Dual-View Whitening on Pre-trained Text Embeddings for Sequential Recommendation

Lingzi Zhang[1,2], **Xin Zhou**[2*], **Zhiwei Zeng**[1], **Zhiqi Shen**[1,2]

[1]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[2]Alibaba-NTU Singapore Joint Research Institute, Nanyang Technological University, Singapore
lingzi001@e.ntu.edu.sg, {xin.zhou, zhiwei.zeng, zqshen}@ntu.edu.sg

Abstract

Recent advances in sequential recommendation models have demonstrated the efficacy of integrating pre-trained text embeddings with item ID embeddings to achieve superior performance. However, our study takes a unique perspective by exclusively focusing on the untapped potential of text embeddings, obviating the need for ID embeddings. We begin by implementing a pre-processing strategy known as whitening, which effectively transforms the anisotropic semantic space of pre-trained text embeddings into an isotropic Gaussian distribution. Comprehensive experiments reveal that applying whitening to pre-trained text embeddings in sequential recommendation models significantly enhances performance. Yet, a full whitening operation might break the potential manifold of items with similar text semantics. To retain the original semantics while benefiting from the isotropy of the whitened text features, we propose a Dual-view Whitening method for Sequential Recommendation (DWSRec), which leverages both fully whitened and relaxed whitened item representations as dual views for effective recommendations. We further examine the advantages of our approach through both empirical and theoretical analyses. Experiments on three public benchmark datasets show that DWSRec outperforms state-of-the-art methods for sequential recommendation.

Introduction

The sequential recommendation aims to provide personalized item recommendations to users over time (Zhou et al. 2020; Lei et al. 2021; Hou et al. 2022; Zhang et al. 2023). Recently, there has been an upsurge of interest in developing sequential recommendation methods that incorporate textual information about items, such as product attributes (Xie, Zhou, and Kim 2022), descriptions (Hou et al. 2022; Zhou et al. 2023a), and reviews (Shuai et al. 2022). These approaches typically meld textual features with ID embeddings, underscoring the pivotal role of ID embeddings in the recommendation process. However, there are three primary benefits in exploring sequential recommendation relying exclusively on textual features (denoted as TextSRec): Firstly, it bolsters performance in cold-start scenarios, circumventing the pitfalls of random ID embeddings initialization for new products. Secondly, TextSRec offers superior efficiency over conventional ID-based methods since it obviates the necessity for substantial tensor storage and computational resources, negating the need for a large, continually updated ID embedding matrix. Lastly, in contrast to the non-transferable nature of ID embeddings, text embeddings maintain versatility across various platforms and domains. Despite these benefits, there exists a noticeable gap in research dedicated to pure TextSRec.

Most existing sequential recommendation models (Hou et al. 2022; Wang et al. 2022b; Yuan et al. 2023) usually utilize text embeddings directly from pre-trained language models, such as BERT (Vaswani et al. 2017). Recent research (Yuan et al. 2023) posits that superior performance from TextSRec is achieved exclusively with advanced pre-trained language models. We contend that these models do not optimally harness pre-trained text embeddings for sequential recommendation. This suboptimal use is, in part, due to the representation degeneration issue inherent in BERT embeddings. Recent studies in NLP indicate that BERT sentence embeddings experience this problem, which leads to anisotropy in the vector space, limiting their efficacy in downstream tasks (Li et al. 2020; Su et al. 2021).

To mitigate the anisotropy in pre-trained text embeddings, we adopt a whitening transformation (Kessy, Lewin, and Strimmer 2018) for sequential recommendation. This transformation refines the distribution of the pre-trained text embeddings into an isotropic Gaussian distribution, eliminating correlations among the axes. Our rigorous evaluation has unequivocally demonstrated that sequential models incorporating whitened text features consistently and substantially outperform those utilizing ID embeddings or text embeddings.

While whitening is advantageous for recommendations, over-whitening might adversely affect the manifold of items with analogous textual semantics. As a result, a more nuanced approach could involve partially decorrelating dimensions. This method maintains a higher degree of original textual semantics, albeit at the cost of embedding uniformity (Weng et al. 2022). Although preserving text semantics might seem beneficial for recommendations, our empirical result indicates that complete whitening produces best results compared to varying levels of partial whitening.

To reap the benefits of full whitening while preserving the partial semantics in original text features, we propose a novel Dual-view Whitening method on pre-trained text em-

*Corresponding author

beddings for Sequential Recommendation, named DWSRec. Initially, we employ a dual-view item encoder with a shared projection head, deriving both fully whitened and relaxed whitened representations from pre-trained text features. The former ensures decorrelation across all dimensions, while the latter targets decorrelation within specific dimensional groups. Then, we utilize a decoupled attention-based dual-view transformer to encode sequences composed of fully and relaxed whitening representations. Namely, we generate two sets of key-query pairs in the attention layer. For learning the sequence embedding based on full whitening, it serves as the value and the initial key-query pair, while the relaxed whitening acts as the second key-query pair. Likewise, when learning the sequence embedding with relaxed whitening, it is used as the value and the primary key-query pair, with the full whitening serving as the second pair. Different extents of whitened representations are employed collectively and interchangeably to enhance the attention calculation of the transformer. Lastly, we leverage a dual-view fusion module to adaptively merge these view-specific sequence embeddings as well as item embeddings using two separate weighted attention layers for recommendation.

To assess the effectiveness of DWSRec, we conduct empirical and theoretical analyses on representation uniformity and alignment, conditioning, and information reconstruction. Results show it can improve user and item representation uniformity and alignment, achieving better performance. The conditioning of the transformed item embedding matrix is also enhanced, promoting training stability. A mathematical analysis indicates DWSRec is superior in preserving information, necessitating less data for training input reconstruction. Our contributions are as follows:

- We show through empirical analysis that anisotropy in pre-trained text embeddings limits TextSRec's performance. We address this by using a whitening transformation, significantly enhancing TextSRec's performance.
- Noting that full whitening might affect similar textual semantics, we introduce DWSRec to leverage different degrees of whitening for effective sequential recommendation. We analyze its benefits in representation, conditioning, and information reconstruction.
- Extensive experiments are conducted on three benchmark datasets. Notably, DWSRec outperforms state-of-the-art models across all metrics for all three datasets.

