Barlow Twins for Sequential Recommendation

Ivan Razvorotnev
ivan.razvorotnev@skoltech.ru
Skoltech, Higher School of Economics
Moscow, Russia

Marina Munkhoeva
AIRI
Moscow, Russia

Evgeny Frolov
AIRI, HSE University
Moscow, Russia

Abstract

Sequential recommendation models must navigate sparse interaction data, popularity bias, and conflicting objectives like accuracy versus diversity. While recent contrastive self-supervised learning (SSL) methods offer improved accuracy, they come with trade-offs: large batch requirements, reliance on handcrafted augmentations, and negative sampling that can reinforce popularity bias. In this paper, we introduce BT-SR, a novel non-contrastive SSL framework that integrates the Barlow Twins redundancy-reduction principle into a Transformer-based next-item recommender. BT-SR learns embeddings that align users with similar short-term behaviors while preserving long-term distinctions—without requiring negative sampling or artificial perturbations. This structure-sensitive alignment allows BT-SR to more effectively recognize emerging user intent and mitigate the influence of noisy historical context. Our experiments on five public benchmarks demonstrate that BT-SR consistently improves next-item prediction accuracy and significantly enhances long-tail item coverage and recommendation calibration. Crucially, we show that a single hyperparameter can control the accuracy-diversity trade-off, enabling practitioners to adapt recommendations to specific application needs.

CCS Concepts

• **Information systems → Recommender systems**.

Keywords

Sequential Recommendations, Non-Contrastive Learning, Self-Supervised Learning, Recommendation Fairness

1 Introduction

Sequential recommendation models have demonstrated impressive performance by capturing the rich temporal dynamics inherent in user–item interactions across many recommender system domains and applications. Transformer-based [33] architectures remain the state of the art [24, 26, 17] thanks to their ability to model long-range dependencies and complex sequential patterns in users' consumption histories. However, the extreme sparsity typical for such behavioral data often hampers these models' capacity to learn robust user representations from limited interaction information.

Real-world sequential recommender systems must contend with several intertwined problems. First, *popularity bias* skews recommendations toward a small subset of "short-head" items. It marginalizes long-tail content and reduces overall catalog coverage. Second, *metric trade-offs* force practitioners to balance conflicting objectives such as precision versus recall, short-list accuracy versus long-list engagement, or accuracy versus novelty, therefore making one-size-fits-all solutions impractical. Third, achieving true *personalization*

requires reconciling short-term session dynamics with stable long-term user preferences, a tension that often leads to either myopic or overly generic suggestions. Finally, *diversification* is essential to avoid repetitive item sequences and to expose users to a broader range of content, yet naive diversification can compromise relevance.

To address the sparsity issue, recent work has integrated contrastive learning (CL), a class of self-supervised learning (SSL) methods, into sequential recommenders [35, 27, 11, 23]. By pulling together augmented views of the same user sequence and pushing apart different sequences, these methods can substantially boost recommendation accuracy. Yet contrastive approaches introduce new challenges: (1) *popularity-based negative sampling* can exacerbate popularity bias [3], reducing novelty and catalog coverage and reinforcing a feedback loop that disproportionately favors already-popular items; (2) acquiring high-quality negatives demands *large batch sizes*, inflating memory and computational overhead [28, 20]; and (3) *representation collapse* [14], where embeddings become overly similar, diminishing the model's expressivity.

Conversely, *non-contrastive self-supervised methods* have recently achieved remarkable results in computer vision [5] and natural language processing [31] without relying on negative samples [40]. Among these, the Barlow Twins framework [38] stands out: it jointly enforces invariance to perturbations and reduces redundancy across embedding dimensions.

Applied to sequential recommendation, Barlow Twins can help learn diverse, informative user representations with far lower computational cost and without sampling negatives. By explicitly minimizing redundancy across embedding dimensions, the learned representations can better capture *varied aspects* of user behavior, which is critical for increasing recommendation diversity and long-tail coverage. However, the potential of this approach has not been fully studied in prior recommender system literature, which sets the basis of the current work.

This research investigates the application of non-contrastive learning principles to enhance sequential recommender systems. Specifically, we address the following research questions: (i) can non-contrastive learning produce higher-quality user embedding spaces than contrastive learning within the recommendation domain? (ii) does non-contrastive learning effectively mitigate popularity bias in recommendations? (iii) how sensitive are the resulting recommendations to the hyperparameters of the non-contrastive loss function?

This paper presents BT-SR (Barlow Twins for Sequential Recommendation), the first framework to systematically adapt the Barlow Twins redundancy-reduction objective to next-item prediction. Our approach augments a standard Transformer-based recommender with an auxiliary Barlow Twins loss that (i) enforces invariance

under sequence augmentations and (ii) discourages redundant features in the embedding space. By tuning the relative weight of this loss, BT-SR enables practitioners to shift recommendation behavior—balancing precision on head items against long-tail coverage, and short-list accuracy against long-range engagement—without sacrificing overall accuracy.

Our main contributions are:

- **A multi-task Transformer architecture:** We seamlessly integrate the Barlow Twins redundancy-reduction objective with next-item prediction in a unified training loop.
- **Controllable recommendation behavior:** We show how a single hyperparameter in BT-SR steers the model along the spectrum of popularity bias—trading off precision on head items versus enhanced coverage of long-tail items, and balancing short-list versus long-list metrics. This controllability enables system designers to align recommendation behavior with application-specific goals.
- **Reduced popularity bias and improved calibration:** Through extensive experiments, we demonstrate that BT-SR not only mitigates over-recommendation of popular items but also produces more confident and better-calibrated predictions.

Extensive empirical studies on several benchmark datasets confirm that BT-SR consistently outperforms state-of-the-art sequential recommenders and contrastive-learning alternatives.

2 Related Works

2.1 Transformers and Self-Supervised Learning in Sequential Recommendation

SASRec [17] and BERT4Rec [32] pioneered the application of Transformer-based architectures to sequential recommendation. These models have since become the de facto standard due to their capacity to capture long-range dependencies and encode rich user intent representations. Subsequent research has focused on refining their architectural and training components, including attention mechanisms, positional encodings, and optimization objectives [26].

To mitigate data sparsity and enhance representation learning, recent advances have integrated self-supervised learning (SSL) into Transformer-based recommenders. A predominant line of work follows the contrastive learning (CL) paradigm, wherein positive views are generated via data augmentations and pulled closer in the embedding space, while negatives are repelled. CL4Rec [35] introduced four canonical augmentation strategies—cropping, reordering, masking, and substitution—whereas DuoRec [27] proposed target-aware augmentations to preserve user intent and sequence semantics. Further refinements, such as CBiT [11] and SCL [30], have incorporated bidirectional attention mechanisms and embedding uniformity regularization to stabilize optimization and improve generalization.

Despite their empirical success, CL-based frameworks suffer from several intrinsic limitations. First, their reliance on negative sampling incurs additional computational cost and often introduces popularity bias [3]. Second, their performance exhibits high sensitivity to batch size and temperature hyperparameters [28], making them less robust in large-scale or imbalanced recommendation

settings. These challenges have motivated growing interest in non-contrastive self-supervised paradigms that eliminate the dependency on explicit negatives.

2.2 Next-item Prediction Objectives

To establish a principled baseline for our next-item recommendation model, we build upon the SASRec architecture and systematically compare three loss formulations—binary cross-entropy (BCE), full softmax cross-entropy (CE), and scalable cross-entropy (SCE). Each objective exhibits distinct trade-offs between predictive accuracy, optimization stability, and computational scalability. Below, we formalize these objectives and discuss their respective characteristics.

We first formulate next-item recommendation as a binary classification problem and optimize the binary cross-entropy loss:

$$\mathcal{L}_{\text{BCE}}(\theta) = -\sum_{u \in \mathcal{U}} \left[\sum_{t=2}^{n} \log p_\theta(i_t^u, t) + \sum_{j \in S_{u,t}} \log\big(1 - p_\theta(j, t)\big) \right], \quad (1)$$

where

$$p_\theta(i, t) = \sigma\big(r_\theta(i, t)\big), \quad \sigma(x) = \frac{1}{1 + e^{-x}},$$

and $S_{u,t} \subset \mathcal{I}$ denotes a set of negative samples (items not yet interacted with by u before step t). In practice, one or a few negatives are drawn per step to approximate the full objective.

