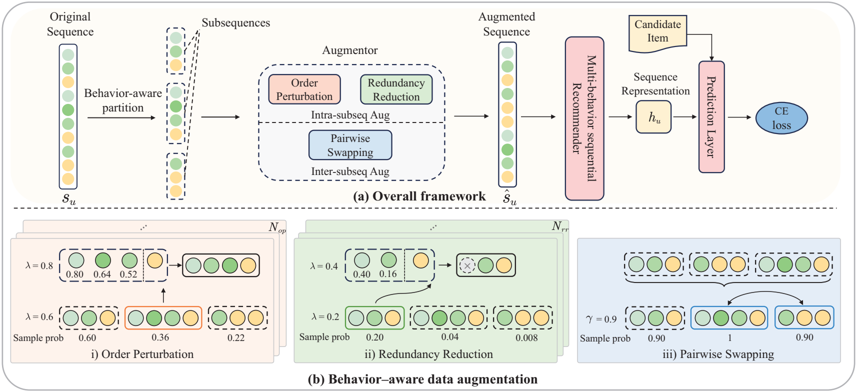

Figure: The overall architecture of the proposed MBASR model.

Behavior-aware Partition

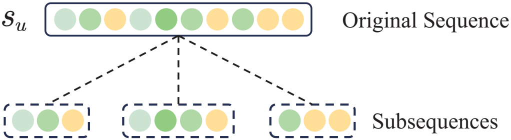

Figure: The illustration of Behavior-aware Partition. The yellow circles symbolize purchase and the green circles denote click.

- Users usually click and compare multiple related items before making a purchase. Therefore, each purchase action and the click behaviors between two consecutive purchases can be regarded as a user's short-term preference within a specific period.

Position-based Sampling

- Our sampling strategy hinges on the order of item positions, with items closer to the current position assigning a lower probability of being sampled.
- We denote $g(\cdot)$ as the score function, which assigns higher scores to items that hold lower positional significance. At the k-th position, $g(k)$ can be formulated as follows:

$$g(k) = \lambda^k \tag{2}$$

And we can get the sampling probability $f(k)$ by:

$$f(k) = g(k) / \sum_{\ell=0}^{L-1} g(\ell) \tag{3}$$

Behavior-aware Data Augmentation

- We propose three data augmentation methods for MBSR, i.e., subseq **o**rder **p**erturbation (OP), subseq **r**edundancy **r**eduction (RR), and **p**airwise subseq **s**wapping (PS).
- Among them, both subseq order perturbation and subseq redundancy reduction occur within subsequences, summarized as Intra-subseq Augmentation, while pairwise subseq swapping takes place between subsequences, referred to as Inter-subseq Augmentation.

Order Perturbation (OP) (1/3)

- Users often click a series of similar items before making a purchase, these click behaviors do not strictly adhere to a specific order.
- We propose to inject controlled order perturbations into the clicked items within subsequences to introduce diverse sequential patterns.

Order Perturbation (OP) (2/3)

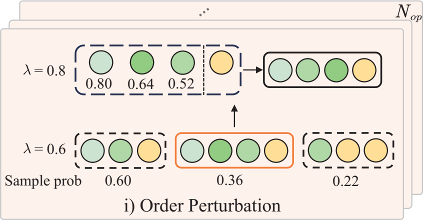

Figure: The illustration of order perturbation.

- Firstly, we calculate the sampling probability f_s for each subsequence in the sequence $\mathcal{S}_u$ based on Equations (2 - 3).
- Then, we sample a certain subsequence $\underline{s}^{op}$ from $\mathcal{S}_u$ using the computed probabilities f_s. To keep the operation simple, within $\underline{s}^{op}$, we again employ Equations (2 - 3) to determine the sampling probability f_v for each clicked item.

Order Perturbation (OP) (3/3)

- Finally, an order perturbation operation is performed based on f_v. The new sequence can be formulated as:

$$\mathcal{S}_u^{OP} = OP(\mathcal{S}_u) = \left\{ s^1, s^2, \cdots, \underline{s}^{op}, \cdots, s^m \right\} \quad (4)$$

$$\underline{s}^{op} = \{\hat{v}_1, \hat{v}_2, \cdots, \hat{v}_c, \cdots v_n\} \quad (5)$$

  where m and n denote the number of subsequences in $\mathcal{S}_u$ and the length of $\underline{s}^{op}$, respectively. Additionally, c denotes the number of clicked items in $\underline{s}^{op}$.
- Moreover, we iterate the above operation $N_{op} = \lfloor \alpha * m \rfloor$ times, i.e., we have collectively perturbed a percentage α of subsequences, where $0 \leq \alpha \leq 1$ and $\lfloor \cdot \rfloor$ is the flooring function. Ultimately, we can obtain the augmented complete sequence $\hat{\mathcal{S}}_u^{OP}$.