Related Work

Sequential Recommendation Systems One of the earliest approaches is Markov Chain-based models (Rendle, Freudenthaler, and Schmidt-Thieme 2010), which models the probability of transitions between items in a sequence. However, these models often suffer from the cold-start problem and have limited capability of handling complex sequence patterns. Another approach views the sequence as an image and has led to the development of a line of works based on Convolution Neural Network (Tang and Wang 2018; Yuan et al. 2019). Recurrent Neural Network (Donkers, Loepp, and Ziegler 2017; Peng et al. 2021) has shown remarkable performance in utilizing sequential information for recommendation. Graph Neural Networks (Chang et al. 2021; Zhang et al. 2022) have been explored to model complex item transition patterns. Recently, methods based on transformer architecture (Kang and McAuley 2018; Sun et al. 2019; Zhou et al. 2020; Liu et al. 2021; Xie et al. 2022) have shown strong performance in capturing long-range dependencies in a sequence.

Text-enhanced Recommendation Systems Recent works (Zhou et al. 2020; Xie, Zhou, and Kim 2022; Zhang et al. 2019; Hou et al. 2022; Yuan et al. 2023) have attempted to leverage textual data of items, such as descriptions, attributes, or brands of products to improve item representations for recommendations. Some works (Zhou et al. 2020; Xie, Zhou, and Kim 2022) focus on modeling item attributes and optimizing the model with the attribute prediction task. With the fast development of NLP techniques, more works (Zhang et al. 2019; Wang et al. 2022b; Yuan et al. 2023; Zhou and Shen 2023) extract the pre-trained features from product descriptions using pre-trained language models. Although these works achieve promising results, they directly utilize the pre-trained text embeddings without analyzing their potential problems. Note that UniSRec (Hou et al. 2022) uses a linear transformation to address text anisotropy. However, our experimental results show that this method does not necessarily yield whitened outputs that remove correlation across feature dimensions, thereby resulting in suboptimal performance.

Whitening Whitening, or decorrelation, avoids feature dimension collapse by decorrelating each dimension (Kessy, Lewin, and Strimmer 2018). One of the earliest approaches to whitening is Principal Component Analysis (PCA). Compared with PCA, Zero-phase Component Analysis (ZCA) (Bell and Sejnowski 1997) whitening introduces an additional rotation back to the original coordinate system. Cholesky Decomposition (CD) (Dereniowski and Kubale 2004) whitening proposed by (Siarohin, Sangineto, and Sebe 2019) decomposes the covariance matrix into a lower triangular matrix and its conjugate transpose. Recently, UniSRec (Hou et al. 2022) adopts a parametric whitening (PW) method which uses a linear layer for whitening transformation for better generalizability. In the field of deep learning, prior research efforts (Ioffe and Szegedy 2015; Huang et al. 2018; Hua et al. 2021) explore the application of whitening to the activation of intermediate layers in neural networks. Lately, another research direction (Ermolov et al. 2021; Weng et al. 2022; Bardes, Ponce, and LeCun 2022) has emerged, focusing on employing whitening for self-supervised learning, which seeks to avoid the collapse of augmented representations into a single point. Different from these studies, our work leverages different extents of decorrelation strength during the whitening process of pre-trained text embeddings for sequential recommendations.

Preliminaries

Task Formulation In sequential recommendation, we denote a set of users as $\mathcal{U}$ with size $|\mathcal{U}|$ and a set of items as $\mathcal{V}$ with size $|\mathcal{V}|$. Each user in $\mathcal{U}$ is associated with a sequence

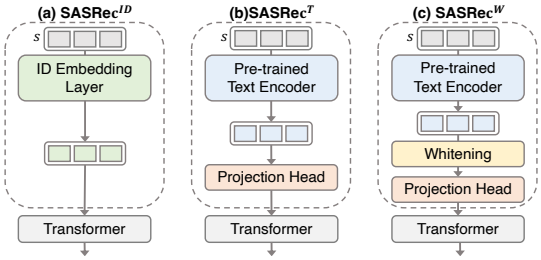


Figure 1: Three variations of SASRec.

of items $s = \{v_1, \cdots, v_{|s|}\}$ where $v_t \in \mathcal{V}$ denotes the item that the user has interacted with at the t-th timestamp. $|s|$ is the sequence length. The objective is to consider the past item sequences of a user and predict the next item ($v_{|s|+1}$) that the user is likely to adopt.

Whitening Transformation Whitening (Kessy, Lewin, and Strimmer 2018) is a linear transformation to project elements of a feature matrix onto a spherical distribution. Here, we perform ZCA whitening on pre-trained text features of all items, denoted as $\mathbf{X} \in \mathbb{R}^{d_t \times |\mathcal{V}|}$. d_t is the feature dimension. Specifically, the output of ZCA whitening is:

$$\mathbf{Z} = \mathbf{D}\Lambda^{-\frac{1}{2}}\mathbf{D}^\top(\mathbf{X} - \mu \cdot \mathbf{1}^\top), \tag{1}$$

where $\Lambda = \text{diag}(\sigma_1, \cdots, \sigma_{d_t})$ and $\mathbf{D} = [\mathbf{d}_1, \cdots, \mathbf{d}_{d_t}]$ are the eigenvalues and associated eigenvectors of $\mathbf{\Sigma} = \mathbf{D}\Lambda\mathbf{D}^\top$. $\mathbf{\Sigma} = \frac{1}{|\mathcal{V}|}(\mathbf{X} - \mu \cdot \mathbf{1}^\top)(\mathbf{X} - \mu \cdot \mathbf{1}^\top)^\top + \epsilon\mathbf{I}$ is the covariance matrix of the centered input $\mathbf{X}$. The ZCA output has the property of $\mathbf{Z}\mathbf{Z}^\top = \mathbf{I}_{d_t}$ to make $\mathbf{X}$ fully whitened. Given that ZCA assumes a full rank covariance matrix, conducting ZCA on all items' feature matrix $\mathbf{X}$ in which the cardinality of $\mathcal{I}$ significantly exceeds the dimensionality of d_t (*i.e.*, $|\mathcal{V}| \gg d_t$) ensures that $\mathbf{\Sigma}$ is full rank. ZCA whitening is stringent, decorrelating all dimensions of $\mathbf{X}$. We can relax the whitening with "group whitening" by standardizing covariance matrices within dimensional groups (Huang et al. 2018; Hua et al. 2021). Given the input $\mathbf{X} \in \mathbb{R}^{d_t \times |\mathcal{V}|}$, the output is a matrix $\mathbf{Y} \in \mathbb{R}^{d_t \times |\mathcal{V}|}$:

$$\mathbf{Y}^{[h]} = \text{ZCA}(\mathbf{X}^{[h]}), \tag{2}$$

where $\mathbf{X}^{[h]} = \left(\left(\mathbf{X}_{(h-1)\cdot\frac{d_t}{G}+1}\right)^\top, \cdots, \left(\mathbf{X}_{h\cdot\frac{d_t}{G}}\right)^\top\right)^\top \in \mathbb{R}^{\frac{d_t}{G} \times |\mathcal{V}|}$ and $\mathbf{Y}^{[h]} \in \mathbb{R}^{\frac{d_t}{G} \times |\mathcal{V}|}$. Namely, $\mathbf{Y}$ is derived by dividing all feature dimensions d_t into G groups and applying ZCA whitening to each group independently. When G=1, full whitening that decorrelates all dimensions of $\mathbf{X}$ is performed. When G>1, the whitening is relaxed, decorrelating only intra-group dimensions.

