DIFF: Dual Side-Information Filtering and Fusion for Sequential Recommendation

Hye-young Kim
Sungkyunkwan University
Suwon, Republic of Korea
khyaa3966@skku.edu

Minjin Choi
Samsung Research
Seoul, Republic of Korea
min_jin.choi@samsung.com

Sunkyung Lee
Sungkyunkwan University
Suwon, Republic of Korea
sk1027@skku.edu

Ilwoong Baek
Sungkyunkwan University
Suwon, Republic of Korea
alltun100@skku.edu

Jongwuk Lee*
Sungkyunkwan University
Suwon, Republic of Korea
jongwuklee@skku.edu

Abstract

Side-information Integrated Sequential Recommendation (SISR) benefits from auxiliary item information to infer hidden user preferences, which is particularly effective for sparse interactions and cold-start scenarios. However, existing studies face two main challenges. (i) They fail to remove noisy signals in item sequence and (ii) they underutilize the potential of side-information integration. To tackle these issues, we propose a novel SISR model, ***D**ual Side-**I**nformation **F**iltering and **F**usion (**DIFF**)*, which employs *frequency-based noise filtering* and *dual multi-sequence fusion*. Specifically, we convert the item sequence to the frequency domain to filter out noisy short-term fluctuations in user interests. We then combine early and intermediate fusion to capture diverse relationships across item IDs and attributes. Thanks to our innovative filtering and fusion strategy, DIFF is more robust in learning subtle and complex item correlations in the sequence. DIFF outperforms state-of-the-art SISR models, achieving improvements of up to 14.1% and 12.5% in Recall@20 and NDCG@20 across four benchmark datasets.

CCS Concepts

• **Information systems** → **Recommender systems**.

Keywords

Sequential recommendation; Side-information; Information fusion

ACM Reference Format:
Hye-young Kim, Minjin Choi, Sunkyung Lee, Ilwoong Baek, and Jongwuk Lee. 2025. DIFF: Dual Side-Information Filtering and Fusion for Sequential Recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3726302.3729948

*Corresponding author

SIGIR '25, July 13–18, 2025, Padua, Italy

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3729948

1 Introduction

Sequential Recommendation (SR) [7, 26] aims to predict the next item the user will likely interact with by analyzing past user behavior. It is crucial in various web applications, including e-commerce and streaming services. Existing SR models employ diverse neural architectures to encode an item sequence into user representation. Among them, attention-based models [11, 17, 24] have shown outstanding performance gains by capturing intricate item correlations. However, these models only focus on item IDs, neglecting to utilize valuable item attributes.

Recently, *Side-information Integrated Sequential Recommendation (SISR)* [15, 30, 33] addresses these limitations by modeling the item sequence using side-information. It incorporates various item attributes, *e.g.*, "Brand" and "Category", into the recommendation process. SISR models demonstrate enhanced capability in capturing diverse collaborative signals across items, proving particularly effective in sparse user interaction and cold-start item settings.

Depending on item attribute fusion strategies, existing SISR models can be broadly categorized into three pillars: *early*, *late*, and *intermediate fusion*[1]. Early fusion combines item ID and attribute embeddings before feeding to the model, enabling rich interactions across attributes. However, due to inherent differences in representation spaces, this simple aggregation may result in *information invasion* [15, 29], in which the fused item characteristics become dominated or distorted by the ID or attribute information. Meanwhile, Late fusion [31] encodes IDs and attributes separately, delaying the fusion until the final prediction layer. Although each sequence is modeled effectively, it struggles to capture the correlation between item IDs and attributes. As an alternative, intermediate fusion [15, 27, 29] computes the attention weight of attributes and leverages them to only guide the item correlation, preventing unnecessary interference between IDs and attributes.

While these fusion strategies have shown promising results in leveraging item attributes, they still face two critical challenges that need to be addressed.

(i) Noisy signals in item sequences. Item sequences often contain inconsistent patterns not aligned with the user preferences, *e.g.*, accidental clicks, or short-term intent drift. However, most existing studies utilize all available information to derive user representations, resulting in potential deviation from actual user preferences due to

[1]Although existing work [27] designs it as hybrid fusion, it does not explicitly combine different fusion types. Thus, we rename it intermediate fusion to avoid ambiguity.

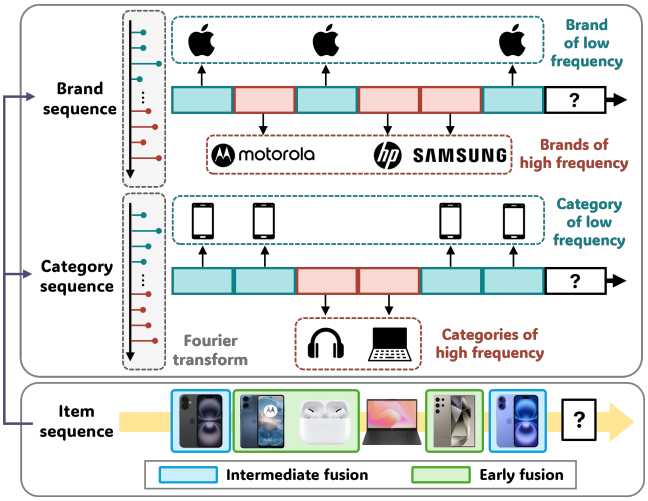


Figure 1: (i) Frequency signals and (ii) fusion types in side information integrated sequential recommendation. Frequency-based noise filtering removes the fourth item with inconsistent signals. Intermediate fusion (blue) highlights items aligned with key signals, while early fusion (green) captures broader combinations.

noise interference. Recent studies [5, 6, 21, 34] have attempted to address this issue by eliminating noise and emphasizing crucial information with embedding filtering techniques. Nevertheless, they are limited in considering a single sequence, focusing solely on item IDs. While DLFSRec [16] introduces a frequency-based learnable filter in the multi-sequence, it overlooks sequence-level denoising. To overcome this limitation, it is necessary to filter irrelevant signals across individual multiple sequences of item IDs and attributes.

(ii) Limited utilization of side-information. Although intermediate fusion addresses the issue in early and late fusion strategies, it primarily focuses on utilizing item attributes to guide the importance of item IDs. Specifically, NOVA [15], DIF-SR [29], and ASIF [27] exploit item attributes only for calculating attention weights. The final user representation is then obtained by aggregating the item ID vectors in the sequence. As a result, it fails to directly integrate item attributes into user representations, thereby missing strong collaborative signals across attributes.