Redundancy Reduction (RR) (1/3)

- When users click items in a short period, similarities or redundancies will inevitably occur. We believe that user representations can be enhanced by mitigating redundancy.
- Intentionally reducing redundancy allows the model to capture skip-level sequential patterns.
- By randomly removing some similar or redundant items, we can simulate the randomness and diversity inherent in a user's decision-making process.

Redundancy Reduction (RR) (2/3)

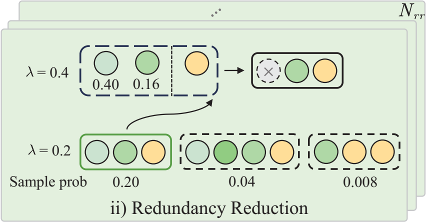

Figure: The illustration of redundancy reduction.

- Firstly, we sample a subsequence $\underline{s}^{RR}$ based on Equations (2 - 3).
- Then, to further mitigate the information loss due to the deletion of items, we again use Equations (2 - 3) to sample an item to be deleted. The new sequence can be formulated as:

$$\mathcal{S}_u^{RR} = RR(\mathcal{S}_u) = \left\{ s^1, s^2, \cdots, \underline{s}^{rr}, \cdots, s^m \right\} \tag{6}$$

Redundancy Reduction (RR) (3/3)

$$\underline{s}^{rr} = \{\hat{v}_1, \hat{v}_2, \cdots, \hat{v}_{c-1}, \cdots, v_{n-1}\} \tag{7}$$

- Similar to the order perturbation operation, we use $N_{rr} = \lfloor \beta * m \rfloor$ to control the number of times the above operation is repeated to obtain the augmented sequence $\hat{\mathcal{S}}_u^{RR}$, where $0 \leq \beta \leq 1$.

Pairwise Swapping (PS) (1/4)

- Based on the assumption that a subsequence characterizes a user's interests during a specific period, we take it as the smallest unit, allowing for a moderate shuffling of the order between subsequences.
- In addition, to achieve a trade-off between the diversity of data samples and model performance, we choose to perform a swap operation on only one pair of subsequences.

Pairwise Swapping (PS) (2/4)

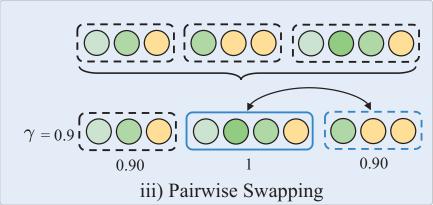

Figure: The illustration of pairwise swapping.

- Firstly, we randomly select a *source* subsequence $\underline{s}^{idx_{source}}$ from sequence $\mathcal{S}_u$ and obtain its index idx_{source}.

Pairwise Swapping (PS) (3/4)

- To minimize noise amplification during the swap operation process, we design a sampling probability $p(\cdot)$ based on the position awareness of the *source* subsequence.
- Subsequences closer to the the *source* subsequence have a higher probability of being sampled as the *destination* subsequence $\underline{s}^{idx_{dest}}$. For the k-th subsequence, its sampling probability, denoted as $p(k)$, is formalized as:

$$p(k) = \frac{x(k)}{\sum_{\ell=0}^{L-1} x(\ell)} \tag{8}$$

$$x(k) = \gamma^{|k - idx_{source}|} \tag{9}$$

Pairwise Swapping (PS) (4/4)

- Here, $0 \leq \gamma \leq 1$ is a parameter that controls the probability distribution. This relative position sampling strategy ensures a higher priority for positions closer to the *source* subsequence when sampling the *destination* subsequence.
- Given the original sequence $\mathcal{S}_u$, once the *source* subsequence and the *destination* subsequence are identified, we enhance $\mathcal{S}_u$ by swapping these two subsequences, resulting in the augmented sequence $\hat{\mathcal{S}}_u^{PS}$.

$$\mathcal{S}_u = \left\{ s^1, \cdots, \underline{s}^{idx_{source}}, \cdots, \underline{s}^{idx_{dest}}, \cdots, s^m \right\} \tag{10}$$

$$\hat{\mathcal{S}}_u^{PS} = PS(\mathcal{S}_u) = \left\{ s^1, \cdots, \underline{s}^{idx_{dest}}, \cdots, \underline{s}^{idx_{source}}, \cdots, s^m \right\} \tag{11}$$

Research Questions

We conduct extensive experiments to answer the following research questions (RQs):

- **RQ1**: How does integrating our MBASR into different mainstream MBSR models perform compared with the original ones and the state-of-the-art data augmentation model?
- **RQ2**: How do the different data augmentation operations proposed in our MBASR enhance the performance of downstream models?
- **RQ3**: How will the data sparsity affect the performance of our MBASR?
- **RQ4**: How is the training efficiency of our MBASR compared with that of the corresponding original model?