Recent work has demonstrated that replacing SASRec's binary cross-entropy with the full softmax-based cross-entropy substantially improves next-item prediction accuracy by framing the task as a multinomial logistic regression problem. Given the scoring function $r_\theta(i, t)$, the probability of recommending item i at step t is expressed as

$$p_\theta^c(i, t) = \text{softmax}\big(r_\theta(i, t)\big) \quad (\text{over all } i \in \mathcal{I}), \quad (2)$$

yielding the cross-entropy objective

$$\mathcal{L}_{\text{CE}}(\theta) = -\sum_{u \in \mathcal{U}} \sum_{t=2}^{n} \log p_\theta^c(i_t^u, t). \quad (3)$$

While this formulation achieves superior predictive performance, computing the full softmax over the entire item catalog incurs $O(|\mathcal{U}| \cdot N)$ time and space complexity, which becomes intractable for large-scale recommendation systems. Sampling-based approximations or partial normalization strategies thus remain essential for scalability.

To address this computational bottleneck, [24] proposed the Scalable Cross-Entropy (SCE)—a state-of-the-art approximation that replaces the full normalization term with a dynamically sampled candidate set. Let $C_{u,t} \subset \mathcal{I}$ denote a subset containing the true next item i_t^u and K negative samples drawn uniformly or from a learned proposal distribution. Then,

$$p_\theta^s(i, t) = \frac{\exp\big(r_\theta(i, t)\big)}{\sum_{j \in C_{u,t}} \exp\big(r_\theta(j, t)\big)} \quad (i \in C_{u,t}),$$

and the corresponding sampled cross-entropy loss is given by

$$\mathcal{L}_{\text{SCE}}(\theta) = -\sum_{u \in \mathcal{U}} \sum_{t=2}^{n} \log p_\theta^s(i_t^u, t). \quad (4)$$

By constraining the normalization to the sampled candidate set $C_{u,t}$, SCE reduces both time and memory complexity to $O(|\mathcal{U}| \cdot K)$ per update, thereby enabling efficient training on catalogs with millions

of items. In this study, we combine the Barlow Twins regularization term with each of the BCE, CE, and SCE objectives to assess its influence across different backbone–loss configurations.

2.3 Non-Contrastive Learning for Representation Learning

Non-contrastive self-supervised learning (NCL) aims to learn invariant and discriminative representations without relying on explicit negative samples. Representative NCL frameworks include BYOL [13], SimSiam [6], VICReg [2], and Barlow Twins [37]. These methods employ architectural asymmetries—such as stop-gradient operations, momentum encoders, or prediction heads—and objective-specific regularizations, including redundancy reduction and variance normalization, to avert representational collapse.

Although these paradigms have achieved remarkable success in computer vision and have recently been extended to NLP [31], their application to recommender systems, and particularly to sequential models, remains relatively underexplored. Existing efforts primarily adopt an offline pretraining–fine-tuning pipeline: for instance, CLUE [7] adapts BYOL for collaborative filtering through offline self-supervised pretraining, and SelfCF [23] integrates self-supervision into non-sequential matrix factorization models.

The most relevant prior work, NCL-SR [39], extends non-contrastive learning to sequential recommendation; however, it relies on external side information to construct augmented views and lacks a systematic analysis of model controllability and representation behavior. In contrast, our BT-SR framework introduces a unified multi-task training scheme that integrates the Barlow Twins objective directly with the next-item prediction loss, enabling controllable redundancy reduction during training.

Furthermore, recent work [22] employed the Barlow Twins objective for SSL-based pretraining followed by fine-tuning on downstream recommendation tasks, including next-item prediction. We argue that such a two-stage pipeline is suboptimal for the recommendation domain: (i) SSL pretraining is substantially more time-consuming than joint multi-task optimization, and (ii) the multitask setup, wherein the Barlow Twins loss functions as a regularizer, allows for explicit control over recommendation diversity and behavioral bias through a single hyperparameter. This property is central to our contribution and will be further discussed in subsequent sections.

3 Proposed Approach

The challenges outlined in Section 1—such as popularity bias, personalization trade-offs, and embedding sparsity—are rooted in how user interaction sequences are represented. Transformer-based sequential recommenders rely on fixed-length embeddings to summarize user histories, but these embeddings often struggle to balance two critical goals: capturing shared short-term intent and preserving user-specific long-term preferences. This imbalance leads to brittle recommendations, especially in sparse or skewed data settings.

In this section, we introduce **BT-SR**, a non-contrastive learning framework that enhances sequence embeddings by reducing feature redundancy and promoting invariance under meaningful augmentations. We first formalize the problem setting and describe

the next-item prediction objective used as a base task (Section 3.1). Next, we introduce *our supervised augmentation strategy, which defines semantically meaningful views of user sequences* (Section 3.2). Finally, we present the Barlow Twins loss as an auxiliary objective that encourages redundancy reduction and representation alignment across augmented views (Section 3.3).

3.1 Problem Setup and Preliminaries

We consider a standard sequential recommendation setting where the goal is to predict the next item a user will interact with, given their past behavior. Let $\mathcal{U}$ denote the set of users and $\mathcal{I} = \{i_1, \ldots, i_N\}$ the catalog of items. Each user $u \in \mathcal{U}$ has a chronological interaction sequence:

$$S_u = \left(i^u_{\pi^u_1}, i^u_{\pi^u_2}, \ldots, i^u_{\pi^u_{|S_u|}}\right),$$

where π^u_t indexes interactions by timestamp. The goal of sequential recommendation is to learn a function that, given the n most recent items from S_u, accurately predicts the next item. Formally, the model is trained to maximize the likelihood

$$P_\theta\left(i^u_t \mid i^u_{t-n}, \ldots, i^u_{t-1}\right), \tag{5}$$

where θ denotes model parameters.

In practice, this is implemented by encoding the user history at moment t in the form of the item sequence $s_u(t) = [i^u_{t-n}, \ldots, i^u_{t-1}]$ into a representation $z_u(t) = f_\theta(s_u(t))$ using a Transformer encoder. The next-item prediction task is then formulated as a classification problem based on the relevance scores over the entire item catalog. The scoring function is typically defined in a matrix-factorization style as a scalar product between an item embedding e_i from the catalog and the current sequence state: $r_\theta(i, t) = e_i^\top z_u(t)$. We denote the classification loss as $\mathcal{L}_{\text{pred}}$.

As baselines, we use SASRec variants trained with binary cross-entropy (BCE), full softmax cross-entropy (CE), and scalable sampled softmax (SCE), described in detail in Section 4.1.

While effective for optimizing next-item accuracy, these objectives do not explicitly encourage structural alignment across semantically similar user sequences. In what follows, we introduce an augmentation scheme designed to reveal such alignment, followed by an auxiliary redundancy-reduction objective that strengthens the representational structure of the embedding space.

3.2 Designing Supervised Augmentations

To induce semantically meaningful alignment between users, we design a supervised augmentation scheme guided by the next-item label. Inspired by prior work on label-guided contrastive learning [27], we construct augmentations based on shared recent behavior. Specifically, given an anchor sequence $S_u = [i^u_1, \ldots, i^u_t]$ ending with target item i^u_t, we uniformly sample another sequence $S_{u'}$ from the training set whose final item is also i^u_t. These two sequences, though originating from different users, reflect convergent behavioral patterns that are likely to lead to the same recommendation target.

We treat such sequence pairs $(S_u, S_{u'})$ as positive examples for the Barlow Twins loss. This encourages the model to produce similar embeddings for distinct consumption paths that converge to the same intent, while still allowing diversity across user histories.

Crucially, we avoid applying random augmentations (e.g., masking, cropping, dropout-based perturbations), as we find that they introduce noise and diminish performance in our multi-task, non-contrastive setup.

This augmentation strategy is natural for the next-item prediction task and emphasizes behavioral convergence as a signal for alignment, allowing the model to generalize across different interaction paths while preserving user-specific information.

We avoid synthetic augmentations (masking, cropping, dropout) proposed in [36], as they introduce stochastic noise, harm stability in our non-contrastive multi-task setup, and do not provide controllability in recommendation behavior.

3.3 Redundancy-Reduction-Based Regularization

Building on the behavioral alignment provided by our augmentation scheme, we now define a redundancy-reduction objective that enhances the quality of sequence embeddings by promoting feature diversity and invariance. This objective, based on the Barlow Twins (BT) framework, is integrated as an auxiliary loss in our multi-task training setup.