SASRec^{ID} & SASRec^{T} & SASRec^{W} SASRec (Kang and McAuley 2018), by utilizing the self-attention mechanism, serves as a foundational model for state-of-the-art sequential recommendation methods (Qiu et al. 2022; Hou et al. 2022). Hence, we examine three variants of SASRec (Kang and McAuley 2018) to illustrate the anisotropy issue inherent in pre-trained text embeddings: SASRec^{ID}, SASRec^{T}, and SASRec^{W}. Their primary distinction is in item embedding encoding before the transformer layers as presented in Fig. 1. **SASRec^{ID}** is the base SASRec. It employs a randomly initialized ID embedding matrix, denoted as $\mathbf{E} \in \mathbb{R}^{d \times |\mathcal{V}|}$, to depict items. d indicates the embedding size. **SASRec^{T}** presumes that items possess textual information. The item embeddings are initialized using pre-trained text features $\mathbf{X}$, and are not updated throughout the training process. $\mathbf{X}$ is then passed to an MLP projector with two hidden layers and ReLU activations to reduce dimensionality. **SASRec^{W}** mirrors SASRec^{T} but pre-processes $\mathbf{X}$ with the whitening transformation on all items.

Model	Arts		Toys		Tools	
	R@50	N@50	R@50	N@50	R@50	N@50
SASRec^{ID}	0.1967	0.0887	0.1581	0.0558	0.0941	0.0463
SASRec^{T}	0.2129	0.0850	0.1542	0.0539	0.1055	0.0448
SASRec^{W}	**0.2348**	**0.0939**	**0.1798**	**0.0639**	**0.1196**	**0.0519**

Table 1: Performance of SASRec^{ID}, SASRec^{T}, SASRec^{W}.

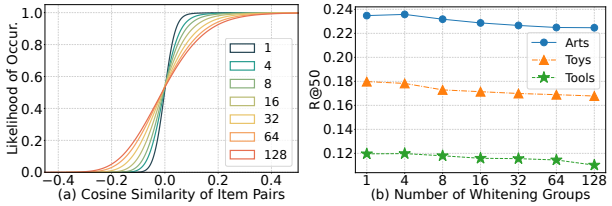


Figure 2: (a) CDF plot of item pairs for Arts. (b) Performance of SASRec^{W} *w.r.t* the number of whitening groups.

Methods and Main Results

Full Whitening Enhances Performance

Recent studies in NLP highlight that BERT sentence embeddings often degenerate into an anisotropic shape (Li et al. 2020; Su et al. 2021). Such degeneration compromises the performance of downstream language modeling tasks. Given the frequent use of BERT embeddings in text-based recommendation models, we show that the whitening of BERT embeddings, which addresses the anisotropy issue, substantially improves sequential recommendation performance.

To validate our claim, we evaluate the performance of SASRec^{ID}, SASRec^{T}, and SASRec^{W} (G=1, using fully whitened representations). As in Table 1, SASRec^{W} significantly outperforms the others across all datasets, which can be attributed to the whitening transformation that addresses the anisotropy issue in text features. It notably improves performance without adding any trainable parameters.

Relaxed Whitening Retains Text Semantics yet Degrades Performance

Using fully whitened representations with G=1 may negatively impact the manifold of items with similar textual semantics compared to the original text representation. We illustrate this by visualizing the Cumulative Distribution

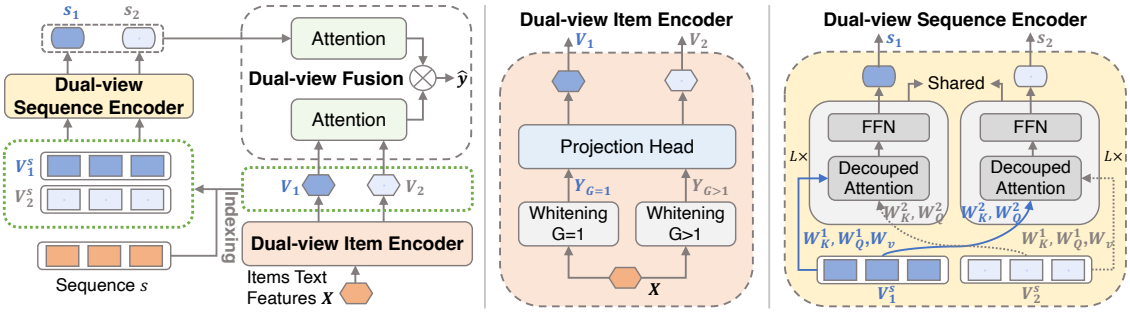


Figure 3: The illustration of DWSRec: 1) Dual-view Item Encoder extracts item embeddings using both full and relaxed whitening; 2) Dual-view Sequence Encoder optimizes attention calculations by leveraging different degrees of whitening; 3) Dual-view Fusion to merge view-specific embeddings via attention layers.

Function (CDF) plot, which shows the cosine similarities between item pairs with varying extents of whitening on text embeddings of Arts (*i.e.*, different G) in Fig. 2(a). The legend denotes the G involved in whitening transformation, with a smaller G implying a higher degree of decorrelation and stronger suppression of redundant information. As Fig.2(a) shows, weaker whitening results in a broader, less concentrated CDF line, preserving textual semantics through increasingly similar item representations.

While retaining text semantics is intuitively beneficial for recommendation tasks, it can lead to cluttered item embedding distributions. As G increases, the distribution becomes increasingly non-uniform. As presented in Fig. 2(b), experiments on SASRec^W using G from $\{1, 4, 8, 16, 32, 64, 128\}$ show that a smaller G achieves optimal performance. This suggests that enhanced decorrelation improves representation learning in the sequential recommendation.

DWSRec: Dual-view Whitening for Sequential Recommendation

Previous sections demonstrate that employing whitening techniques to decorrelate dimensions resolves the feature degeneration problem, thereby improving sequential recommendation performance. However, straying from full whitening to retain original text semantics leads to suboptimal results. To maximize whitening benefits while preserving some original semantics, we introduce DWSRec, which utilizes varying degrees of whitened representations as multi-faceted views to enhance user and item modeling. The illustration of DWSRec is shown in Fig. 3.