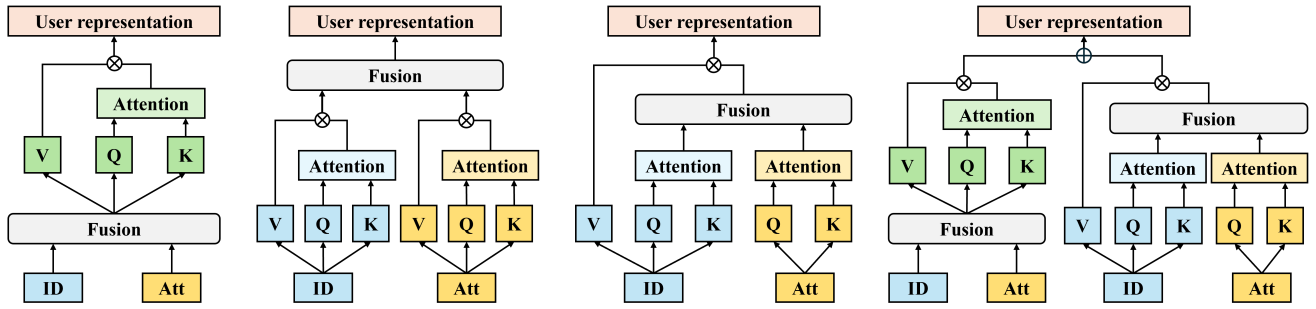
We first employ *frequency-based noise filtering* to remove noisy signals and extract salient patterns. Specifically, we transform each sequence into a frequency signal using discrete Fourier transforms. We then apply a frequency-based filtering algorithm, a common technique in digital signal processing [3, 19, 22]. It can consider periodicity and patterns that may be difficult to discern in the time domain [6, 21, 34]. Since essential information differs across item IDs and attributes, we apply frequency-based filtering to each sequence. For instance, as illustrated in Figure 1, the third item belongs to the "*Brand*" of "*Apple*", which represents a consistent pattern that should be emphasized. However, from the perspective of the category sequence, "*Earphone*" may appear as an inconsistent pattern within the category sequence. Therefore, we employ attribute-level filtering for each sequence to identify and prioritize meaningful signals across different attributes effectively.

We then introduce *dual multi-sequence fusion*, combining intermediate and early fusion. Intermediate fusion effectively aggregates *ID-centric* correlation within the sequence [14, 15, 27, 29]. As depicted in Figure 1, the brand and category sequences may highlight "*Apple*" and "*Cellular phone*", respectively. Intermediate fusion aggregates these highlighted attributes into an item ID value matrix, assigning higher attention scores to items that align with them, such as the first and last items, corresponding to "*Apple cellular phone*". This approach ensures that the most critical attribute combinations are emphasized, allowing the model to focus on items that best represent the user's core preferences. However, it primarily captures relationships within a single attribute and may overlook broader patterns across different attributes. We thus adopt early fusion, which is more effective for identifying correlations between various attributes. For example, if a user consistently prefers the "*Apple*" brand across different categories, early fusion can recognize this preference even when the item is not specifically highlighted in the category sequences, such as "*Apple earphone*". Similarly, if a user prefers the "*Cellular phone*" category regardless of brand, early fusion can effectively capture this pattern by identifying relevant items, such as "*Motorola cellular phone*" and "*Samsung cellular phone*". By representing items with a combination of attributes, early fusion provides a holistic view of user preferences that may not be fully captured through intermediate fusion alone. To mitigate information invasion of the naïve early fusion [15, 29], we also align ID and attribute representations in the same space. This allows the dual fusion approach to mitigate potential drawbacks while leveraging the strengths of both early and intermediate fusion.

To this end, we propose a novel side-information integrated sequential recommendation model, namely ***D**ual Side-**I**nformation **F**iltering and **F**usion model (**DIFF**)*. It consists of two key components: (i) *Frequency-based Noise Filtering* and (ii) *Dual Multi-sequence Fusion*. First, we remove noise and maintain only essential information based on the frequency domain. We then adjust high- and low-frequency signals for each item ID and attribute. Subsequently, filtered ID and attribute sequences are utilized in dual fusion. It consists of two distinct fusion blocks corresponding to intermediate and early fusion. As *ID-centric Fusion*, intermediate fusion captures the intra-attribute correlation across items. As *Attribute-enriched Fusion*, early fusion enables us to identify inter-attribute correlations across various attributes. With the proposed filtering and fusion strategy, DIFF is more robust in learning subtle and complex item relationships in multiple sequences. Experimental results show that DIFF significantly outperforms the state-of-the-art SISR models, improving performance by up to 14.1% and 12.5% on Recall@20 and NDCG@20 across four real-world datasets.

2 Related Work

Sequential Recommendation (SR). It aims to deliver the next item based on the user's sequential interaction history. Numerous studies have employed neural architectures as encoders, *e.g.*, Convolutional Neural Networks (CNNs) [25], Recurrent Neural Networks (RNNs) [10, 13], Graph Neural Networks (GNNs) [9, 28], and transformers [11, 17, 24]. Recently, some studies [5, 6, 21, 34] have shifted from the time domain to the frequency domain, identifying salient patterns in user behavior. However, they primarily focus on



(a) Early fusion (b) Late fusion (c) Intermediate fusion (d) Dual fusion (Ours)

Figure 2: Comparison of side information fusion methods. Existing methods are broadly categorized into (a) early, (b) late, and (c) intermediate fusion. We introduce (d) *dual fusion*, which benefits from early and intermediate fusion.

learning item correlations only with item ID sequences, neglecting side-information that provides a rich context for user behavior.

Side-Information Integrated SR (SISR). It utilizes both item IDs and attributes in the user's sequential history. S^3-Rec [33] adopts self-supervised auxiliary tasks to learn the relationship between item IDs and attributes. DLFSRec [16] utilizes a distribution-based learnable filter, representing ID and attributes by Gaussian distribution to capture their uncertainty. Then, various fusion methods [14, 15, 27, 29, 31] have been proposed to combine item IDs and attributes in the self-attention mechanism. As depicted in Figure 2, they can be categorized into three pillars as follows [1, 2].

- **Early fusion**: It incorporates item ID and attributes at the input level as illustrated in Figure 2(a). GRU4Rec_F and SASRec_F [33] create a unified representation by combining sequences of IDs and attributes as input before feeding it into the model by concatenation, summation, or gating. Although they combine the ID-attribute interactions via fused embeddings, it is challenging to learn the entangled embedding space of IDs and attributes as pointed out in the previous work, *i.e.*, information invasion [15].
- **Late fusion**: It delays the integration of item IDs and attributes until the model's final layer as in Figure 2(b). FDSA [31] adopts separated self-attention blocks to encode item IDs and attributes independently. While it can capture the different contexts of individual item sequences, it risks missing out on interactions between IDs and attributes.
- **Intermediate fusion**: It considers the interaction between item IDs and attributes in the intermediate layer as shown in Figure 2(c). It first extracts meaningful patterns from each sequence before combining them. Concretely, NOVA [15] DIF-SR [29], ASIF [27] integrate ID and attributes at an intermediate layer to calculate the query and key matrices in the self-attention block. However, they utilize attributes to obtain attention scores, overlooking direct correlations across attributes.

Under the categorization above, some methods adopt the combination of intermediate and late fusion. ESIF [23] aggregates intermediate fused attention utilizing each attribute value matrix, MSSR [14] introduces both intra-sequence and inter-sequence attention to consider the correlation between item ID and attribute sequences. However, existing studies do not explicitly combine the advantages of early and intermediate fusion.

3 Preliminaries

Problem Formulation. Let $\mathcal{I} = \{i_1, \ldots, i_n\}$ represent a set of n items. The user's item sequence is denoted as $s = [i_1, \ldots, i_{|s|}]$, where i_j is the j-th item in the sequential order, and $|s|$ is the total number of items interacted with by the user. Following the previous studies [14, 27, 31], we mainly consider item-related side-information, *e.g.*, brand and category. For side-information integrated sequential recommendation, each item $i \in \mathcal{I}$ is described by its unique item ID and multiple attributes. Specifically, it is represented as $i_j = \{v_j, a_{1,j}, \ldots, a_{m,j}\}$, where v_j is its item ID, $a_{k,j}$ is the k-th attribute type, and m is the total number of attributes. Our goal is to predict the next item the user is most likely to prefer, expressed as $\text{argmax}_{j \in \mathcal{I}} P(i_{|s|+1} = j \mid s)$.