Let Z^A and Z^B denote the original and augmented batches of sequence embeddings produced by the backbone network. We assume both views are ℓ_2-normalized and mean-centered across the batch. We apply ℓ_2 normalization instead of the learned projection head originally used in the BT framework, as it is parameter-free and empirically more stable. The cross-correlation matrix $C \in \mathbb{R}^{D \times D}$ is computed as

$$C_{ij} = \frac{1}{B} \sum_{b=1}^{B} \frac{Z_{b,i}^A Z_{b,j}^B}{\sqrt{\sum_{b'=1}^{B}(Z_{b',i}^A)^2}\sqrt{\sum_{b'=1}^{B}(Z_{b',j}^B)^2}}, \qquad (6)$$

where b indexes samples in a batch of size B, and i, j index embedding dimensions. Each $C_{ij} \in [-1, 1]$, with 1 indicating perfect correlation and -1 indicating perfect anti-correlation.

The corresponding BT loss is composed of two terms:

$$\mathcal{L}_{BT} = \sum_{i=1}^{D}(1 - C_{ii})^2 + \lambda \sum_{i=1}^{D}\sum_{\substack{j=1\\j \neq i}}^{D} C_{ij}^2, \qquad (7)$$

where λ trades off invariance against decorrelation. Driving the diagonal elements of C toward 1 enforces perturbation invariance, while pushing the off-diagonal elements toward 0 reduces redundancy.

We now define the complete loss function used to train BT-SR, combining next-item prediction with redundancy reduction in a single multi-task objective:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{pred}} + \alpha \mathcal{L}_{\text{BT}}, \qquad (8)$$

where $\mathcal{L}_{\text{pred}}$ is the standard next-item prediction loss and $\mathcal{L}_{\text{BT}}$ is the Barlow Twins redundancy-reduction loss. The hyperparameter α modulates the influence of self-supervised regularization during training.

This formulation enables the model to learn embeddings that are both robust to behavioral variation and structurally expressive. As shown in Section 5, tuning α allows practitioners to steer recommendation behavior in a controllable and interpretable way.

Table 1: NDCG@10 performance comparison of three SAS-Rec variants (BCE loss, CE loss, Scalable CE loss) with and without the Barlow Twins regularization term. Results highlight consistent improvements from Barlow Twins across all scenarios.

	BCE	FCE	SCE	BCE + BT	FCE + BT	SCE + BT
ML1M	0.0443	0.0494	0.0503	0.0490	0.0583	0.0562
YELP	0.0146	0.0127	0.0137	0.0149	0.0150	0.0141
Gowalla	0.0419	0.0364	0.0457	0.0434	0.0481	0.0483
Beauty	0.0434	0.0524	0.0543	0.0440	0.0563	0.0549
Kindle Store	0.0705	0.0700	0.0765	0.0733	0.0746	0.0796

4 Experimental Setup

Datasets. We follow the experimental protocol proposed in [24] for benchmarking backbone architectures. Experiments are conducted on five public datasets: Behance [16], Kindle Store [25], Yelp [1], Gowalla [8], and MovieLens-1M [15]. Users and items with fewer than five interactions are filtered out. To prevent temporal leakage, we adopt a timestamp-based split strategy [12]. A global timestamp at the 0.95 quantile of all interactions defines the boundary between training and test data: interactions before this point form the training set, and users with subsequent interactions (excluded from training) constitute the test pool. For each test user, we apply a standard leave-one-out protocol, using the latest interaction for testing and the second-to-last interaction for validation. All experiments are conducted on a single NVIDIA A100 GPU. The code and preprocessing scripts are publicly available[1].

Metrics. Following best practices [4, 9, 19], we evaluate models using unsampled top-K ranking metrics computed over the full item catalog. We report Normalized Discounted Cumulative Gain (ndcg@K) and Hit Rate (hr@K) for $K = 1, 5, 10, 50$, averaged across all test users. To assess recommendation diversity, we report item coverage (cov@K), defined as the fraction of unique items appearing in top-K recommendations across all users. In addition to ranking and coverage metrics, we analyze the structure of learned user embeddings using the *effective rank* [29]. Given the singular values $\sigma_1, \sigma_2, \ldots, \sigma_D$ of the user-embedding matrix (obtained via singular value decomposition), we normalize them as $p_i = \sigma_i / \sum_{j=1}^{D} \sigma_j$ and define the effective rank as the exponential of the Shannon entropy:

$$r_{\text{eff}} = \exp\left(-\sum_{i=1}^{D} p_i \log p_i\right). \qquad (9)$$

A higher r_{eff} indicates a flatter singular-value spectrum and therefore a more diverse, less redundant embedding space.

Hyperparameters. All models are implemented in PyTorch and optimized with Adam using a learning rate of 0.001. The maximum sequence length is fixed to 50, truncating longer histories to the most recent interactions. L2 weight decay is applied for regularization, with the coefficient tuned on the validation set (typically 10^{-4}–10^{-5}). We perform a grid search over the Barlow Twins hyperparameters α and λ, selecting values from $\{0.05, 0.10, \ldots, 0.50\}$ based on validation performance. The auxiliary loss $\mathcal{L}_{BT}$ is applied to different SASRec variants trained with $\mathcal{L}_{CE}$, $\mathcal{L}_{\text{Scalable CE}}$ [24], and $\mathcal{L}_{BCE}$ [18]. We report the best-performing model for each dataset.

---

[1]https://github.com/RAZVOR/barlow_twins_sasrec

Table 2: Performance comparison across all datasets. Bold scores are the best on the dataset for the given metric, underlined scores are the second best.

Dataset	Metric	SasRec(BCE)	SasRec(CE)	SasRec(SCE)	CL4SRec	DuoRec	EC4SRec	BT-SR(Ours)	Improve
ML 1M	hr@1	0.0160	0.0213	0.0200	0.0182	0.0200	0.0213	**0.0240*** ± 0.0048	+12.7%
	hr@5	0.0493	0.0600	0.0587	0.0476	0.0480	0.0534	**0.0639*** ± 0.0024	+6.5%
	hr@10	0.0893	0.0847	0.0933	0.0834	0.0800	0.0880	**0.0945*** ± 0.0080	+1.3%
	ndcg@5	0.0319	0.0404	0.0393	0.0342	0.0353	0.0364	**0.0436*** ± 0.0020	+7.9%
	ndcg@10	0.0443	0.0494	0.0503	0.0464	0.0456	0.0487	**0.0534*** ± 0.0037	+6.2%
	ndcg@50	0.0799	0.0818	0.0800	0.0712	0.0723	0.0748	**0.0833*** ± 0.0033	+1.8%
	cov@1	0.0320	0.0887	**0.0961**	0.0335	0.0361	0.0387	0.0816 ± 0.0035	-15.1%
	cov@5	0.1473	0.2265	**0.2568**	0.0969	0.0936	0.1002	0.2223 ± 0.0083	-13.4%
	cov@10	0.2434	0.3185	**0.3645**	0.1329	0.1459	0.1591	0.3189 ± 0.0123	-12.5%
YELP	hr@1	0.0045	0.0043	0.0040	0.0028	0.0027	0.0028	**0.0049*** ± 0.0005	+8.9%
	hr@5	0.0162	0.0137	0.0152	0.0087	0.0094	0.0090	**0.0183*** ± 0.0008	+13.0%
	hr@10	**0.0292**	0.0257	0.0277	0.0192	0.0188	0.0199	0.0288 ± 0.0010	−1.4%
	ndcg@5	0.0104	0.0088	0.0097	0.0060	0.0061	0.0062	**0.0117*** ± 0.0006	+12.5%
	ndcg@10	0.0146	0.0127	0.0137	0.0085	0.0091	0.0092	**0.0150*** ± 0.0006	+2.7%
	ndcg@50	**0.0280**	0.0263	0.0258	0.0178	0.0172	0.0184	0.0267 ± 0.0007	−4.6%
	cov@1	0.0103	**0.0233**	0.0135	0.0057	0.0059	0.0059	0.0218 ± 0.0034	−6.4%
	cov@5	0.0333	0.0621	0.0413	0.0170	0.0161	0.0171	**0.0636*** ± 0.0089	+2.4%
	cov@10	0.0541	0.0911	0.0661	0.0276	0.0255	0.0275	**0.0998*** ± 0.0128	+9.5%
Gowalla	hr@1	0.0173	0.0143	0.0196	0.0157	0.0144	0.0142	**0.0205*** ± 0.0010	+4.6%
	hr@5	0.0506	0.0431	0.0548	0.0433	0.0473	0.0440	**0.0592*** ± 0.0010	+8.0%
	hr@10	0.0756	0.0660	0.0811	0.0694	0.0712	0.0757	**0.0851*** ± 0.0020	+4.9%
	ndcg@5	0.0339	0.0289	0.0372	0.0318	0.0310	0.0331	**0.0400*** ± 0.0009	+7.5%
	ndcg@10	0.0419	0.0364	0.0457	0.0411	0.0387	0.0397	**0.0483*** ± 0.0013	+5.7%
	ndcg@50	0.0613	0.0528	0.0640	0.0535	0.0587	0.0549	**0.0667*** ± 0.0015	+4.2%
	cov@1	0.0230	0.0228	0.0214	0.0252	0.0255	0.0270	**0.0321*** ± 0.0039	+25.9%
	cov@5	0.0892	0.0725	0.0921	0.0950	0.0939	0.0998	**0.1280*** ± 0.0140	+36.3%
	cov@10	0.1557	0.1153	0.1625	0.1538	0.1595	0.1650	**0.2187*** ± 0.0237	+34.6%
Beauty	hr@1	0.0179	0.0269	0.0305	0.0292	0.0305	0.0303	**0.0326*** ± 0.0013	+6.9%
	hr@5	0.0538	0.0591	0.0582	0.0555	0.0573	0.0574	**0.0606*** ± 0.0025	+2.5%
	hr@10	0.0789	0.0896	0.0905	0.0830	0.0860	0.0815	**0.0937*** ± 0.0019	+3.5%
	ndcg@5	0.0355	0.0425	0.0442	0.0412	0.0436	0.0455	**0.0463*** ± 0.0002	+1.8%
	ndcg@10	0.0434	0.0524	0.0543	0.0530	0.0527	0.0537	**0.0569*** ± 0.0001	+4.8%
	ndcg@50	0.0659	0.0777	0.0794	**0.0833**	0.0789	0.0794	0.0813 ± 0.0035	-2.4%
	cov@1	0.0620	0.0546	0.0499	0.0603	0.0607	**0.0679**	0.0491 ± 0.0078	-20.7%
	cov@5	0.2200	0.1782	0.1760	**0.2212**	0.2100	0.2103	0.1532 ± 0.0357	-29.2%
	cov@10	**0.3448**	0.2752	0.2766	0.3362	0.3234	0.3236	0.3234 ± 0.0612	-30.5%
Kindle Store	hr@1	0.0451	0.0471	0.0533	0.0469	0.0471	0.0505	**0.0564*** ± 0.0007	+5.8%
	hr@5	0.0832	0.0813	0.0899	0.0813	0.0762	0.0785	**0.0903*** ± 0.0018	+0.4%
	hr@10	0.0990	0.0965	0.1022	0.0910	0.0897	0.0955	**0.1058*** ± 0.0012	+3.5%
	ndcg@5	0.0655	0.0652	0.0726	0.0635	0.0625	0.0652	**0.0747*** ± 0.0009	+2.9%
	ndcg@10	0.0705	0.0700	0.0765	0.0662	0.0668	0.0672	**0.0796*** ± 0.0005	+4.1%
	ndcg@50	0.0786	0.0803	0.0852	0.0731	0.0771	0.0736	**0.0883*** ± 0.0007	+3.6%
	cov@1	0.0396	0.0337	0.0409	0.0363	0.0356	0.0369	**0.0440*** ± 0.0005	+7.6%
	cov@5	0.1278	0.1076	0.1492	0.1077	0.1123	0.1082	**0.1665*** ± 0.0019	+11.6%
	cov@10	0.1920	0.1665	0.2346	0.1598	0.1719	0.1703	**0.2673*** ± 0.0021	+13.9%