Datasets (1/4)

We conduct our experiments on three public recommendation datasets:

- **Tmall[1]:** A public e-commerce dataset released at the IJCAI Competitions 2015, which contains users' shopping logs in six months.
- **UB[2]:** Another e-commerce dataset with multiple behaviors released at the IJCAI Competitions 2016.
- **JD[3]:** A dataset released by the competition of JD in 2019.

[1]https://tianchi.aliyun.com/dataset/dataDetail?dataId = 42
[2]https://tianchi.aliyun.com/dataset/dataDetail?dataId = 649
[3]https://jdata.jd.com/html/detail.html?id=8

Datasets (2/4)

For these datasets, we only keep the clicks and purchases [Zhan et al., 2022], and preprocess them as follows:

- (i) we discard the cold-start items with fewer than 10 records in UB, and 20 in Tmall and JD;
- (ii) we discard the cold-start users with fewer than 10 records in Tmall and 5 in the other two datasets;
- (iii) we order each dataset by the timestamps, and only keep the first (user, item, behavior) triple for repeated ones in a sequence;

Datasets (3/4)

- (iv) following the LOO (leave-one-out)-based validation, in each dataset, we take the last purchase as the test set, the penultimate purchase as the validation set, and the rest as the training set;
- (v) To simulate the sparsity of the data, we take a portion of items at equal intervals in each user's interaction sequence for training. We process Tmall and UB at intervals of 7 and 5, respectively, and JD is not sparsified.

Datasets(4/4)

Table: Statistics of the processed datasets.

Dataset	#Users	#Items	#Clicks	#Purchases	Avg. length
Tmall	17,209	16,177	446,442	223,265	8.89
UB	20,858	30,793	470,731	136,250	5.82
JD	11,367	12,266	131,298	75,774	16.16

Evaluation Metrics

- We adopt two common top-k metrics, i.e., hit ratio (HR) and normalized discounted cumulative gain (NDCG).
- HR@k is defined as the fraction of cases that the ground-truth next item is among the top k items recommended, which emphasizes the accuracy of the model.
- NDCG@k is a position-aware metric, which emphasizes the rank of items, i.e., the top-ranked items are more important.

Baselines (1/3)

- To showcase the effectiveness of our MBASR, we integrate it into a wide range of representative models, including some RNN-based, attention-based and GNN-based models for multi-behavior sequential recommendation.
  - RLBL [Liu et al., 2017].An RNN-based model that combines RNN and LBL (log-bilinear) to capture long-term and short-term preferences.
  - RIB [Zhou et al., 2018]. An RNN-based model that concatenates the item embedding and the behavior embedding as the input of a GRU layer. It then adopts an attention layer to distinguish the effects of different types of behaviors.
  - BINN [Li et al., 2018]. An RNN-based model that designs a contextual long short-term memory (CLSTM) network to integrate users' historical and current preferences.

Baselines (2/3)

- GRUBAR [He et al., 2022]. An extension model of BAR with GRU4Rec [Hidasi and Karatzoglou, 2018] as its backbone model.
- SASBAR [He et al., 2022]. An extension model that utilizes BAR with SASRec [Kang and McAuley, 2018] as the backbone model.
- GPG4HSR [Chen et al., 2022]. A GNN-based model that constructs a global graph to capture the transitions between different types of behaviors, and utilizes a personalized graph to enhance the sequence representation with context information.

Baselines (3/3)

- Moreover, we compare the boosted performance with the very state-of-the-art data augmentation method RSS [Petrov and Macdonald, 2022], as it is a data-oriented augmentation method like our MBASR, which guarantees fairness in the experiments. It leverages a recency-based sampling strategy to generate new training samples such that every item in a sequence can be selected as the target item.

RQ1: Overall Performance Comparison (1/2)

Table: Experimental results of different MBSR models with (w/) or without (w/o) our MBASR on the three datasets. The best results are boldfaced, and "Imprv." indicates the improvement on HR@10.