Discrete Fourier Transform (DFT). The DFT is a fundamental component of digital signal processing, converting a sequence in the time domain into the frequency domain. Given a sequence with length N, the DFT is represented as $\mathcal{F} : \mathbb{R}^N \to \mathbb{C}^N$, and its inverse, *i.e.*, the inverse discrete Fourier transform (IDFT), is denoted as $\mathcal{F}^{-1} : \mathbb{C}^N \to \mathbb{R}^N$. The DFT can be performed by multiplying a sequence matrix $\mathbf{X} \in \mathbb{R}^{N \times d}$ by the matrix $\mathbf{F} \in \mathbb{C}^{N \times N}$.

$$\bar{\mathbf{X}} = \mathcal{F}(\mathbf{X}) = \mathbf{F}\mathbf{X} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & e^{\frac{-2\pi i}{N}} & \cdots & e^{\frac{-2\pi i(N-1)}{N}} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & e^{\frac{-2\pi i(N-1)}{N}} & \cdots & e^{\frac{-2\pi i(N-1)^2}{N}} \end{bmatrix} \mathbf{X}, \tag{1}$$

where i is the imaginary unit, and $\bar{\mathbf{X}} \in \mathbb{C}^{N \times d}$ is the frequency component of sequence $\mathbf{X}$. Interestingly, $\bar{\mathbf{X}}$ can be separated into two parts: *low-frequency* and *high-frequency* components. We define the first c rows as a low-frequency component $\bar{\mathbf{X}}_{LFC} \in \mathbb{C}^{c \times d}$ and the remaining rows as a high-frequency component $\bar{\mathbf{X}}_{HFC} \in \mathbb{C}^{(N-c) \times d}$. IDFT is then applied to convert each component into a different signal type.

$$\begin{aligned} \tilde{\mathbf{X}}_{LFC} &= \mathcal{F}^{-1}(\bar{\mathbf{X}}_{LFC}) = [\mathbf{f}_1^{*\top}, \ldots, \mathbf{f}_c^{*\top}]\bar{\mathbf{X}}_{LFC}, \\ \tilde{\mathbf{X}}_{HFC} &= \mathcal{F}^{-1}(\bar{\mathbf{X}}_{HFC}) = [\mathbf{f}_{c+1}^{*\top}, \ldots, \mathbf{f}_N^{*\top}]\bar{\mathbf{X}}_{HFC}, \end{aligned} \tag{2}$$

where $\mathbf{f}_i$ represents the i-th row vector in the matrix $\mathbf{F}$, and '$*$' denotes the conjugate operation. The low-frequency component $\tilde{\mathbf{X}}_{LFC} \in \mathbb{R}^{N \times d}$ captures the overall trend of the sequence, representing the signal that does not change frequently. In contrast, the

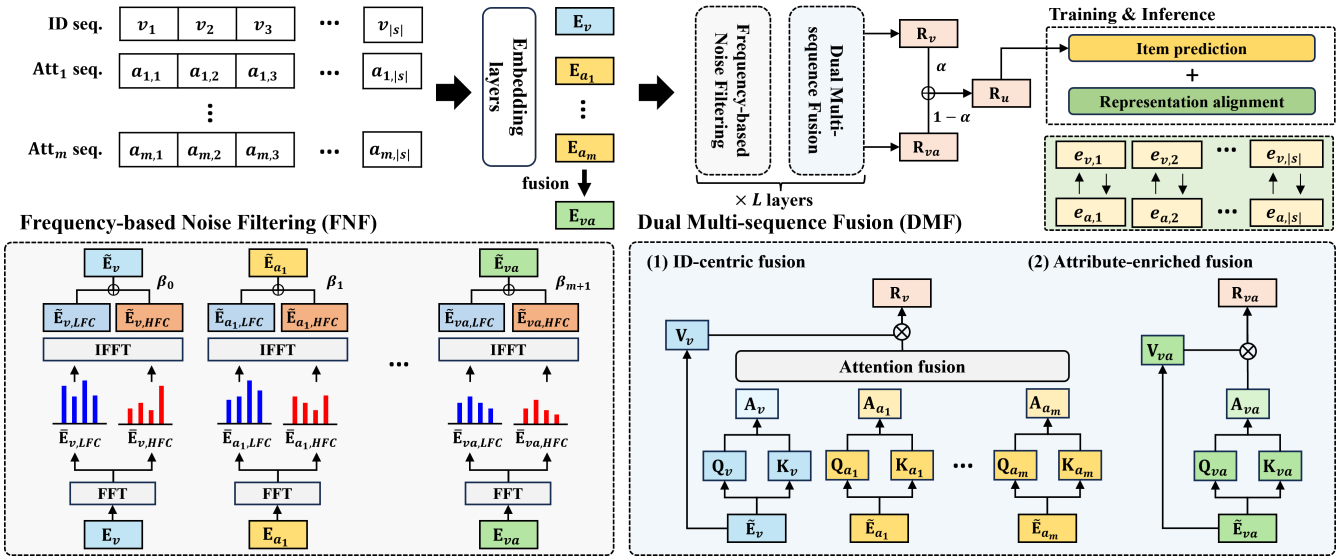


Figure 3: An overview of DIFF. DIFF processes both independent sequences and early fused sequences via L layers of two components: (i) Frequency-based Noise Filtering and (ii) Dual Multi-sequence Fusion. DIFF yields filtered user representations that fully integrates item attributes. Multi-task learning with representation alignment ensures smooth ID-attribute fusion.

high-frequency component $\tilde{\mathbf{X}}_{HFC} \in \mathbb{R}^{N \times d}$ represents the signal with abrupt variations. Note that we utilize Fast Fourier Transform (FFT) [4, 8], an efficient algorithm computing the DFT and IDFT.

4 Proposed Model: DIFF

In this section, we present the ***D**ual Side-**I**nformation **F**iltering and **F**usion (**DIFF**)* model, which effectively removes noisy signals and fully leverages the correlation across item IDs and attributes. Figure 3 depicts the overall architecture of DIFF, which consists of two main components: (i) *Frequency-based Noise Filtering* and (ii) *Dual Multi-sequence Fusion*. Specifically, frequency-based noise filtering is used to eliminate noise and extract essential signals (Section 4.1). Subsequently, dual multi-sequence fusion is employed to learn complex interactions across filtered item ID and attribute sequences (Section 4.2). We also adopt an alignment loss to prevent information invasion between item IDs and attributes (Section 4.3). Lastly, we explain the training and inference of DIFF (Section 4.4).

4.1 Frequency-based Noise Filtering

We employ *frequency-based noise filtering* to reduce irrelevant variations and distinguish essential patterns associated with consistent user preferences. The item sequence is converted to a frequency signal using the Fourier transform. Since item IDs and attributes exhibit different patterns, frequency-based filtering is applied independently to the item ID and attribute sequences.