To ensure statistical significance, we report the mean and standard deviation over five independent runs. Significance is verified using paired t-tests ($p < 0.05$) against the second-best baseline, with statistically significant improvements marked by an asterisk (*). The low observed variance demonstrates the stability of our method.

4.1 Baselines

As baselines, we adopt SASRec variants trained with: (1) binary cross-entropy (BCE) [17], (2) standard cross-entropy (CE), and (3) scalable cross-entropy (SCE) [24]. For contrastive-learning baselines, we include CL4SRec [36], DuoRec [27], and EC4Rec, an explanation-guided contrastive framework that leverages training gradients to identify positive and negative items [34].

While recent LLM- and LoRA-based recommenders perform well in text-rich or cold-start settings, they underperform in ID-based domains and incur substantially higher inference costs. Moreover, prior studies [10, 21] show that large language models often memorize public recommendation datasets, raising concerns of target leakage rather than genuine generalization. For these reasons, we exclude LLM-based baselines from our evaluation.
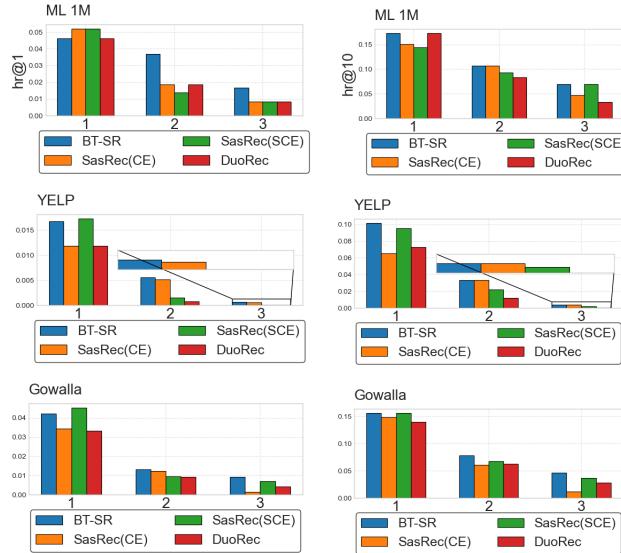
Figure 1: HR@1 (left) and HR@10 (right) metrics for the three item-popularity buckets across three datasets.

Figure 2: (left) Comparison of score density distributions for positive and negative candidate pairs across three datasets. We quantify model's confidence in distinguishing relevant candidates by the histogram overlap factor (*Overlap*)—lower values indicate better separation. (right) Singular value spectra of the sequence embeddings for each dataset, annotated with their computed effective ranks (in legend), illustrate the effective dimensionality of the learned representations.

5 Results

We first evaluate the impact of adding the Barlow Twins term to three SasRec objectives ($\mathcal{L}_{BCE}$, $\mathcal{L}_{CE}$, $\mathcal{L}_{SCE}$). As shown in Table 1, the Barlow Twins term consistently improves performance across all base losses in the multi-task setup, demonstrating the universality of our approach.

Next, we select the best-performing objective for each dataset. $\mathcal{L}_{SCE}$ achieves the strongest results on Gowalla and Kindle Store, while $\mathcal{L}_{CE}$ performs better on the remaining datasets. Using these selected models, we compare against strong baselines on the hold-out test set. Table 2 reports our best variant (BT-SR), which consistently outperforms all baselines in both utility and coverage across most datasets. To further investigate the role of the Barlow Twins loss in representation learning, we provide a detailed analysis in the following section.

5.1 Analysis

5.1.1 What Drives BT-SR's Outperformance?

Although BT-SR achieves only modest coverage gains on Movie-Lens and Amazon Beauty—remaining below the second-best baseline—it delivers substantially higher coverage on the other three datasets. To unpack this behavior, we split each item corpus into three popularity-based buckets (each containing roughly one-third of total interactions), ordered from most to least popular. Figure 1 shows HR@1 and HR@10 for each bucket.

Surprisingly, BT-SR underperforms slightly at HR@1 in the top-popularity bucket, yet it markedly outperforms all baselines on the mid- and low-popularity buckets—evidence of its enhanced personalization. Moreover, BT-SR also leads at HR@10 even for the most popular items, indicating that *it can elevate niche items into top-rank positions without sacrificing performance on blockbusters.*
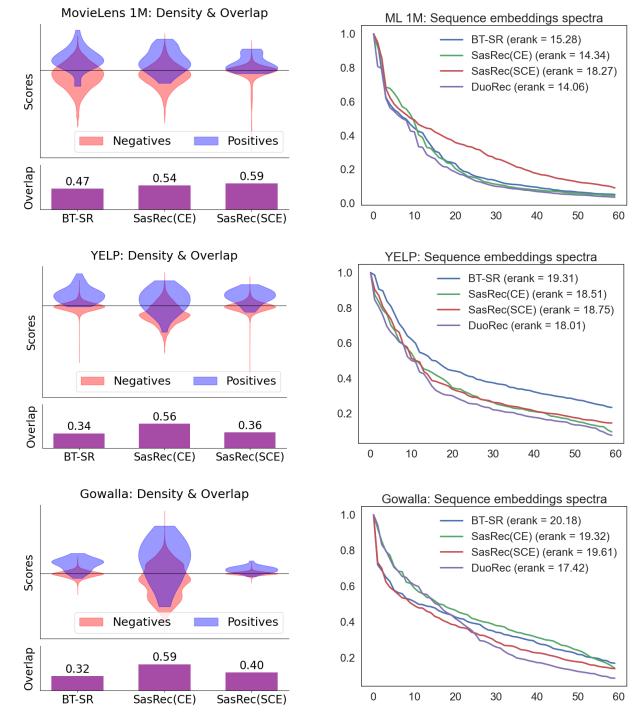
This pattern highlights how the Barlow Twins loss both prevents representation collapse and *mitigates popularity bias.*

Further, in Section 5.4 we demonstrate that BT-SR can be further tuned for industrial applications via an optional hyperparameter that explicitly balances HR@1 against HR@10—allowing practitioners to prioritize the metric that best suits their use case.