Dual-view Item Encoder We apply both the full and relaxed whitening on item text features $\mathbf{X}$. The resultant embeddings are denoted as $\mathbf{Y}_{G=1}$ and $\mathbf{Y}_{G>1}$, respectively. Then, we map both $\mathbf{Y}_{G=1}$ and $\mathbf{Y}_{G>1}$ into a latent space using a shared projection head consisting of two MLP layers. The outputs from the projection are denoted as $\mathbf{V}_1 \in \mathbb{R}^{d\times|\mathcal{V}|}$ and $\mathbf{V}_2 \in \mathbb{R}^{d\times|\mathcal{V}|}$ for $\mathbf{Y}_{G=1}$ and $\mathbf{Y}_{G>1}$ respectively.

Dual-view Sequence Encoder The transformer model (Vaswani et al. 2017) has become a preeminent method for sequence encoding. However, its conventional design primarily focuses on the self-correlation within a single type of sequence. Inspired by (Xie, Zhou, and Kim 2022), we adapt the transformer with a decoupled attention mechanism to encode sequences from diverse whitening views. This refines the attention computation for a given view by leveraging an alternative view, thus enabling adaptive gradient adjustments to capture the nuances of different whitenings.

As shown in Fig. 3, the dual-view sequence encoder comprises two transformer modules with shared parameters. A transformer module contains multiple stacked blocks, with each block consisting of a decoupled attention layer and a feed-forward layer (FFN). These blocks utilize two input types generated from the dual-view item encoder: full and relaxed whitening. Given a user sequence s, the embeddings for all items in s retrieved from $\mathbf{V}_1$ and $\mathbf{V}_2$ are denoted as $\mathbf{V}_1^s \in \mathbb{R}^{d\times|s|}$ and $\mathbf{V}_2^s \in \mathbb{R}^{d\times|s|}$ respectively. For each decoupled attention layer, two sets of key-query projection matrices and one value projection matrix are generated for each of h heads. They are denoted as $\mathbf{W}_K^{1,i}$, $\mathbf{W}_Q^{1,i}$, $\mathbf{W}_K^{2,i}$, $\mathbf{W}_Q^{2,i}$, $\mathbf{W}_V^i \in \mathbb{R}^{d_h\times d}$, $i \in [h]$, and $d_h = d/h$. We first learn the sequence representation using $\mathbf{V}_1^s$ and correlate it with $\mathbf{V}_2^s$. We use $\mathbf{V}_1^s$ as the input to the value and first key-query pair projections, whereas $\mathbf{V}_2^s$ is used for the second key-query pair projections in attention computation. Specifically, attention matrices of a head i for $\mathbf{V}_1^s$ given $\mathbf{V}_2^s$ are formulated as:

$$att_1^i = (\mathbf{W}_Q^{1,i}\mathbf{V}_1^s)(\mathbf{W}_K^{1,i}\mathbf{V}_1^s)^\top, att_2^i = (\mathbf{W}_Q^{2,i}\mathbf{V}_2^s)(\mathbf{W}_K^{2,i}\mathbf{V}_2^s)^\top.$$

These matrices are fused with a function $\mathcal{F}$ using addition to produce outputs for each head:

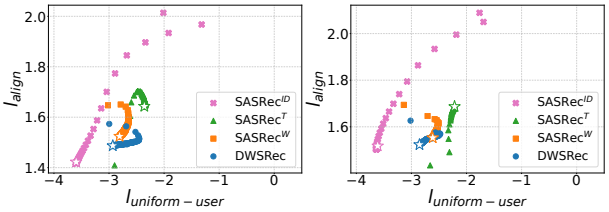
$$head^i = \sigma(\frac{\mathcal{F}(att_1^i, att_2^i)}{\sqrt{d}})(\mathbf{W}_V^i\mathbf{V}_1^s). \tag{3}$$

The concatenated outputs of all attention heads serve as the input to the FFN. After L decoupled transformer layers, following (Zhou et al. 2020), the embedding of the sequence's last item represents the sequence and is denoted as $\mathbf{s}_1 \in \mathbb{R}^d$.

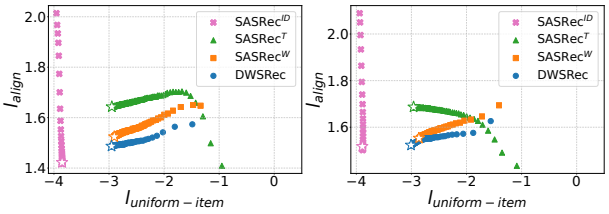
Next, we derive the sequence representation using $\mathbf{V}_2^s$ and correlate it with $\mathbf{V}_1^s$. Here, $\mathbf{V}_2^s$ serves as input to the value and first key-query pair projections, whereas $\mathbf{V}_1^s$ is employed for the second key-query pair projections. The corresponding attention matrices for $\mathbf{V}_2^s$ in relation to $\mathbf{V}_1^s$ are detailed below:

$$\widehat{att}_1^i = (\mathbf{W}_Q^{1,i}\mathbf{V}_2^s)(\mathbf{W}_K^{1,i}\mathbf{V}_2^s)^\top, \widehat{att}_2^i = (\mathbf{W}_Q^{2,i}\mathbf{V}_1^s)(\mathbf{W}_K^{2,i}\mathbf{V}_1^s)^\top,$$

$$\widehat{head}^i = \sigma(\frac{\mathcal{F}(\widehat{att}_1^i, \widehat{att}_2^i)}{\sqrt{d}})(\mathbf{W}_V^i\mathbf{V}_2^s). \tag{4}$$

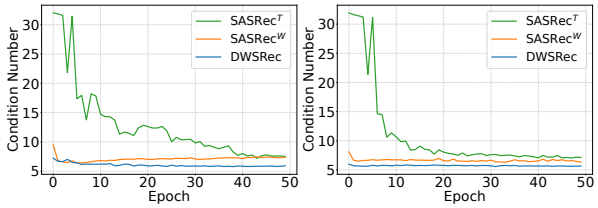


(a) Arts-Users (b) Toys-Users

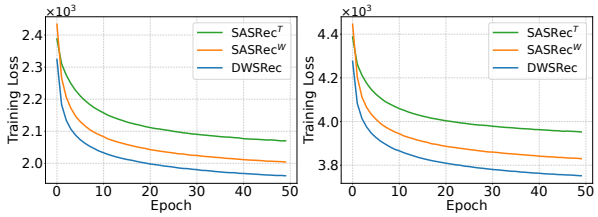


(c) Arts-Items (d) Toys-Items

Figure 4: $l_{align} - l_{uniform}$ plots for representations of users during training. We visualize these two metrics in each epoch, and the stars indicate the last converged epoch. For l_{align} and $l_{uniform}$, lower numbers are better.