Embedding Layer. Given a user sequence $s = [i_1, i_2, \ldots, i_{|s|}]$, we first obtain the embedding matrices for item ID sequence and attribute sequences.

$$\begin{aligned} \mathbf{E}_v &= \mathcal{E}_v(v_1, v_2, \ldots, v_{|s|}), \\ \mathbf{E}_{a_k} &= \mathcal{E}_{a_k}(a_{k,1}, a_{k,2}, \ldots, a_{k,|s|}) \ \forall k \in [1, m], \end{aligned} \tag{3}$$

where $\mathbf{E}_v$ and $\mathbf{E}_{a_k} \in \mathbb{R}^{|s| \times d}$ are the resulting embedding matrices for the item ID sequence and the k-th attribute sequence, respectively. Also, $\mathcal{E}_v$ and $\mathcal{E}_{a_k}$ are embedding layers for the item ID and k-th item attribute, respectively.

While existing studies [14, 29] have primarily focused on optimizing the intermediate fusion, our approach considers both early and intermediate fusion to capture essential patterns through integrated embeddings across item IDs and attributes. To achieve this, we obtain a fused embedding $\mathbf{E}_{va}$ for early fusion that combines the item ID and all attributes:

$$\mathbf{E}_{va} = \text{Fusion}\left(\mathbf{E}_v, \mathbf{E}_{a_1}, \ldots, \mathbf{E}_{a_m}\right), \tag{4}$$

where $\mathbf{E}_{va} \in \mathbb{R}^{|s| \times d}$ and $\text{Fusion}(\cdot)$ denotes the fusion function for item ID and attribute embeddings. Following the prior studies [14, 15, 27, 29], various fusion functions can be used, *i.e.*, summation, concatenation, or gating.

Frequency-based Filtering. We employ the filtering method to remove noise and spurious signals for each sequence. As pointed out in previous studies [16, 27], it is crucial to enhance the utilization of side-information by alleviating noisy interference. To achieve this, we utilize the discrete Fourier transform to project a sequence into the frequency domain. Specifically, we define the low- and high-frequency components of item ID embeddings as $\bar{\mathbf{E}}_{v,LFC} \in \mathbb{C}^{c \times d}$ and $\bar{\mathbf{E}}_{v,HFC} \in \mathbb{C}^{(|s|-c) \times d}$, respectively.

$$\begin{aligned} \tilde{\mathbf{E}}_{v,LFC} &= \mathcal{F}^{-1}(\bar{\mathbf{E}}_{v,LFC}), \\ \tilde{\mathbf{E}}_{v,HFC} &= \mathcal{F}^{-1}(\bar{\mathbf{E}}_{v,HFC}). \end{aligned} \tag{5}$$

$\tilde{\mathbf{E}}_{v,LFC}, \tilde{\mathbf{E}}_{v,HFC} \in \mathbb{R}^{|s| \times d}$ represent the low- and high-frequency components of the item ID embedding, respectively. Similarly, we obtain the low- and high-frequency components for each attribute

embedding and fused embedding through frequency-based filtering.

$$\begin{aligned}\tilde{\mathbf{E}}_{a_k,LFC} &= \mathcal{F}^{-1}(\bar{\mathbf{E}}_{a_k,LFC}),\ \ \forall k \in [1,m],\\ \tilde{\mathbf{E}}_{a_k,HFC} &= \mathcal{F}^{-1}(\bar{\mathbf{E}}_{a_k,HFC}),\ \ \forall k \in [1,m],\end{aligned} \tag{6}$$

$$\begin{aligned}\tilde{\mathbf{E}}_{va,LFC} &= \mathcal{F}^{-1}(\bar{\mathbf{E}}_{va,LFC}),\\ \tilde{\mathbf{E}}_{va,HFC} &= \mathcal{F}^{-1}(\bar{\mathbf{E}}_{va,HFC}),\end{aligned} \tag{7}$$

where $\tilde{\mathbf{E}}_{a_k,LFC}, \tilde{\mathbf{E}}_{a_k,HFC}, \tilde{\mathbf{E}}_{va,LFC}, \tilde{\mathbf{E}}_{va,HFC} \in \mathbb{R}^{|s|\times d}$.

From a frequency perspective, low-frequency signals represent stable patterns that change minimally over a sequence, while high-frequency signals exhibit rapid fluctuations. In the context of item sequences, the low-frequency component can be interpreted as representing long-term and consistent user interests. In contrast, the high-frequency component reflects short-term and volatile interests. While user's long-term consistent interests are crucial for making accurate recommendations, short-term interests that emerge suddenly are often less significant and may serve as noisy information.

To prioritize long-term stable user interests, we derive the filtered embeddings $\tilde{\mathbf{E}}_v$, $\tilde{\mathbf{E}}_{a_k}$, and $\tilde{\mathbf{E}}_{va}$ for each sequence by adjusting the impact of the high-frequency component.

$$\begin{aligned}\tilde{\mathbf{E}}_v &= \tilde{\mathbf{E}}_{v,LFC} + \beta_0 \tilde{\mathbf{E}}_{v,HFC},\\ \tilde{\mathbf{E}}_{a_k} &= \tilde{\mathbf{E}}_{a_k,LFC} + \beta_k \tilde{\mathbf{E}}_{a_k,HFC},\ \forall k \in [1,m],\\ \tilde{\mathbf{E}}_{va} &= \tilde{\mathbf{E}}_{va,LFC} + \beta_{m+1} \tilde{\mathbf{E}}_{va,HFC},\end{aligned} \tag{8}$$

where $\beta_0, \beta_1, ..., \beta_{m+1}$ are trainable scalar parameters used to adjust the high-frequency components of each input embedding. Empirically, we observe that β is trained to a very small value, *i.e.*, the impact of short-term fluctuating interests is reduced.

4.2 Dual Multi-sequence Fusion

We leverage both early and intermediate fusion to fully exploit the potential of side-information. Since two fusion strategies can capture different correlations across items and attributes, our dual fusion can be more effective than solely relying on intermediate fusion [15, 27, 29]. Specifically, early fusion effectively captures inter-attribute correlations, while intermediate fusion focuses on intra-attribute correlations within individual attributes.

ID-centric Fusion. We employ ID-centric fusion to better capture the correlation between item IDs. This approach, a form of intermediate fusion, is also utilized in existing studies [29]. We project ID and attribute embedding sequences onto different query and key matrices. The query and key matrices for the h-th attention head are as follows:

$$\begin{aligned}\mathbf{Q}_v^h = \tilde{\mathbf{E}}_v \mathbf{W}_{Q,v}^h, \mathbf{K}_v^h = \tilde{\mathbf{E}}_v \mathbf{W}_{K,v}^h,\\ \mathbf{Q}_{a_k}^h = \tilde{\mathbf{E}}_{a_k} \mathbf{W}_{Q,a_k}^h, \mathbf{K}_{a_k}^h = \tilde{\mathbf{E}}_{a_k} \mathbf{W}_{K,a_k}^h,\ \forall k \in [1,m],\end{aligned} \tag{9}$$

where $\mathbf{W}_{Q,v}^h, \mathbf{W}_{K,v}^h \in \mathbb{R}^{d\times d_h}$ are query and key projection matrices for item IDs, and $\mathbf{W}_{Q,a_k}^h, \mathbf{W}_{K,a_k}^h \in \mathbb{R}^{d\times d_h}$ are query and key projection matrices for the k-th item attribute. We then compute the attention score for each sequence via the dot-product of the query-key pairs.