5.1.2 Can non-contrastive learning produce higher-quality user embedding spaces than contrastive learning in recommendation?

Our method outperforms contrastive learning (CL) baselines, not only demonstrating greater effectiveness but also suggesting that it learns higher-quality user embeddings.

We compared the score distributions for positive (ground-truth) and negative items. Figure 2 shows that BT-SR consistently assigns higher scores to positive items than competing models across three datasets. This confirms that our approach sharpens user representations and more clearly distinguishes true preferences from noise.

To analyze the embedding geometry, we performed singular value decomposition (SVD) on each set of user embeddings and plotted their singular-value spectra (Figure 2). We then computed
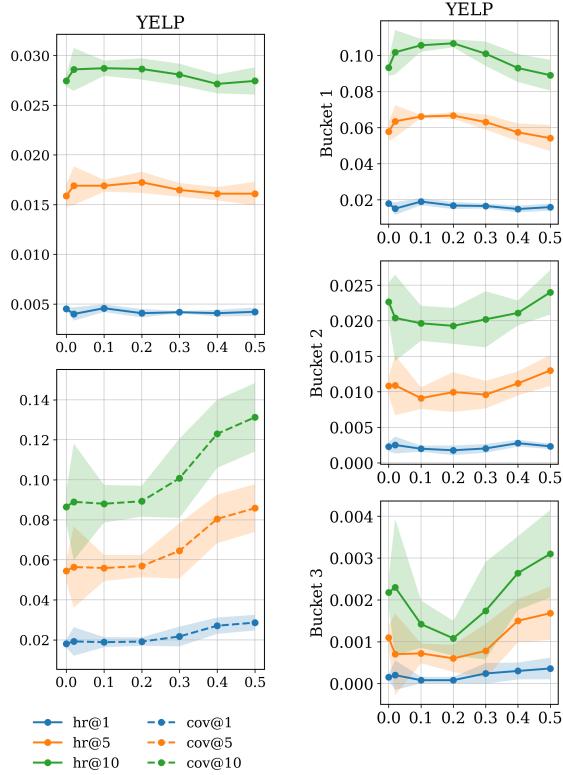
Figure 3: Parameter sensitivity wrt α for YELP dataset. Left: common metrics, Right: item-popularity bucket metrics. Picture demonstrates controllability of recommendations via hyperparameter.
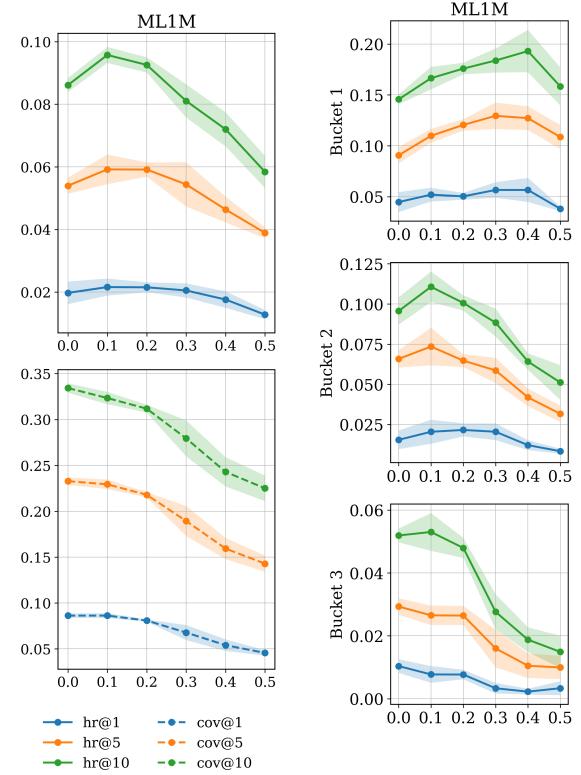

Figure 4: Parameter sensitivity wrt α for MovieLens dataset. Left: common metrics, Right: item-popularity bucket metrics. Picture demonstrates controllability of recommendations via hyperparameter.

the effective rank (Eq. 9) to quantify the flatness of the singular-value spectrum. On Yelp and Gowalla, BT-SR produces a flatter spectrum, whereas on MovieLens the spectrum is steeper.

5.2 Ablation Study

To analyze the balance between the primary recommendation loss and our Barlow–Twins regularizer, we performed a hyperparameter sweep over $\alpha \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5\}$, where $\alpha = 0$ effectively removes the BT term in Equation 8. We also evaluated sensitivity to the decorrelation weight λ in Equation (11). To isolate each effect, we first fixed λ at its optimal value and varied α, and then held α constant while sweeping λ. Figures 3 and 4 plot (i) hit-rate metrics (hr@1, hr@5, hr@10) and (ii) coverage@K as functions of α. Additional ablation studies are available in Appendix A.

When $\alpha \leq 0.1$, the model favors a short, high-precision recommendation list: hr@1 and hr@5 peak, while coverage remains low. As α increases, precision at top-1 drops by only 2–3 pp, but hr@10 improves and coverage@10 nearly doubles. In effect, α acts as a dial between a *short-list/high-precision* mode and a *long-list/high-diversity* mode.

To further dissect this trade-off, we studied metrics for 3 item-popularity buckets. Figures 4 and 3 show hr@K and coverage@K within each bucket as α varies. At low α, almost all hits come from

Bucket 1 and Bucket 3 performance is near zero. As α exceeds 0.2, performance in Buckets 2 and 3 rises sharply (e.g., coverage@10 in Bucket 3 grows from 0 to 0.015), while Bucket 1 performance declines only modestly.

We observe the same qualitative behavior on Yelp, Gowalla, and Amazon Beauty: a low-α regime tuned for head-item precision versus a high-α regime that enhances long-tail coverage. MovieLens (ML-1M), however, shows weaker bucket separation: decreasing α improves head-bucket metrics but degrades both mid- and tail-bucket performance in tandem. We attribute this to the extreme interaction skew in MovieLens, which makes our uniform bucket split conflate heterogeneous user subgroups and obscure the head-vs-tail trade-off.

We further demonstrate the practical aspect of this effect on Figure 5. We provide two setups of our BT-SR method with two different values of α on the Yelp dataset. Both setups allow outperforming the baseline in terms of integral recommendations accuracy yet they yield completely different internal structure of recommendations with respect to item popularity. As demonstrated in the figure, one can select between two regimes. In one regime, corresponding to the lower value of α, the recommendations are steered towards more generic user interests, which is indicated by a higher performance in the first bucket in terms of hr@1 metric. In contrast,
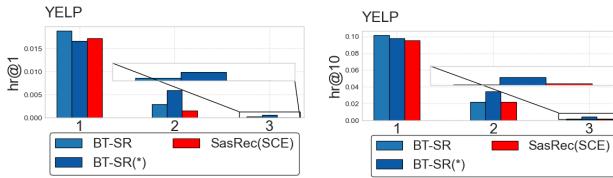
Figure 5: Performance comparison (HR@1 and HR@10) for the BT-SR method on the Yelp dataset under two regimes: $\alpha = 0.1$ favors popular items (high HR@1), while $\alpha = 0.4$ promotes diverse, less popular items early in the list, boosting personalization and maintaining high overall accuracy (HR@10).

	SASRec(BCE)	BT-SR($\alpha = 0.1$)	BT-SR($\alpha = 0.4$)
ndcg@20	0.0176	0.0180	0.0182

Table 3: Integral metric for Figure 7

the second regime with higher α compensates for lower scores in the first bucket by a better performance in the second and third buckets, thus promoting less popular yet relevant recommendations at the beginning of the recommendation list. It helps increasing the diversity of recommendations without compromising the overall accuracy and boosts personalization.

In summary, our framework not only improves overall recommendation quality but also exposes a single, intuitive hyperparameter α that practitioners can tune to balance short-list precision against long-tail diversity. In practice, we find that moderate values of α (e.g., 0.2−0.4) offer the best balance across datasets.