(a) Arts-Condition Number (b) Toys-Condition Number



(c) Arts-Training Loss (d) Toys-Training Loss

Figure 5: Conditioning analysis for SASRec^T, SASRec^W, and DWSRec. We plot the condition number (log-scale) calculated for the item embedding matrix after projection and the training loss with respect to each epoch.

The output sequence embedding is denoted as $\mathbf{s}_2 \in \mathbb{R}^d$.

Our proposed dual-view sequence encoder leverages diverse views of whitened embeddings to interchangeably update the attention heads within the transformer model. This approach can be perceived as generating augmented data instances, thereby enriching the training process and enhancing overall performance.

Dual-view Fusion Given two views of sequence embeddings, $\mathbf{s}_1$ and $\mathbf{s}_2$, and two views of item embeddings $\mathbf{V}_1$ and $\mathbf{V}_2$, we aim to adaptively merge these view-specific embeddings using learnable attentive weights. The resulting aggregated sequence representation, $\mathbf{s}$, is computed as follows:

$$\mathbf{s} = \sum_{\mathbf{e}_i \in \{\mathbf{s}_1, \mathbf{s}_2\}} f(\mathbf{e}_i)\mathbf{e}_i, \quad f(\mathbf{e}_i) = \frac{\exp(\mathbf{a}^\top \cdot \mathbf{W}_a \mathbf{e}_i)}{\sum_i \exp(\mathbf{a}^\top \cdot \mathbf{W}_a \mathbf{e}_i)},$$

where $\mathbf{a} \in \mathbb{R}^d$ and $\mathbf{W}_a \in \mathbb{R}^{d \times d}$ are trainable attention parameters. Similarly, the aggregated item representation $\mathbf{V}$ is computed as:

$$\mathbf{V} = \sum_{\mathbf{E}_i \in \{\mathbf{V}_1, \mathbf{V}_2\}} f(\mathbf{E}_i)\mathbf{E}_i, \quad f(\mathbf{E}_i) = \frac{\exp(\mathbf{b}^\top \cdot \mathbf{W}_b \mathbf{E}_i)}{\sum_i \exp(\mathbf{b}^\top \cdot \mathbf{W}_b \mathbf{E}_i)},$$

where $\mathbf{b} \in \mathbb{R}^d$ and $\mathbf{W}_b \in \mathbb{R}^{d \times d}$ are trainable attention parameters. With $\mathbf{V} \in \mathbb{R}^{d \times |\mathcal{V}|}$ and $\mathbf{s} \in \mathbb{R}^d$, the model is optimized with the cross-entropy loss:

$$\mathcal{L} = -\log(\hat{\mathbf{y}})\,\text{onehot}(\mathbf{y}), \quad \hat{\mathbf{y}} = \text{softmax}(\mathbf{V}^\top \mathbf{s}), \qquad (5)$$

where $\mathbf{y}$ is the ground-truth next item given a user sequence.

Discussion and Analysis

Uniformity and Alignment We analyze the user embedding $\mathbf{s}_u$ and the target item embedding $\mathbf{v}_i$ retrieved from $\mathbf{V}$ concerning their uniformity and alignment under the assumption that vectors are normalized (Wang and Isola 2020; Wang et al. 2022a). We denote the alignment between representations of positive-related user-item pairs (*i.e.*, $\langle \mathbf{s}_u, \mathbf{v}_i \rangle$) as l_{align}, the uniformity within user representations $\mathbf{s}_u$ as $l_{uniform-user}$, and the uniformity of item representations $\mathbf{v}_i$ as $l_{uniform-item}$. We visualize these three metrics for four methods (*i.e.*, SASRec^{ID}, SASRec^T, SASRec^W, DWSRec) in Fig.4 for Arts and Toys datasets. Both SASRec^W and DWSRec achieve better alignment and user uniformity compared to SASRec^T, leading to enhanced performance. DWSRec further improves all metrics compared with SASRec^W and achieves the best performance. Notably, SASRec^{ID} shows high uniformity but poor performance, suggesting that excessive uniformity can impair recommendations by overlooking semantically similar items.

Conditioning Analysis We highlight the benefits of SASRec^W and DWSRec in achieving improved conditioning of item embedding matrix $\mathbf{V}$. Given the covariance matrix $\mathbf{A}$ of $\mathbf{V}$, we measure its conditioning by the condition number (Song, Sebe, and Wang 2022): $\kappa(\mathbf{A}) = \lambda_{max}(\mathbf{A})\lambda^{-1}_{min}(\mathbf{A})$, where $\lambda(\cdot)$ is the eigenvalue of the matrix. Well-conditioned matrices possess a low condition number, in contrast to ill-conditioned matrices with high values. In neural networks, ill-conditioned covariance matrices adversely impact training stability and optimization. Fig. 5(a)-(d) display the evolution of condition numbers and training loss over training epochs for Arts and Toys. Results indicate that SASRec^W and DWSRec converge faster and exhibit superior conditioning than SASRec^T. This shows the efficiency of whitening transformation in streamlining the optimization process, with DWSRec exhibiting the best conditioning and achieving the fastest convergence rate.

More Preserved Information in DWSRec When $\mathbf{X}$ is fully whitened with transformation $\mathbf{Q}$, we obtain $\hat{\mathbf{X}} = \mathbf{QX} = [\mathbf{I} \cdots]$. Here, $\mathbf{Q} \in \mathbb{R}^{d \times d}$ is the inverse of the sub-