$$\begin{aligned}\mathbf{A}_v^h &= \mathbf{Q}_v^h \left(\mathbf{K}_v^h\right)^\top,\\ \mathbf{A}_{a_k}^h &= \mathbf{Q}_{a_k}^h \left(\mathbf{K}_{a_k}^h\right)^\top,\ \forall k \in [1,m],\end{aligned} \tag{10}$$

where $\mathbf{A}_v^h \in \mathbb{R}^{|s|\times|s|}$ and $\mathbf{A}_{a_k}^h \in \mathbb{R}^{|s|\times|s|}$ denote attention score matrices of ID sequence and the k-th attribute sequence obtained from h-th attention head. Finally, we fuse the item correlations from item IDs and attributes, *i.e.*, attention score matrices, and aggregate them into an item ID value matrix.

$$\begin{aligned}\mathbf{R}_v &= \text{FFN}(\text{concat}(\mathbf{R}_v^1, \ldots, \mathbf{R}_v^H)\mathbf{W}_v),\\ \text{where } \mathbf{R}_v^h &= \text{softmax}\left(\frac{\text{Fusion}\left(\mathbf{A}_v^h, \cdots, \mathbf{A}_{a_m}^h\right)}{\sqrt{d_h}}\right)\mathbf{V}_v^h.\end{aligned} \tag{11}$$

Here, H is the number of attention heads and $\mathbf{V}_v^h = \tilde{\mathbf{E}}_v \mathbf{W}_{V,v}^h$ denotes the value matrix of item IDs at the h-th attention head. $\mathbf{W}_v \in \mathbb{R}^{d\times d}$ is a weight parameter matrix, concat$(\cdot)$ and FFN$(\cdot)$ indicate concatenation and feed-forward network, respectively.

Attribute-enriched Fusion. We adopt attribute-enriched fusion to reflect the inter-attribute correlations across various attributes, *i.e.*, early fusion. Specifically, we apply self-attention to the fused embeddings as follows.

$$\mathbf{Q}_{va}^h = \tilde{\mathbf{E}}_{va} \mathbf{W}_{Q,va}^h,\ \mathbf{K}_{va}^h = \tilde{\mathbf{E}}_{va} \mathbf{W}_{K,va}^h, \tag{12}$$

where $\mathbf{W}_{Q,va}^h, \mathbf{W}_{K,va}^h \in \mathbb{R}^{d\times d_h}$ are query, key projection matrices for the fused sequence, respectively. The attention score is computed using the query and key matrices of the fused sequence, thereby explicitly modeling strong correlations across item IDs and attributes.

$$\mathbf{A}_{va}^h = \mathbf{Q}_{va}^h \left(\mathbf{K}_{va}^h\right)^\top, \tag{13}$$

where $\mathbf{A}_{va}^h \in \mathbb{R}^{|s|\times|s|}$ is an attention score matrix of the fused embedding sequence for h-th attention head. We then derive the user representation from these fused embeddings.

$$\begin{aligned}\mathbf{R}_{va} &= \text{FFN}(\text{concat}(\mathbf{R}_{va}^1, \ldots, \mathbf{R}_{va}^H)\mathbf{W}_{va}),\\ \text{where } \mathbf{R}_{va}^h &= \text{softmax}\left(\frac{\mathbf{A}_{va}^h}{\sqrt{d_h}}\right)\mathbf{V}_{va}^h,\end{aligned} \tag{14}$$

where $\mathbf{V}_{va}^h \in \mathbb{R}^{d\times d_h}$ denotes a projected value matrix for the fused embeddings. $\mathbf{W}_{va} \in \mathbb{R}^{d\times d}$ is a weight parameter matrix.

User Representation. The final user representation is obtained by aggregating early and intermediate fusion results. While the ID-centric representation via intermediate fusion emphasizes fine-grained interactions at the individual item level, the attribute-enriched representation via early fusion explicitly captures strong attribute correlations. The final representation is computed as:

$$\mathbf{R}_u = \alpha \mathbf{R}_v + (1-\alpha)\mathbf{R}_{va}, \tag{15}$$

where $\mathbf{R}_u \in \mathbb{R}^{|s|\times d}$ and α denotes the representation aggregating hyperparameter. The last element in $\mathbf{R}_u$, *i.e.*, $\mathbf{r}_{u,|s|} \in \mathbb{R}^d$, is used as the user representation vector for prediction.

4.3 Representation Alignment

Item IDs and attributes are initially embedded in separate spaces. However, they need to be semantically consistent since both ID-centric and attribute-enriched representations are used for the final user representation. For that, we leverage a contrastive loss to align the embedding spaces of item IDs and fused attributes. Inspired by

Table 1: Data statistics after preprocessing. Avg. Length indicates the average number of interactions per user.

Dataset	Yelp	Beauty	Sports	Toys
# Users	30,449	22,363	35,598	19,412
# Items	20,068	12,101	18,357	11,924
# Interactions	317,182	198,502	296,337	167,597
Avg. Length	10.4	8.9	8.3	8.6
Sparsity	99.95%	99.93%	99.95%	99.93%

previous work [27], we align the similarity between the item ID and the fused attribute embedding vectors.

$$\hat{\mathbf{Y}}_{v,a} = \text{softmax}\left(\frac{\mathbf{E}_v \mathbf{E}_a^\top}{\tau}\right), \hat{\mathbf{Y}}_{a,v} = \text{softmax}\left(\frac{\mathbf{E}_a \mathbf{E}_v^\top}{\tau}\right), \quad \text{where } \mathbf{E}_a = \text{Fusion}(\mathbf{E}_{a_1}, \ldots, \mathbf{E}_{a_m}). \tag{16}$$

Here, $\mathbf{E}_a \in \mathbb{R}^{|s| \times d}$ is a fused embedding matrix of item attributes with the fusion function. For that, the summation function is used. In this process, we use normalized item ID and fused attribute embeddings for stable training. The learnable temperature τ is used as a scaling factor. The final alignment loss is defined as follows:

$$\mathcal{L}_{align} = -\frac{1}{2b} \sum_{i=1}^{b} \sum \left(\mathbf{Y}^i \odot \log \hat{\mathbf{Y}}^i_{v,a} + \mathbf{Y}^i \odot \log \hat{\mathbf{Y}}^i_{a,v} \right), \tag{17}$$

where $\odot$ denotes element-wise product, $\mathbf{Y}^i \in \{0,1\}^{|s| \times |s|}$ is the ground truth of the i-th sequence, and b is the number of sequences in the mini-batch. Each element of $\mathbf{Y}^i$ is defined as follows.

$$\mathbf{Y}^i_{j,k} = \begin{cases} 1 & \text{if } \mathbf{E}^j_a = \mathbf{E}^k_a \\ 0 & \text{otherwise} \end{cases} \text{ for } j,k \in \{1, \ldots, |s|\}, \tag{18}$$

where $\mathbf{E}^j_a$ and $\mathbf{E}^k_a$ are the fused attribute embedding vectors obtained from the j-th and k-th items in i-th sequence, respectively.

4.4 Training and Inference

For inference, we make predictions using the final user representation vector $\mathbf{r}_{u,|s|}$ and the item ID embedding matrix $\mathbf{E}$.

$$\hat{\mathbf{y}} = \text{softmax}(\mathbf{r}_{u,|s|} \mathbf{E}^\top), \tag{19}$$

where $\hat{\mathbf{y}} \in \mathbb{R}^n$. To calculate the recommendation loss, we employ the cross-entropy loss function.

$$\mathcal{L}_{rec} = -\frac{1}{b} \sum_{i=1}^{b} \mathbf{y}^{(i)} \log \hat{\mathbf{y}}^{(i)}, \tag{20}$$

where $\mathbf{y}^{(i)} \in \{0,1\}^n$ is the one-hot encoded ground truth vector of the i-th sequence in the mini-batch, with the element corresponding to the target item set to 1 and all others to 0.