Impact of Augmentation Strategy. We next assess the impact of our behaviorally aligned augmentation strategy. Following prior contrastive learning approaches such as CL4SRec [35], we experiment with perturbation-based augmentations (e.g., random masking, cropping, and reordering). However, we observe that these noise-based augmentations lead to performance degradation in our non-contrastive, multi-task setting. We further evaluate item-based augmentations on the YELP and Kindle Store datasets and find a performance drop of at least 6%. These findings support our choice of goal-preserving augmentations derived from user intent, as described in Section 3.2.

6 Conclusion

We introduce BT-SR, a novel integration of the Barlow Twins redundancy-reduction objective into Transformer-based sequential recommenders. BT-SR outperforms six strong baselines, mitigates popularity bias, and produces sharper, more discriminative user embeddings. Moreover, our method allows adapting to different trade-offs between accuracy and diversity of recommendations, therefore better suiting various user engagement scenarios used in real applications. These results demonstrate that non-contrastive

self-supervision offers a compact, effective pathway toward fairer, high-performing recommendation models.

References

[1] Nabiha Asghar. 2016. Yelp dataset challenge: review rating prediction. (2016). arXiv: 1605.05362 [cs.CL].

[2] Adrien Bardes, Jean Ponce, and Yann LeCun. 2022. VICReg: variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*. https://openreview.net/forum?id=xm6YD62D1Ub.

[3] Miaomiao Cai, Lei Chen, Yifan Wang, Haoyue Bai, Peijie Sun, Le Wu, Min Zhang, and Meng Wang. 2024. Popularity-aware alignment and contrast for mitigating popularity bias. (2024). https://arxiv.org/abs/2405.20718 arXiv: 2405.20718 [cs.IR].

[4] Rocío Cañamares and Pablo Castells. 2020. On target item sampling in offline recommender system evaluation. In *Proceedings of the 14th ACM Conference on Recommender Systems* (RecSys '20). Association for Computing Machinery, Virtual Event, Brazil, 259–268. ISBN: 9781450375832. doi:10.1145/3383313.3412259.

[5] Soumitri Chattopadhyay, Soham Ganguly, Sreejit Chaudhury, Sayan Nag, and Samiran Chattopadhyay. 2023. An evaluation of non-contrastive self-supervised learning for federated medical image analysis. (2023). https://arxiv.org/abs/2303.05556 arXiv: 2303.05556 [cs.CV].

[6] Xinlei Chen and Kaiming He. 2020. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*.

[7] Mingyue Cheng, Fajie Yuan, Qi Liu, Xin Xin, and Enhong Chen. 2021. Learning transferable user representations with sequential behaviors via contrastive pre-training. *2021 IEEE International Conference on Data Mining (ICDM)*, 51–60. https://api.semanticscholar.org/CorpusID:244128946.

[8] Eunjoon Cho, Seth A. Myers, and Jure Leskovec. 2011. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD '11). Association for Computing Machinery, San Diego, California, USA, 1082–1090. ISBN: 9781450308137. doi:10.1145/2020408.2020579.

[9] Alexander Dallmann, Daniel Zoller, and Andreas Hotho. 2021. A case study on sampling strategies for evaluating neural sequential item recommendation models. In *Proceedings of the 15th ACM Conference on Recommender Systems* (RecSys '21). Association for Computing Machinery, Amsterdam, Netherlands, 505–514. ISBN: 9781450384582. doi:10.1145/3460231.3475943.

[10] Dario Di Palma, Felice Antonio Merra, Maurizio Sfilio, Vito Walter Anelli, Fedelucio Narducci, and Tommaso Di Noia. 2025. Do llms memorize recommendation datasets? a preliminary study on movielens-1m. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR '25). Association for Computing Machinery, Padua, Italy, 2582–2586. ISBN: 9798400715921. doi:10.1145/3726302.3730178.

[11] Hanwen Du, Hui Shi, Pengpeng Zhao, Deqing Wang, Victor S. Sheng, Yanchi Liu, Guanfeng Liu, and Lei Zhao. 2022. Contrastive learning with bidirectional transformers for sequential recommendation. (2022). https://arxiv.org/abs/2208.03895 arXiv: 2208.03895 [cs.IR].

[12] Evgeny Frolov and I. Oseledets. 2022. Tensor-based sequential learning via hankel matrix representation for next item recommendations. *IEEE Access*, 11, 6357–6371. https://api.semanticscholar.org/CorpusID:254563829.

[13] Jean-Bastien Grill et al. 2020. Bootstrap your own latent a new approach to self-supervised learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (NIPS '20) Article 1786. Curran Associates Inc., Vancouver, BC, Canada, 14 pages. ISBN: 9781713829546.

[14] Xingzhuo Guo, Junwei Pan, Ximei Wang, Baixu Chen, Jie Jiang, and Mingsheng Long. 2024. On the embedding collapse when scaling up recommendation models. (2024). https://arxiv.org/abs/2310.04400 arXiv: 2310.04400 [cs.LG].

[15] F. Maxwell Harper and Joseph A. Konstan. 2015. The movielens datasets: history and context. *ACM Trans. Interact. Intell. Syst.*, 5, 4, Article 19, (Dec. 2015), 19 pages. doi:10.1145/2827872.

[16] Ruining He, Chen Fang, Zhaowen Wang, and Julian McAuley. 2016. Vista: a visually, socially, and temporally-aware model for artistic recommendation. In *Proceedings of the 10th ACM Conference on Recommender Systems* (RecSys '16). Association for Computing Machinery, Boston, Massachusetts, USA, 309–316. ISBN: 9781450340359. doi:10.1145/2959100.2959152.

[17] Wang-Cheng Kang and Julian J. McAuley. 2018. Self-attentive sequential recommendation. *CoRR*, abs/1808.09781. http://arxiv.org/abs/1808.09781 arXiv: 1808.09781.

[18] Anton Klenitskiy and Alexey Vasilev. 2023. Turning dross into gold loss: is bert4rec really better than sasrec? In *Proceedings of the 17th ACM Conference on Recommender Systems* (RecSys '23). Association for Computing Machinery, Singapore, Singapore, 1120–1125. ISBN: 9798400702419. doi:10.1145/3604915.3610644.

[19] Walid Krichene and Steffen Rendle. 2020. On sampled metrics for item recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (KDD '20). Association for Computing Machinery, Virtual Event, CA, USA, 1748–1757. ISBN: 9781450379984. doi:10.11 45/3394486.3403226.

[20] Gayan K. Kulatilleke, Marius Portmann, and Shekhar S. Chandra. 2022. Efficient block contrastive learning via parameter-free meta-node approximation. (2022). https://arxiv.org/abs/2209.14067 arXiv: 2209.14067 [cs.LG].

[21] Qijiong Liu, Jieming Zhu, Lu Fan, Kun Wang, Hengchang Hu, Wei Guo, Yong Liu, and Xiao-Ming Wu. 2025. Benchmarking llms in recommendation tasks: A comparative evaluation with conventional recommenders. *CoRR*, abs/2503.05493. arXiv: 2503.05493. doi:10.48550/ARXIV.2503.05493.

[22] Yuhan Liu, Lin Ning, Neo Wu, Karan Singhal, Philip Mansfield, Devora Berlowitz, Sushant Prakash, and Bradley Green. 2025. Enhancing user sequence modeling through barlow twins-based self-supervised learning. (May 2025). doi:10.48550 /arXiv.2505.00953.

[23] Zhiwei Liu, Yongjun Chen, Jia Li, Philip S. Yu, Julian McAuley, and Caiming Xiong. 2021. Contrastive self-supervised sequential recommendation with robust augmentation. (2021). https://arxiv.org/abs/2108.06479 arXiv: 2108.064 79 [cs.IR].

[24] Gleb Mezentsev, Danil Gusak, Ivan Oseledets, and Evgeny Frolov. 2024. Scalable cross-entropy loss for sequential recommendations with large item catalogs. In *18th ACM Conference on Recommender Systems* (RecSys '24). ACM, (Oct. 2024), 475–485. doi:10.1145/3640457.3688140.

[25] 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 (EMNLP-IJCNLP)*. Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, (Eds.) Association for Computational Linguistics, Hong Kong, China, (Nov. 2019), 188–197. doi:10.18 653/v1/D19-1018.

[26] Aleksandr V. Petrov and Craig Macdonald. 2024. Transformers for sequential recommendation. In *ECIR (5)*, 369–374. https://doi.org/10.1007/978-3-031-5606 9-9_49.