Model	Arts				Toys				Tools			
	R@20	R@50	N@20	N@50	R@20	R@50	N@20	N@50	R@20	R@50	N@20	N@50
GRCN	0.0851	0.1296	0.0411	0.0499	0.0651	0.0981	0.0304	0.0369	0.0452	0.0682	0.0234	0.0280
BM3	0.1233	0.1782	0.0642	0.0750	0.0965	0.1383	0.0478	0.0560	0.0530	0.0714	0.0299	0.0335
SASRec^{ID}	0.1410	0.1967	0.0776	0.0887	0.1121	0.1581	0.0467	0.0558	0.0712	0.0941	0.0418	0.0463
HGN	0.1293	0.1880	0.0693	0.0810	0.0983	0.1466	0.0435	0.0530	0.0647	0.0902	0.0375	0.0425
CL4SRec	0.1388	0.1967	0.0653	0.0768	0.1094	0.1609	0.0426	0.0528	0.0781	0.1027	0.0385	0.0433
SASRec^{T}	0.1476	0.2129	0.0721	0.0850	0.0983	0.1542	0.0429	0.0539	0.0739	0.1055	0.0386	0.0448
SASRec^{T+ID}	0.1435	0.2009	0.0766	0.0879	0.1163	0.1664	0.0511	0.0610	0.0728	0.0954	0.0445	0.0490
FDSA	0.1284	0.1788	0.0785	0.0888	0.0895	0.1242	0.0475	0.0543	0.0633	0.0812	0.0432	0.0468
S^3-Rec	0.1411	0.2007	0.0762	0.0880	0.1068	0.1533	0.0488	0.0581	0.0707	0.0943	0.0424	0.0470
DIF-SR	0.1510	0.2126	0.0701	0.0823	0.1176	0.1663	0.0487	0.0584	0.0732	0.0955	0.0414	0.0458
UniSRec^{T}	0.1500	0.2165	0.0738	0.0869	0.1042	0.1607	0.0451	0.0563	0.0772	0.1091	0.0407	0.0470
UniSRec^{T+ID}	0.1611	0.2322	0.0774	0.0915	0.1257	0.1801	0.0513	0.0621	0.0828	0.1116	0.0420	0.0477
SASRec^{W}	0.1625	0.2348	0.0796	0.0939*	0.1201	0.1798	0.0521	0.0639*	0.0861*	0.1196*	0.0453	0.0519*
DWSRec	**0.1710***	**0.2419***	**0.0822***	**0.0962***	**0.1307***	**0.1931***	**0.0560***	**0.0683***	**0.0918***	**0.1254***	**0.0479***	**0.0546***

Table 2: Performance of different methods on the warm-start setting. The best results are in boldface, and the best results for baselines are underlined. * denotes SASRec^{W} or DWSRec surpasses the best baseline using a paired t-test ($p < 0.01$).

matrix formed by the first d columns of $\widehat{\mathbf{X}}$. Since the first d columns are deterministic, $\widehat{\mathbf{X}}$ can preserve $(n-d)d$ real values. As an analogy, we infer that DWSRec, which performs group whitening on G groups, can preserve $(n - \frac{d}{G})d$ real values. Thus, it retains more information than SASRec^{W}.

Time Complexity The time complexity of DWSRec mainly stems from the MLP projection head, decoupled attention-based transformers, and attention layers for fusion. The total time complexity is $\mathcal{O}(|s|d_t d + |s|d^2 + |s|^2 d)$, which shows no order of magnitude difference with SASRec^{W}.

Experiments

Experimental Settings

Datasets To evaluate the proposed methods, we conduct experiments on three categories of widely-used Amazon review dataset (Ni, Li, and McAuley 2019): *Arts, Crafts and Sewing*, *Toys and Games*, and *Tools and Instruments*. We abbreviate them as Arts, Toys, and Tools.

Baseline Methods We compare our methods with state-of-the-art models, which fall into three groups: 1) general recommendation models with text features: GRCN (Wei et al. 2020), BM3 (Zhou et al. 2023b)); 2) sequential recommendation models: SASRec^{ID} (Kang and McAuley 2018), HGN (Ma, Kang, and Liu 2019), CL4SRec (Xie et al. 2022); 3) sequential recommendation models with text features: SASRec^{T}, SASRec^{T+ID}, FDSA (Zhang et al. 2019), S^3-Rec (Zhou et al. 2020), DIF-SR (Xie, Zhou, and Kim 2022), UniSRec^{T}, UniSRec^{T+ID} (Hou et al. 2022).

Evaluation 1) Warm-start Settings. Following (Zhou et al. 2020), we keep five-core datasets and discard users and items with fewer than five interactions. We evaluate performance using the *leave-one-out* strategy: for each user, the last item in the interaction sequence is for testing, the second last for validation, and the rest for training. 2) Cold-start Settings. Following (Wei et al. 2021), a subset of items (15% of all items) is randomly selected, and all user-item interactions related to this subset are removed. We preserve sequences containing the aforementioned "cold" items as target items in the validation and testing sets. Since these items are not encountered by the model during training, we can assess the model's capability to generalize to previously unseen items.

Each method is evaluated on the entire item set without sampling to avoid inconsistent results (Krichene and Rendle 2020). The recommendation performance is evaluated by two widely-used metrics, *i.e.*, Recall@K and Normalized Discounted Cumulative Gain@K (respectively denoted by R@K and N@K). K is empirically set to 20 and 50.

Implementation Details All models are implemented by Pytorch (Paszke et al. 2019) and RecBole (Zhao et al. 2021). The Adam optimizer (Kingma and Ba 2015) is used to learn model parameters. We standardize maximum sequence length, embedding size, and batch size at 50, 300, and 1024, respectively, and set the number of self-attention blocks, attention heads, and MLP layers in the projection head at 2. Other hyper-parameters of baseline methods are chosen as per their original papers. For our proposed methods, we tune the learning rate in $\{1e^{-5}, 5e^{-5}, 1e^{-4}, 5e^{-4}, 1e^{-3}\}$ and weight decay in $\{0, 1e^{-3}, 1e^{-4}, 1e^{-6}\}$. The group number G is empirically set to 4. The number of decoupled attention-based transformer layers L is set to 2. To avoid over-fitting, we apply an early stopping strategy when N@20 on the validation data does not increase for 10 epochs.

Performance Comparison

Overall Performance Table 2 presents the performance in warm-start settings. We have the following findings: 1) General recommendation methods using text features perform worse than sequential methods, highlighting the effectiveness of sequence encoders in capturing sequential data

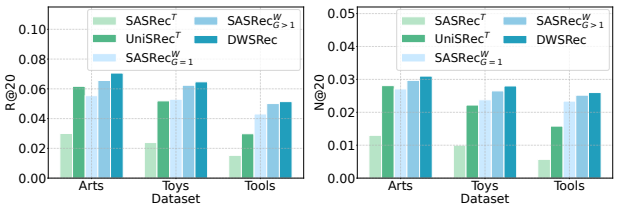


Figure 6: Performance comparison on the cold-start setting.