Finally, we train our model by combining the recommendation loss and representation alignment loss.

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda \mathcal{L}_{align}, \tag{21}$$

where λ is the hyperparameter to control the loss $\mathcal{L}_{align}$.

5 Experimental Setup

Datasets. We conduct extensive experiments on four real-world datasets following [14, 29], *i.e.*, Yelp [2] and Amazon review dataset [18] [3]. **Yelp** is a well-known business recommendation dataset. The attributes of categories, cities, and positions are utilized as side-information. We select three widely used subcategories that are constructed from the Amazon review datasets: **Beauty**, **Sports**, and **Toys**. They consist of item metadata and reviews collected from 1996 to 2014, and we utilize the categories, brands, and positions as side-information. As in the previous works [14, 29], we use the 5-core setting, which removes users and items that occur less than five times. The detailed statistics for the pre-processed datasets are shown in Table 1.

Evaluation Protocols and Metrics. Following [14, 29], we adopt the *leave-one-out* strategy to split train, validation, and test sets. For each user sequence, we use the last item for testing, the second last item for validation, and the rest items for training. All models are evaluated in a *full ranking* scenario on all items rather than sampled items following [14, 29]. We opt not to penalize repeated items unlike [16, 21] that have previously appeared within the user history to maintain consistency across diverse datasets. Applying such penalties can negatively impact models on datasets like Yelp, where repeated interactions are common, leading to biased performance estimation. For evaluation metrics, we employ top-k Recall (R@k) and top-k Normalized Discounted Cumulative Gain (N@k) with $k = \{10, 20\}$.

Baselines. We thoroughly compare our proposed method with two categories: sequential recommendation (SR) and side-information integrated sequential recommendation (SISR) baselines. For SR baselines, **SASRec** [11] adopts the uni-directional self-attention method to capture the user interest. **DuoRec** [20] enhances SASRec [11] with contrastive learning. **FMLPRec** [34] proposes a filter-enhanced MLP to eliminate frequency domain noise. **BSARec** [21] leverages the Fourier transform to inject an inductive bias for modeling user patterns. For SISR baselines, **GRU4Rec$_\mathbf{F}$** and **SASRec$_\mathbf{F}$** are enhanced versions of GRU4Rec [10] and SASRec [11]. Following the previous work [14], the item ID and attributes are fused before feeding to the model via summation and concatenation for GRU4Rec$_\text{F}$ and SASRec$_\text{F}$, respectively. **S^3-Rec** [33] utilizes mutual information maximization to capture the correlations between items, sequences, and attributes. **FDSA** [31] adopts late fusion by utilizing multiple self-attention blocks. **NOVA** [15] adopts non-invasive self-attention mechanism for effective attention learning. **DIF-SR** [29] decouples the attention calculation of item ID and attributes. **DLFSRec** [16] proposes distribution-based learnable filters to effectively utilize side-information. **MSSR** [14] models the multiple user representations via a multi-sequence integrated attention layer. **ASIF** [27] utilizes side-information without noisy interference via fused attention with untied position information.

Implementation Details. We implement all models on the open-source recommendation framework Recbole [32] [4] or published code. All models are optimized using Adam optimizer [12], and tune the learning rate in $\{10^{-4}, 10^{-3}\}$. We set the maximum sequence

[2] https://www.yelp.com/dataset
[3] https://jmcauley.ucsd.edu/data/amazon/
[4] https://github.com/RUCAIBox/RecBole

Table 2: Overall performance comparison on four datasets. * denotes that DIFF shows statistically significant improvement ($p < 0.05$) over the best competitive model. The best results are marked in bold, and the second best results are underlined.

Dataset	Metric	SR baselines				SISR baselines									DIFF	Gain
		SASRec	DuoRec	FMLPRec	BSARec	GRU4Rec$_F$	SASRec$_F$	S^3-Rec	DLFSRec	FDSA	NOVA	DIF-SR	MSSR	ASIF		
Yelp	R@10	0.0607	0.0631	0.0711	0.0701	0.0414	0.0435	0.0598	0.0551	0.0537	0.0614	0.0686	0.0712	0.0724	**0.0815***	12.5%
	R@20	0.0875	0.0909	0.1029	0.1023	0.0679	0.0706	0.0869	0.0857	0.0856	0.0886	0.0998	0.1040	0.1052	**0.1200***	14.1%
	N@10	0.0383	0.0385	0.0424	0.0423	0.0213	0.0225	0.0377	0.0312	0.0284	0.0384	0.0415	0.0425	0.0427	**0.0470***	10.2%
	N@20	0.0451	0.0455	0.0506	0.0503	0.0280	0.0293	0.0445	0.0388	0.0364	0.0452	0.0493	0.0507	0.0510	**0.0567***	11.1%
Beauty	R@10	0.0842	0.0865	0.0855	0.0871	0.0682	0.0804	0.0839	0.0774	0.0811	0.0817	0.0891	0.0883	0.0920	**0.0935***	1.6%
	R@20	0.1191	0.1225	0.1239	0.1260	0.0991	0.1123	0.1186	0.1217	0.1152	0.1169	0.1281	0.1256	0.1322	**0.1347***	1.9%
	N@10	0.0424	0.0448	0.0426	0.0437	0.0380	0.0468	0.0420	0.0337	0.0461	0.0415	0.0444	0.0454	0.0463	**0.0526***	12.5%
	N@20	0.0511	0.0538	0.0522	0.0535	0.0458	0.0549	0.0508	0.0448	0.0547	0.0504	0.0542	0.0548	0.0564	**0.0632***	12.0%
Sports	R@10	0.0487	0.0489	0.0495	0.0506	0.0410	0.0443	0.0465	0.0402	0.0498	0.0473	0.0534	0.0549	0.0568	**0.0574**	1.1%
	R@20	0.0709	0.0723	0.0743	0.0741	0.0625	0.0648	0.0677	0.0656	0.0723	0.0690	0.0784	0.0809	0.0827	**0.0853***	3.2%
	N@10	0.0231	0.0246	0.0232	0.0239	0.0218	0.0251	0.0226	0.0183	0.0282	0.0229	0.0251	0.0261	0.0268	**0.0310***	10.1%
	N@20	0.0287	0.0305	0.0295	0.0298	0.0272	0.0302	0.0279	0.0246	0.0339	0.0283	0.0314	0.0326	0.0333	**0.0381***	12.5%
Toys	R@10	0.0889	0.0939	0.0923	0.0928	0.0643	0.0789	0.0913	0.0820	0.0884	0.0930	0.1011	0.1020	0.1007	**0.1023**	0.3%
	R@20	0.1225	0.1287	0.1302	0.1293	0.0950	0.1112	0.1238	0.1260	0.1221	0.1253	0.1379	0.1405	0.1393	**0.1425***	1.3%
	N@10	0.0436	0.0481	0.0446	0.0460	0.0350	0.0456	0.0449	0.0364	0.0506	0.0458	0.0504	0.0510	0.0496	**0.0553***	8.6%
	N@20	0.0521	0.0569	0.0541	0.0552	0.0427	0.0537	0.0531	0.0475	0.0591	0.0539	0.0597	0.0607	0.0593	**0.0656***	8.1%

length to 50, and we stop the training if the validation N@20 decreases for ten consecutive epochs. We tune all the hyperparameters on the validation data and report the performance on the test set using the models that show the highest performance on the validation set. For the proposed method, we set both the embedding size and batch size to 256, and both the number of layers and heads are set to 2. We set the frequency component split parameter c to 3 for Beauty, Sports, Yelp datasets and 5 for Toys dataset. We also tune the aggregating hyperparameters α among $\{0.1, 0.3, 0.5, 0.7, 0.9\}$ and loss balancing hyperparameter λ among $\{1, 5, 10, 20, 50, 100\}$. The fusion function Fusion$(\cdot)$ is set to gating for the Yelp dataset and concatenation for the Beauty, Sports, and Toys datasets. For the baseline models, we follow the original papers' settings for other hyperparameters of baselines, and we thoroughly tune them if not available. All results are averaged over five runs with different seeds, and we conducted the significance test using a paired t-test. Our code is available at https://github.com/HyeYoung1218/DIFF.