[27] Ruihong Qiu, Zi Huang, Hongzhi Yin, and Zijian Wang. 2021. Contrastive learning for representation degeneration problem in sequential recommendation. *CoRR*, abs/2110.05730. https://arxiv.org/abs/2110.05730 arXiv: 2110.05730.

[28] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2020. Contrastive learning with hard negative samples. *CoRR*, abs/2010.04592. https://arxiv.org/abs/2010.04592 arXiv: 2010.04592.

[29] Olivier Roy and Martin Vetterli. 2007. The effective rank: a measure of effective dimensionality. In *2007 15th European Signal Processing Conference*, 606–610.

[30] Zhengxiang Shi, Xi Wang, and Aldo Lipani. 2024. Self contrastive learning for session-based recommendation. In *ECIR (1)* (Lecture Notes in Computer Science). Nazli Goharian, Nicola Tonellotto, Yulan He, Aldo Lipani, Graham McDonald, Craig Macdonald, and Iadh Ounis, (Eds.) Vol. 14608. Springer, 3–20. ISBN: 978-3-031-56027-9. http://dblp.uni-trier.de/db/conf/ecir/ecir2024-1.html #ShiWL24.

[31] William Shiao, Zhichun Guo, Tong Zhao, Evangelos E. Papalexakis, Yozen Liu, and Neil Shah. 2023. Link prediction with non-contrastive learning. (2023). https://arxiv.org/abs/2211.14394 arXiv: 2211.14394 [cs.LG].

[32] 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. (2019). https://arxiv.org/abs/1904.06690 arXiv: 1904.06690 [cs.IR].

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762. http://arxiv.org/abs/1706.03762 arXiv: 1706.03762.

[34] Lei Wang, Ee-Peng Lim, Zhiwei Liu, and Tianxiang Zhao. 2022. Explanation guided contrastive learning for sequential recommendation. (2022). https://arxiv.org/abs/2209.01347 arXiv: 2209.01347 [cs.IR].

[35] Xu Xie, Fei Sun, Zhaoyang Liu, Jinyang Gao, Bolin Ding, and Bin Cui. 2020. Contrastive pre-training for sequential recommendation. *CoRR*, abs/2010.14395. https://arxiv.org/abs/2010.14395 arXiv: 2010.14395.

[36] Xu Xie, Fei Sun, Zhaoyang Liu, Shiwen Wu, Jinyang Gao, Bolin Ding, and Bin Cui. 2021. Contrastive learning for sequential recommendation. (2021). https://arxiv.org/abs/2010.14395 arXiv: 2010.14395 [cs.IR].

[37] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. 2021. Barlow twins: self-supervised learning via redundancy reduction. (2021). arXiv: 2103.03230 [cs.CV].

[38] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stephane Deny. 2021. Barlow twins: self-supervised learning via redundancy reduction. In *Proceedings of the 38th International Conference on Machine Learning* (Proceedings of Machine Learning Research). Marina Meila and Tong Zhang, (Eds.) Vol. 139. PMLR, (18–24 Jul 2021), 12310–12320. https://proceedings.mlr.press/v139/zbontar21a.html.

[39] Huimin Zeng, Xiaojie Wang, Anoop Jain, Zhicheng Dou, and Dong Wang. 2025. A non-contrastive learning framework for sequential recommendation with preference-preserving profile generation. In *The Thirteenth International Conference on Learning Representations*. https://openreview.net/forum?id=Ke2 BEL4csm.

[40] Zhijian Zhuo, Yifei Wang, Jinwen Ma, and Yisen Wang. 2023. Towards a unified theoretical understanding of non-contrastive learning via rank differential mechanism. (2023). https://arxiv.org/abs/2303.02387 arXiv: 2303.02387 [cs.LG].

A   Ablation Study

Figure 6 presents the complete results of our ablation study on the parameters α and λ from Equations 7 and 8. Specifically, we set $\alpha = 0$ to entirely remove the Barlow Twins term from the training loss and $\lambda = 0$ to eliminate the decorrelation term.

For sensitivity analysis, we first fix α to its optimal value and vary λ to assess its impact on invariance. Similarly, we fix λ and adjust α to analyze sensitivity with respect to decorrelation. The results demonstrate that both invariance and decorrelation play crucial roles in learning generalizable user representations for sequential recommendation (SR).

We observe dataset-dependent optimal configurations for α and λ. More interestingly, tuning these parameters allows control over recommender behavior, such as the trade-off between the quality of long-tail and short-head recommendations. For instance, on Gowalla, the maximum HR@1 is achieved with $\alpha = 0.5$, while HR@10 peaks at $\alpha = 0.1$. Additionally, increasing α significantly improves Cov@K on YELP and Gowalla. The popularity bucket metrics further (Figure 8) reveal that α can be adjusted to balance recommendation quality across different popularity segments.
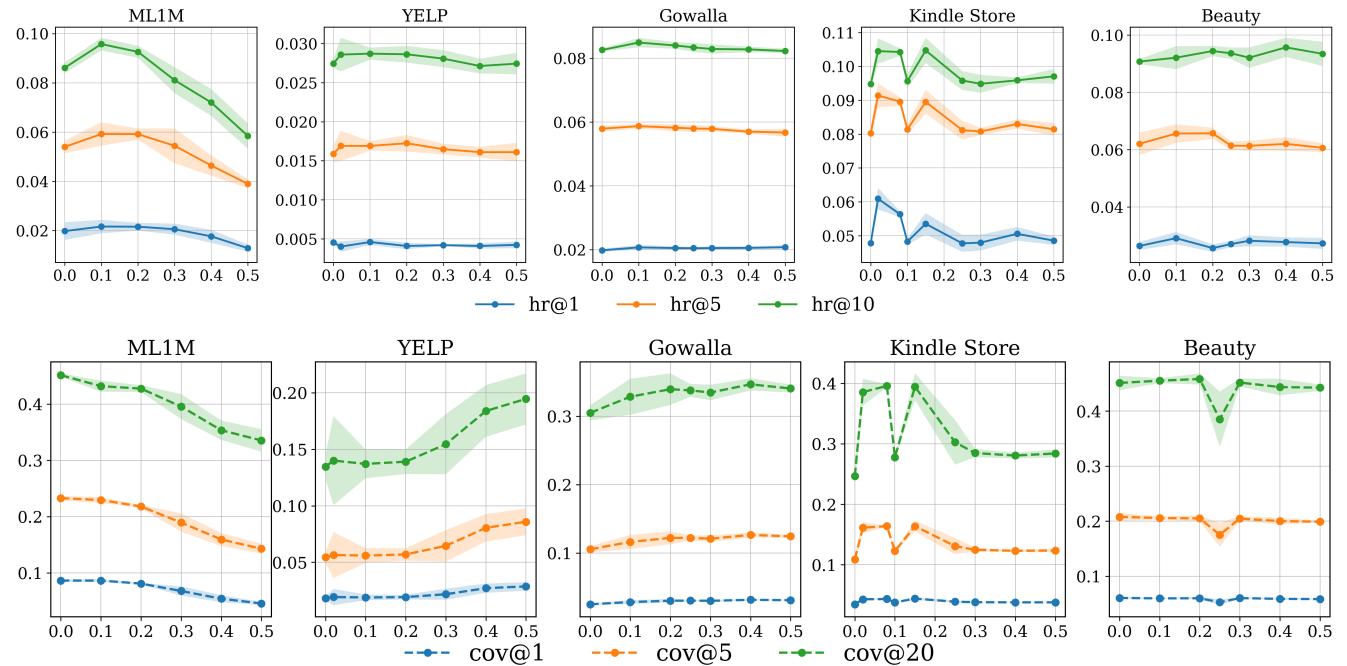
Figure 6: Sensitivity Analysis w.r.t. α of hr@K and cov@K metrics
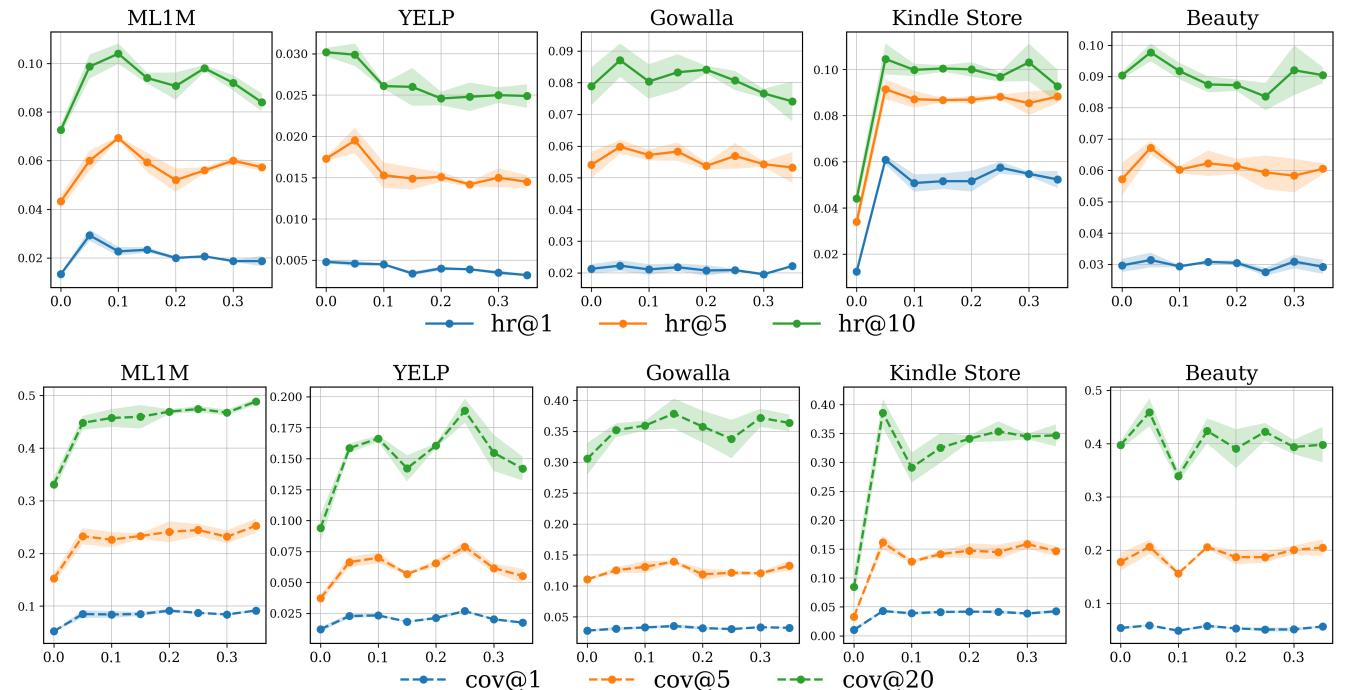