Model	Arts		Toys		Tools	
	R@20	N@20	R@20	N@20	R@20	N@20
DWSRec	**0.1710**	**0.0822**	**0.1307**	**0.0560**	**0.0918**	**0.0479**
1) w/o $\mathbf{s}_1$	0.1650	0.0802	0.1217	0.0525	0.0866	0.0454
2) w/o $\mathbf{s}_2$	0.1652	0.0808	0.1241	0.0538	0.0869	0.0456
3) w/o D-Trm	0.1647	0.0801	0.1222	0.0527	0.0884	0.0463
4) w/o Attn	0.1702	0.0818	0.1253	0.0533	0.0908	0.0468

Table 3: Ablation study on DWSRec components.

patterns. 2) Sequential methods utilizing text features yield better performance overall, suggesting that text features provide rich semantic information about items, and can enhance recommendation accuracy. 3) Our methods SASRec^W and DWSRec surpass both general recommendation methods and sequential recommendation methods employing text features, attesting to the effectiveness of whitening text features extracted from pre-trained encoders. 4) SASRec^W either matches or exceeds the performance of SASRec^{T+ID} or UniSRec^{T+ID}, suggesting the whitening transformation achieves better results without relying on ID embeddings and simultaneously reduces learnable parameters. 5) DWSRec outperforms all methods, showing that using both fully and relaxed whitened representations with the dual-view encoders enhances user and item representation learning.

Performance in Cold-start Settings We conduct cold-start experiments by comparing them with key baselines, namely SASRec^T and UniSRec^T. From Fig. 6, we can observe: 1) UniSRec^T outperforms SASRec^T, highlighting the merit of the Mixture-of-Experts adaptor with parametric whitening for text embeddings in recommendation. 2) Full whitening SASRec$^W_{G=1}$ is either surpassed by or yields similar performance to UniSRec^T in the Arts and Toys datasets. In contrast, relaxed whitening SASRec$^W_{G>1}$ outperforms SASRec$^W_{G=1}$ and baselines, indicating that relaxed whitening enhances generalization for unseen data, resulting in better performance. 3) DWSRec consistently shows superior performance across datasets, confirming the efficacy of leveraging both full and relaxed whitening with the dual-view encoders in the cold-start context.

Ablation Study

To assess DWSRec designs, we examine four variants outlined in Table 3: 1) w/o $\mathbf{s}_1$: excludes the left segment of the dual-view sequence encoder, using only $\mathbf{s}_2$ for prediction. 2) w/o $\mathbf{s}_2$: omits the right segment, relying solely on $\mathbf{s}_1$ for prediction. 3) w/o D-Trm: uses the conventional transformer to derive sequence embeddings via both full and relaxed whitening separately. 4) w/o Attn: replaces attention fusion layers with element-wise summation. From Table 3, it is observed that removing either $\mathbf{s}_1$ or $\mathbf{s}_2$ diminishes performance, emphasizing the importance of using both embeddings to update the transformer and enhance its generalization. Additionally, removing the decoupled attention leads to further decline, illustrating the effectiveness of the interaction facilitated by the decoupled attention mechanism in harnessing fully and relaxed whitened embeddings. Also, the attention fusion module enhances performance by adaptively combining embeddings from various views.

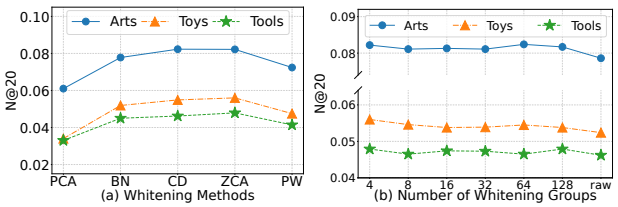


Figure 7: Sensitivity studies of DWSRec *w.r.t*: (a) whitening methods (b) whitening groups.

Parameter Sensitivity Study

Whitening Methods We conduct experiments to explore the effects of different whitening transformations, including non-parametric (PCA, BN, CD, ZCA) and parametric (PW) methods. Results from Fig. 7(a) indicate that PW underperforms compared to non-parametric methods except PCA, possibly because a linear layer does not guarantee a truly whitened output. PCA performs the worst due to stochastic axis swapping issues (Huang et al. 2018), while CD and ZCA surpass BN by further decorrelating axes and have comparable performances across all datasets.

Number of Whitening Groups To evaluate how decorrelation strengths in whitening affect DWSRec, we use the full whitening and adjust G for relaxed whitening in $\{4, 8, 16, 32, 64, 128, \text{raw}\}$, where "raw" denotes unwhitened text features. Fig. 7(b) reveals that the optimal G differs by dataset: the Arts favors a larger G, the Toys favors a smaller one, while the Tools shows no distinct preference. "raw" in general performs the worst across all datasets.

Conclusion

This paper reveals that using text embeddings from pre-trained language models can be sub-optimal in recommendation tasks due to their anisotropic nature. To address this, SASRec^W employs a whitening transformation, converting anisotropic text embeddings into isotropic ones. However, excessive whitening can distort original text semantics. Relying solely on relaxed whitening leads to reduced performance. To harness the best of both, DWSRec incorporates fully and relaxed whitened item representations as dual views to interchangeably update decoupled attention heads of the transformer for enhanced generalization. Experiments show DWSRec's superiority on three benchmark datasets.

Acknowledgments

This research is supported by Alibaba Group and Alibaba-NTU Singapore Joint Research Institute(JRI), Nanyang Technological University, Singapore.

References

Bardes, A.; Ponce, J.; and LeCun, Y. 2022. VICReg: Variance-Invariance-Covariance Regularization For Self-Supervised Learning. In *International Conference on Learning Representations*.

Bell, A. J.; and Sejnowski, T. J. 1997. The "independent components" of natural scenes are edge filters. *Vision Research*, 3327–3338.

Chang, J.; Gao, C.; Zheng, Y.; Hui, Y.; Niu, Y.; Song, Y.; Jin, D.; and Li, Y. 2021. Sequential Recommendation with Graph Neural Networks. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 378–387.

Dereniowski, D.; and Kubale, M. 2004. Cholesky factorization of matrices in parallel and ranking of graphs. In *International Conference on Parallel Processing and Applied Mathematics*, 985–992.

Donkers, T.; Loepp, B.; and Ziegler, J. 2017. Sequential User-based Recurrent Neural Network Recommendations. In *Proceedings of the 11th ACM Conference on Recommender Systems*, 152–160.

Ermolov, A.; Siarohin, A.; Sangineto, E.; and Sebe, N. 2021. Whitening for self-supervised representation learning. In *International Conference on Machine Learning*, 3015–3024.

Hou, Y.; Mu, S.; Zhao, W. X.; Li, Y.; Ding, B.; and Wen, J.-R. 2022. Towards Universal Sequence Representation Learning for Recommender Systems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 585–593.

Hua, T.; Wang, W.; Xue, Z.; Ren, S.; Wang, Y.; and Zhao, H. 2021. On feature decorrelation in self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9598–9608.

Huang, L.; Yang, D.; Lang, B.; and Deng, J. 2018. Decorrelated batch normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 791–800.

Ioffe, S.; and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, 448–456.

Kang, W.-C.; and McAuley, J. 2018. Self-Attentive Sequential Recommendation. In *IEEE International Conference on Data Mining (ICDM)*, 197–206. IEEE.