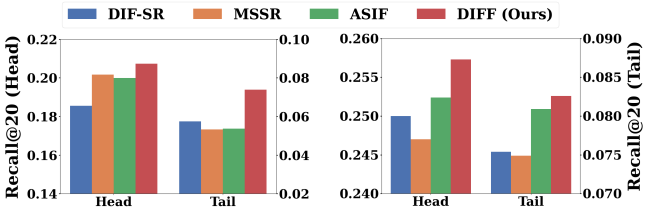
6 Experimental Results

6.1 Overall Performance

Table 2 reports the performance comparison between DIFF and other baselines in four real-world datasets. The key observations are as follows. (i) DIFF consistently achieves state-of-the-art performance on all datasets against the best competitive baseline, improving R@20 and N@20 by up to 14.1% and 12.5%, respectively. Especially, DIFF exhibits the best performance against the best competitive SISR baselines, *e.g.*, MSSR [14] and ASIF [27], yielding average gains of 4.7% and 11.2% on R@20 and N@20. This indicates that DIFF successfully avoids noisy patterns and leverages side-information. (ii) When compared to SR baselines that do not use side-information, SISR baselines, especially MSSR [14], ASIF [27], and DIFF, generally achieve superior performance. It implies that modeling user preferences with rich item context is critical for recommendation performance. (iii) Although DLFSRec [16] leverages frequency-based learnable filters for SISR, it does not primarily focus on fusion methods, which results in comparatively lower performance than late and intermediate fusion approaches (*i.e.*, FDSA [31], NOVA [15], DIF-SR [29], MSSR [14], and ASIF [27]). This highlights the importance of a well-designed fusion strategy in achieving superior performance. (iv) Among SISR baselines, intermediate fusion approaches (*i.e.*, NOVA [15], DIF-SR [29], MSSR [14], and ASIF [27]) generally demonstrate higher performance than early fusion methods (*i.e.*, GRU4Rec$_F$ and SASRec$_F$) and late fusion method (*i.e.*, FDSA [31]). Notably, two early fusion methods (GRU4Rec$_F$ and SASRec$_F$) lose performance of up to 23.6% and 35.0% performance at N@20 compared to GRU4Rec and SASRec, respectively. This underscores the importance of delicately designed fusion methods when utilizing side-information. (v) Among SISR baselines, ASIF [27] and DIFF demonstrate particularly promising performance compared to other SISR models. This highlights that, in addition to designing an effective fusion method, eliminating noisy correlations between IDs and attributes further enhances performance by ensuring meaningful interactions are captured.

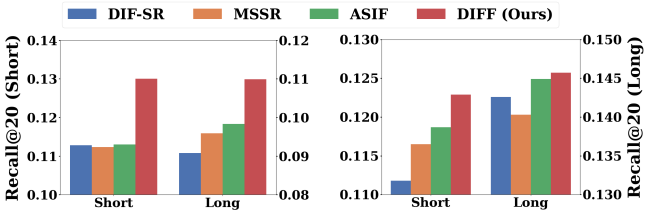
Table 3: Ablation study of DIFF. FNF refers to the Frequency-based Noise Filtering. IF and AF represent ID-centric Fusion and Attribute-enriched Fusion, respectively. Lastly, RA denotes Representation Alignment.

	Metric	w/o FNF	w/o IF	w/o AF	w/o RA	DIFF
Yelp	R@20	0.1045	0.1174	0.1185	0.1114	0.1200
	N@20	0.0512	0.0560	0.0564	0.0542	0.0567
Beauty	R@20	0.1289	0.1290	0.1334	0.1300	0.1347
	N@20	0.0615	0.0629	0.0576	0.0585	0.0632
Sports	R@20	0.0843	0.0795	0.0851	0.0827	0.0853
	N@20	0.0322	0.0375	0.0337	0.0330	0.0381
Toys	R@20	0.1373	0.1357	0.1459	0.1357	0.1420
	N@20	0.0615	0.0651	0.0598	0.0600	0.0657



(a) Yelp (b) Beauty

Figure 4: Performance comparison on different target item popularity groups. The target items of Head group are the top 10% most popular items, while the Tail group includes sequences with less popular target items.



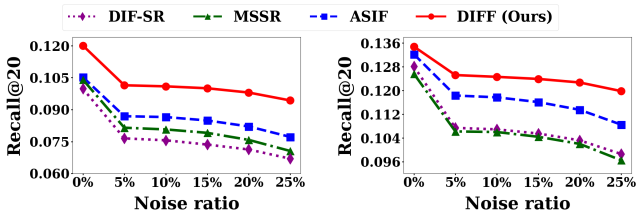
(a) Yelp (b) Beauty

Figure 5: Performance comparison on different sequence length groups. The Short group consists of sequences with a length of five (43% of Yelp and 51% of Beauty dataset), while the Long group includes sequences longer than five.

6.2 In-depth Analysis

Ablation Study. We validate the effectiveness of the key components of the proposed method through the ablation study as shown in Table 3. (i) Frequency-based Noise Filtering (FNF) significantly impacts performance across all datasets, delivering a performance gain of up to 14.8% and 10.7% in R@20 and N@20, respectively. It demonstrates that noisy signals are removed and only essential information is successfully extracted, leading to more accurate user representation. (ii) The proposed dual fusion strategy remarkably improves the accuracy compared to using only ID-centric Fusion (IF) or Attribute-enriched Fusion (AF) by more than 7.3% and 13.1% on R@20 and N@20. It shows that each fusion successfully captures complementary information to another. (iii) The representation alignment loss (RA) seamlessly integrates item ID and attribute information by aligning their embedding spaces, showing gains of up to 7.7% and 14.4% on R@20 and N@20. By harmonizing the representation spaces, ID and attribute information are effectively incorporated into the model.

Performance by Item Popularity. In Figure 4, we evaluate the performance of DIFF and baseline models by dividing the test user sequences into two groups: Head, consisting of sequences with target items from the top 10% most popular items, and Tail, with sequences containing less popular target items. The experimental results demonstrate strong performance of DIFF across both groups, effectively alleviating the popularity bias. By leveraging side-information filtering and fusion mechanisms, DIFF can extract meaningful signals from side-information, compensating for the sparse interactions typically associated with tail items. This suggests that DIFF also outperforms other competitive models for cold-start scenarios. In particular, for the Tail group, DIFF achieves up to 38.9% improvement on Yelp and 10.3% on the Beauty dataset.