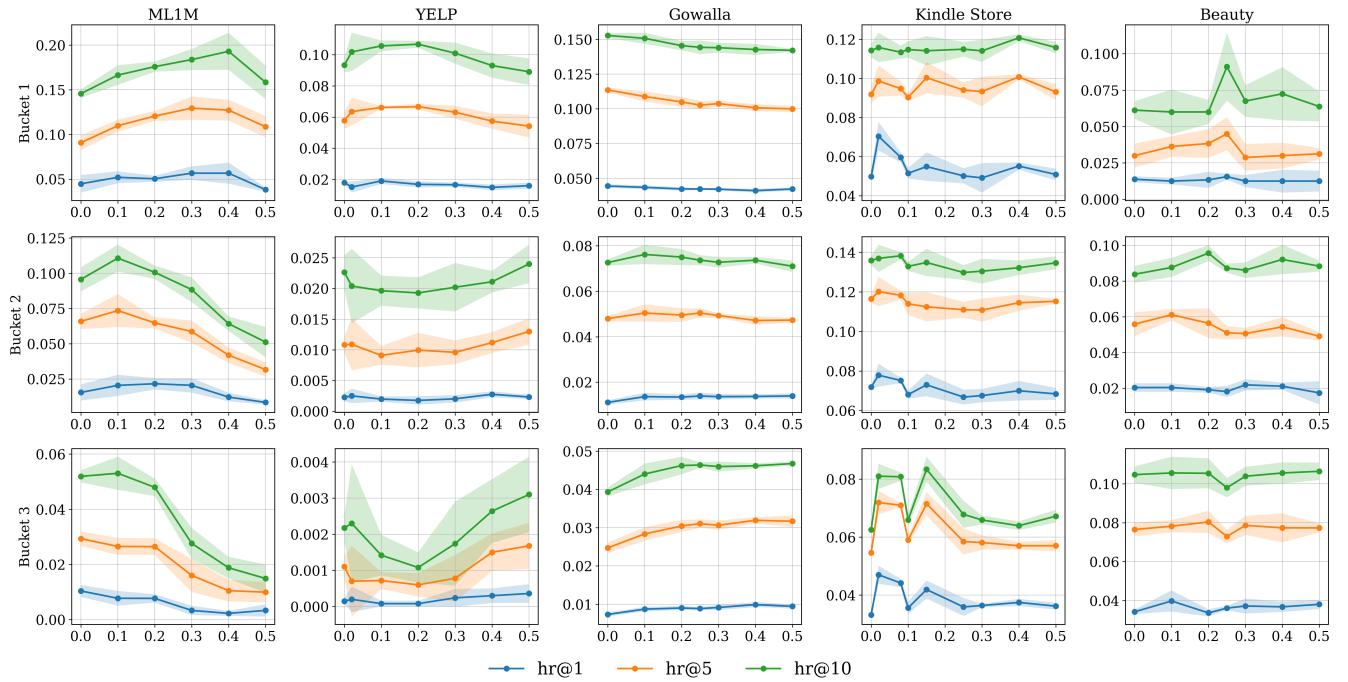Figure 7: Sensitivity Analysis w.r.t. λ of hr@K and cov@K metrics

Figure 8: Sensitivity Analysis w.r.t. α for recommendations over 3 item popularity buckets

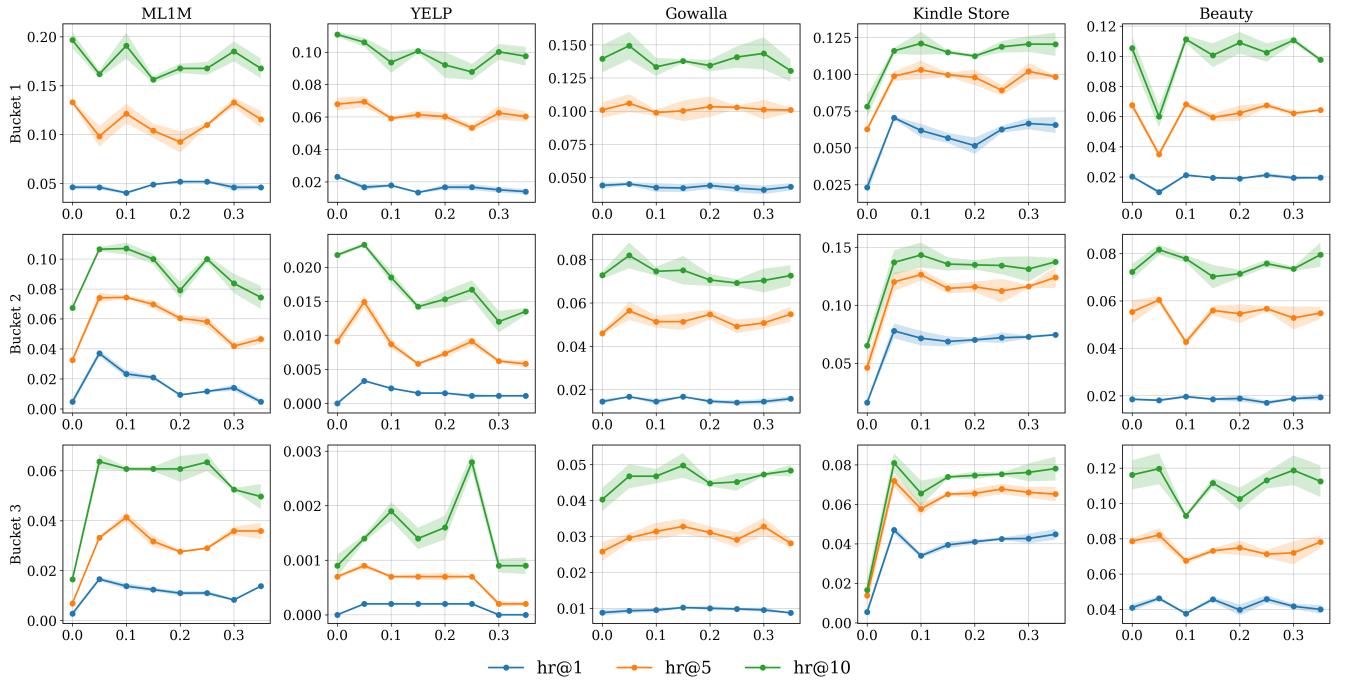Figure 9: Sensitivity Analysis w.r.t. λ for recommendations over 3 item popularity buckets