Kessy, A.; Lewin, A.; and Strimmer, K. 2018. Optimal whitening and decorrelation. *The American Statistician*, 309–314.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations*.

Krichene, W.; and Rendle, S. 2020. On sampled metrics for item recommendation. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1748–1757.

Lei, C.; Liu, Y.; Zhang, L.; Wang, G.; Tang, H.; Li, H.; and Miao, C. 2021. SEMI: A Sequential Multi-Modal Information Transfer Network for E-Commerce Micro-Video Recommendations. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 3161–3171.

Li, B.; Zhou, H.; He, J.; Wang, M.; Yang, Y.; and Li, L. 2020. On the Sentence Embeddings from Pre-trained Language Models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

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*, 1608–1612.

Ma, C.; Kang, P.; and Liu, X. 2019. Hierarchical gating networks for sequential recommendation. In *Proceedings of the 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 825–833.

Ni, J.; Li, J.; and McAuley, J. 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)*, 188–197.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in Neural Information Processing Systems*, 8026–8037.

Peng, B.; Ren, Z.; Parthasarathy, S.; and Ning, X. 2021. HAM: Hybrid Associations Models for Sequential Recommendation. *IEEE Transactions on Knowledge and Data Engineering*.

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*, 813–823.

Rendle, S.; Freudenthaler, C.; and Schmidt-Thieme, L. 2010. Factorizing Personalized Markov Chains for Next-Basket Recommendation. In *Proceedings of the Web Conference 2010*, 811–820.

Shuai, J.; Zhang, K.; Wu, L.; Sun, P.; Hong, R.; Wang, M.; and Li, Y. 2022. A review-aware graph contrastive learning framework for recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1283–1293.

Siarohin, A.; Sangineto, E.; and Sebe, N. 2019. Whitening and Coloring transform for GANs. In *International Conference on Learning Representations*.

Song, Y.; Sebe, N.; and Wang, W. 2022. Improving Covariance Conditioning of the SVD Meta-layer by Orthogonality. In *European Conference on Computer Vision*.

Su, J.; Cao, J.; Liu, W.; and Ou, Y. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

Sun, F.; Liu, J.; Wu, J.; Pei, C.; Lin, X.; Ou, W.; and Jiang, P. 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.

Tang, J.; and Wang, K. 2018. Personalized Top-N Sequential Recommendation via Convolutional Sequence Embedding. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*, 565–573.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. *Advances in Neural Information Processing Systems*, 5998–6008.

Wang, C.; Yu, Y.; Ma, W.; Zhang, M.; Chen, C.; Liu, Y.; and Ma, S. 2022a. Towards Representation Alignment and Uniformity in Collaborative Filtering. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1816–1825.

Wang, J.; Yuan, F.; Cheng, M.; Jose, J. M.; Yu, C.; Kong, B.; Wang, Z.; Hu, B.; and Li, Z. 2022b. TransRec: Learning Transferable Recommendation from Mixture-of-Modality Feedback. *arXiv preprint arXiv:2206.06190*.

Wang, T.; and Isola, P. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, 9929–9939. PMLR.

Wei, Y.; Wang, X.; Li, Q.; Nie, L.; Li, Y.; Li, X.; and Chua, T.-S. 2021. Contrastive learning for cold-start recommendation. In *Proceedings of the 29th ACM International Conference on Multimedia*, 5382–5390.

Wei, Y.; Wang, X.; Nie, L.; He, X.; and Chua, T.-S. 2020. Graph-Refined Convolutional Network for Multimedia Recommendation with Implicit Feedback. In *Proceedings of the 28th ACM International Conference on Multimedia*, 3541–3549.

Weng, X.; Huang, L.; Zhao, L.; Anwer, R.; Khan, S. H.; and Shahbaz Khan, F. 2022. An investigation into whitening loss for self-supervised learning. *Advances in Neural Information Processing Systems*, 29748–29760.

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

Xie, Y.; Zhou, P.; and Kim, S. 2022. Decoupled Side Information Fusion for Sequential Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1611–1621.

Yuan, F.; Karatzoglou, A.; Arapakis, I.; Jose, J. M.; and He, X. 2019. A Simple Convolutional Generative Network for Next Item Recommendation. In *Proceedings of the 12th ACM International Conference on Web Search and Data Mining*, 582–590.

Yuan, Z.; Yuan, F.; Song, Y.; Li, Y.; Fu, J.; Yang, F.; Pan, Y.; and Ni, Y. 2023. Where to Go Next for Recommender Systems? ID-vs. Modality-based recommender models revisited. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2639–2649.

Zhang, L.; Zhou, X.; Zeng, Z.; and Shen, Z. 2023. Multimodal Pre-training Framework for Sequential Recommendation via Contrastive Learning. *arXiv preprint arXiv:2303.11879*.

Zhang, M.; Wu, S.; Yu, X.; Liu, Q.; and Wang, L. 2022. Dynamic graph neural networks for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(5): 4741–4753.

Zhang, T.; Zhao, P.; Liu, Y.; Sheng, V. S.; Xu, J.; Wang, D.; Liu, G.; and Zhou, X. 2019. Feature-level Deeper Self-Attention Network for Sequential Recommendation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 4320–4326.

Zhao, W. X.; Mu, S.; Hou, Y.; Lin, Z.; Chen, Y.; Pan, X.; Li, K.; Lu, Y.; Wang, H.; Tian, C.; et al. 2021. RecBole: Towards a Unified, Comprehensive and Efficient Framework for Recommendation Algorithms. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*, 4653–4664.

Zhou, H.; Zhou, X.; Zhang, L.; and Shen, Z. 2023a. Enhancing Dyadic Relations with Homogeneous Graphs for Multimodal Recommendation. In *ECAI*, volume 372, 3123–3130. IOS Press.

Zhou, K.; Wang, H.; Zhao, W. X.; Zhu, Y.; Wang, S.; Zhang, F.; Wang, Z.; and Wen, J.-R. 2020. S3-Rec: Self-Supervised Learning for Sequential Recommendation with Mutual Information Maximization. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, 1893–1902.

Zhou, X.; and Shen, Z. 2023. A tale of two graphs: Freezing and denoising graph structures for multimodal recommendation. In *Proceedings of the 31st ACM International Conference on Multimedia*, 935–943.

Zhou, X.; Zhou, H.; Liu, Y.; Zeng, Z.; Miao, C.; Wang, P.; You, Y.; and Jiang, F. 2023b. Bootstrap latent representations for multi-modal recommendation. In *Proceedings of the ACM Web Conference 2023*, 845–854.