Datasets	Metric	RLBL		RIB		BINN		GRUBAR		SASBAR		GPG4HSR	
		w/o	w/	w/o	w/	w/o	w/	w/o	w/	w/o	w/	w/o	w/
Tmall	HR@10	0.0238	**0.0300**	0.0341	**0.0349**	0.0289	**0.0299**	0.0322	**0.0359**	0.0493	**0.0539**	0.0494	**0.0531**
	NDCG@10	0.0121	**0.0148**	0.0183	**0.0185**	0.0152	**0.1560**	0.0170	**0.0188**	0.0283	**0.0296**	0.0263	**0.0280**
	Imprv.		**26.05%**		**2.35%**		**3.46%**		**10.31%**		**9.33%**		**7.49%**
UB	HR@10	0.0396	**0.0424**	0.0310	**0.0315**	0.0339	**0.0372**	0.0345	**0.0382**	0.0420	**0.0494**	0.0487	**0.0503**
	NDCG@10	0.0177	**0.0192**	0.0157	**0.0158**	0.0178	**0.0196**	0.0183	**0.0201**	0.0220	**0.0261**	0.0257	**0.0277**
	Imprv.		**7.07%**		**1.61%**		**9.73%**		**9.69%**		**17.62%**		**3.29%**
JD	HR@10	0.1387	**0.1975**	0.3196	**0.3215**	0.3174	**0.3279**	0.3048	**0.3135**	0.3085	**0.3198**	0.3235	**0.3292**
	NDCG@10	0.0621	**0.0939**	0.1797	**0.1797**	0.1825	**0.1866**	0.1743	**0.1787**	0.1652	**0.1718**	0.1789	**0.1871**
	Imprv.		**42.39%**		**0.59%**		**3.31%**		**2.85%**		**3.66%**		**1.76%**

RQ1: Overall Performance Comparison (2/2)

Table: Performance comparison of the data augmentation model RSS and our MBASR. Our MBASR is equipped with pairwise swapping as the data augmentation method. The best results are marked in bold, and the second-best results are underlined.

Model	Method	Tmall		UB		JD	
		HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10
SASRec	-	0.0473	0.0265	0.0431	0.0224	0.3118	0.1674
	RSS	0.0431	0.0233	**0.0468**	0.0244	0.3209	0.1684
	MBASR	**0.0517**	**0.0278**	0.0460	**0.0246**	**0.3365**	**0.1782**
SASBAR	-	0.0493	0.0283	0.0420	0.0220	0.3085	0.1652
	RSS	0.0500	0.0281	0.0408	0.0220	0.3105	0.1613
	MBASR	**0.0539**	**0.0296**	**0.0494**	**0.0261**	**0.3217**	**0.1732**

RQ2: Effect of Augmentation Operators

To comprehensively validate the contributions of different data augmentation operations, we illustrate the performance of models when combining two data augmentation operations, namely, OP and RR.

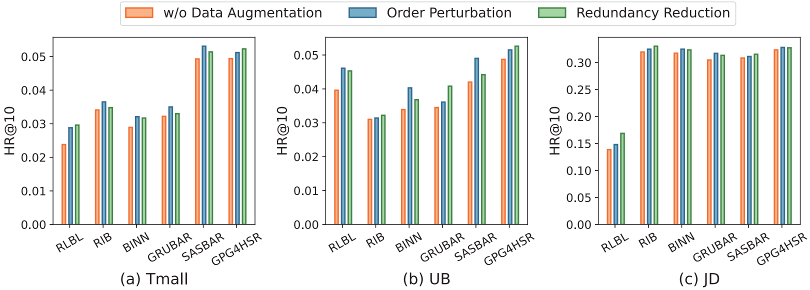

Figure: Experimental results of different MBSR models with our proposed data augmentation operations or without (w/o) any data augmentation on the three datasets.

RQ3: Effect of Data Sparsity

To investigate the impact of data sparsity on our MBASR, we intentionally inject sparsity into UB by extracting items from each user sequence at varying intervals, specifically 1, 3, and 5.

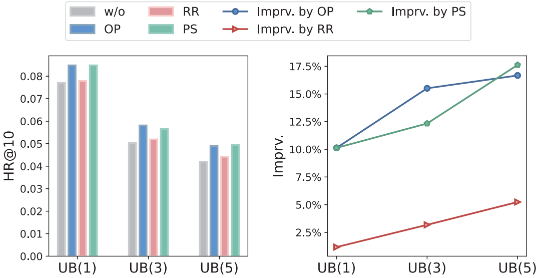

Figure: Impact of data sparsity on our MBASR.

RQ4: Training Efficiency

To visualize the model performance, we present the performance curves of our MBASR with SASBAR as the backbone model and SASBAR over 100 epochs.

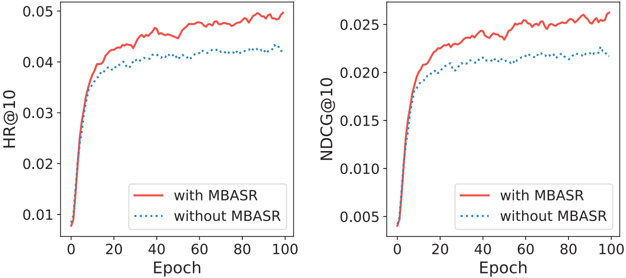

Figure: Training curve on UB.

Conclusions

- We study the problem of data sparsity in multi-behavior sequential recommendation.
- We propose a generic data-oriented framework called MBASR that contains three behavior-aware data augmentation methods (i.e., order perturbation, redundancy reduction, and pairwise swapping).
- We introduce a position-based sampling strategy, which can reduce the level of perturbations and enable a trade-off between enriching the samples and preserving the original information.
- We analyze the performance improvement brought by each augmentation method in different downstream tasks.
- Extensive experiments show that our MBASR can be seamlessly integrated into various multi-behavior sequential recommendation models to improve their performance significantly.

Thank you!

- We thank Miss Shu Chen for her helpful discussions, and the support of Guangdong Basic and Applied Basic Research Foundation (Grant No. 2024A1515010122) and National Natural Science Foundation of China (Nos. 62272315 and 62172283).

Chen, W., He, M., Ni, Y., Pan, W., Chen, L., and Ming, Z. (2022).
Global and personalized graphs for heterogeneous sequential recommendation by learning behavior transitions and user intentions.
In *Proceedings of the 16th ACM Conference on Recommender Systems*, RecSys '22, pages 268–277.

He, M., Pan, W., and Ming, Z. (2022).
BAR: Behavior-aware recommendation for sequential heterogeneous one-class collaborative filtering.
Information Sciences, 608(C):881–899.

Hidasi, B. and Karatzoglou, A. (2018).
Recurrent neural networks with top-k gains for session-based recommendations.
In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, pages 843–852.

Jiang, J., Luo, Y., Kim, J. B., Zhang, K., and Kim, S. (2021).
Sequential recommendation with bidirectional chronological augmentation of transformer.
arXiv preprint arXiv:2112.06460.

Kang, W.-C. and McAuley, J. (2018).
Self-attentive sequential recommendation.
In *Proceedings of the 18th IEEE International Conference on Data Mining*, ICDM '18, pages 197–206.

Li, Z., Zhao, H., Liu, Q., Huang, Z., Mei, T., and Chen, E. (2018).
Learning from history and present: Next-item recommendation via discriminatively exploiting user behaviors.
In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '18, pages 1734–1743.

Liu, Q., Wu, S., and Wang, L. (2017).
Multi-behavioral sequential prediction with recurrent log-bilinear model.
IEEE Transactions on Knowledge and Data Engineering, 29(6):1254–1267.

Liu, Z., Fan, Z., Wang, Y., and Yu, P. S. (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*, SIGIR '21, pages 1608–1612.

Petrov, A. and Macdonald, C. (2022).
Effective and efficient training for sequential recommendation using recency sampling.
In *Proceedings of the 16th ACM Conference on Recommender Systems*, RecSys '22, pages 81–91.

Qiu, R., Huang, Z., Yin, H., and Wang, Z. (2022).
Contrastive learning for representation degeneration problem in sequential recommendation.
In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining*, WSDM '22, pages 813–823.

Wang, Z., Zhang, J., Xu, H., Chen, X., Zhang, Y., Zhao, W. X., and Wen, J.-R. (2021).
Counterfactual data-augmented sequential recommendation.
In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, pages 347–356.

Xie, X., Sun, F., Liu, Z., Wu, S., Gao, J., Zhang, J., Ding, B., and Cui, B. (2022).
Contrastive learning for sequential recommendation.
In *2022 IEEE 38th International Conference on Data Engineering*, ICDE '22, pages 1259–1273.

Zhan, Z., He, M., Pan, W., and Ming, Z. (2022).
Transrec++: Translation-based sequential recommendation with heterogeneous feedback.
Frontiers Comput. Sci., 16(2):162615.

Zhou, M., Ding, Z., Tang, J., and Yin, D. (2018).
Micro behaviors: A new perspective in e-commerce recommender systems.
In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*, WSDM '18, pages 727–735.