(a) Yelp (b) Beauty

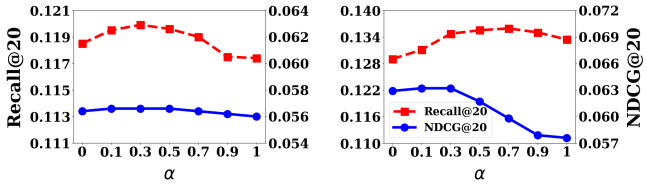
Figure 6: Robustness to noisy sequences on Yelp and Beauty datasets. It shows the performance of DIF-SR, MSSR, ASIF, and DIFF by varying the item substitution ratio.

Performance by Sequence Length. In Figure 5, we evaluate the performance of DIFF and baseline models by dividing user sequences into two groups based on their length. The Short group consists of users with five interacted items, while the Long group includes users with more than five interacted items. The results show that DIFF consistently outperforms the baseline models across both groups, effectively capturing user preferences regardless of sequence length. Notably, DIFF achieves significant improvements in the Short sequence group, where limited user interaction data. In particular, DIFF shows the performance gains in Recall@20 by up to 15.8% and 10% on the Yelp and Beauty dataset, respectively. This indicates that DIFF is particularly effective in scenarios with sparse historical interaction, showcasing its capability to leverage available information more efficiently.

6.3 Robustness to Noisy Sequence

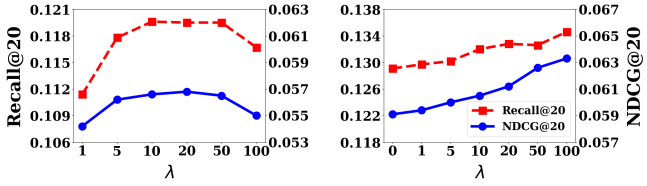
In Figure 6, we examine the robustness of the proposed method to demonstrate the effectiveness of Frequency-based Noise Filtering. Here, we adopt the most competitive SISR models, *i.e.*, DIF-SR [29], MSSR [14], and ASIF [27], for comparison. Following the approach in [5], we simulate noisy conditions by injecting synthetic noise into the test sequences. While they add random uniform noise to the original representations, we adopt a more challenging approach by replacing some items in each item ID sequence with random items, resulting in a more realistic and complex evaluation scenario. These substituted items can be regarded as fluctuating items that should ideally be ignored.

The key findings are as follows. (i) Even with a low noise ratio (*i.e.*, 5%), all models exhibit performance degradation across all datasets, highlighting the challenges posed by noisy inputs. However, DIFF demonstrates greater resilience than DIF-SR, MSSR, and ASIF. Notably, on the Beauty dataset, DIFF shows only a 7.1% performance drop, whereas ASIF, MSSR, and DIF-SR suffer significant performance drops of 16.2%, 15.4%, and 10.5%, respectively. (ii) As the noise ratio increases incrementally up to 25%, the performance gap between DIFF and other models consistently widens across all datasets. Notably, on the Yelp dataset, DIFF exhibits a relatively modest decline of 21.4%, whereas the baseline models show substantial drops, ranging from 26.6% to 32.9%. This suggests that baseline models struggle to capture user preferences under noisy conditions. In contrast, our approach effectively filters out noisy signals, ensuring the preservation of critical information.



(a) Yelp (b) Beauty

Figure 7: Performance with varying representation aggregating hyperparameter α. When $\alpha = 0$, only AF is utilized, and when $\alpha = 1$, only IF is employed.



(a) Yelp (b) Beauty

Figure 8: Performance with varying alignment loss balancing hyperparameter λ.

6.4 Hyperparameter Sensitivity

Representation Aggregating Hyperparameter. Figure 7 illustrates the sensitivity of the proposed method to representation aggregating hyperparameter α on Yelp and Beauty datasets. In the Yelp dataset, we observe that both Recall and NDCG peak at moderate α values around 0.5, with particularly small fluctuations in NDCG. However, the optimal value of α for the Recall@20 and NDCG@20 performance differs on the Beauty dataset. Higher α values improve recall performance, while lower α values lead to more significant NDCG gains, indicating complementary roles of two fusion types. Our dual fusion approach can effectively enhance performance by leveraging the distinct fusion characteristics.

Loss Balancing Hyperparameter. Figure 8 presents the impact of the loss balancing hyperparameter λ across four datasets. The results demonstrate that incorporating the alignment loss consistently improves performance across all datasets. Specifically, we observe performance gains of up to 4.6% and 6.9% in NDCG@20 on the Yelp and Beauty datasets, respectively. The Yelp dataset shows a steady increase and peak performance at $\lambda = 20$, after which performance decreases slightly. This indicates that an excessively high λ may result in over-aligning, which has less impact on performance improvement. For the Beauty dataset, increasing λ results in consistent improvements in Recall@20 and NDCG@20, suggesting that greater alignment contributes to better fusion of diverse features. These findings suggest that an optimal λ value is crucial for balancing alignment and performance, with different datasets exhibiting varying sensitivities to this hyperparameter.

6.5 Case Study

In Figure 9, we conducted a case study on the Yelp dataset to analyze the effectiveness of dual fusion strategies in capturing user preferences. We explore the distribution of attention weights from two fusion strategies, *i.e.*, ID-centric Fusion (IF) and Attribute-enriched Fusion (AF) to understand their individual contributions to the recommendation process. (i) IF allocates the highest attention weight

Figure 9: Case study of attention distribution in the dual fusion types of DIFF, *i.e.*, ID-centric fusion (Left) and Attribute-enriched fusion (Right), on the Yelp dataset.

to i_7 sharing a category of "*Coffee & Tea*" with the target item, demonstrating the ability to prioritize relevant attributes. However, IF alone fails to capture i_2, which shares a different but relevant category with the target item. (ii) AF allocates high attention weight to i_2 and i_7, which shares the "*Sandwiches*" category with the target item, covering more diverse items. However, AF alone does not emphasize i_7 as strongly as IF does, potentially overlooking highly relevant items. These observations indicate that the two fusion strategies capture different aspects of user preferences, with IF excelling at reinforcing specific attribute relevance and AF offering a more diverse coverage. Therefore, the complementary strengths of AF and IF suggest a synergistic potential in combining them into a dual fusion.

7 Conclusion

In this paper, we introduce the novel ***D**ual Side-**I**nformation **F**iltering and **F**usion model (**DIFF**)* model, which aims to effectively eliminate noisy interference and fully leverage side-information. For that, DIFF consists of a two-fold process: *Frequency-based Noise Filtering* and *Dual Multi-sequence Fusion*. It is essential to filter inconsistent patterns when incorporating various side-information, ensuring that only the most relevant signals contribute to learning user preferences. Additionally, we successfully combine intermediate and early fusion by leveraging ID-centric and attribute-enriched interactions. Our empirical evaluation reveals that DIFF achieves new state-of-the-art performance by up to 14.1% and 12.5% gains in Recall@20 and NDCG@20 across four benchmark datasets.

Acknowledgments

This work was partly supported by the Institute of Information & communications Technology Planning & evaluation (IITP) grant and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2019-II190421, RS-2021-II212068, RS-2022-II220680, RS-2025-00564083, and IITP-2025-RS-2024-00437633, each contributing 20% to this research).

References

[1] Pradeep K. Atrey, M. Anwar Hossain, Abdulmotaleb El-Saddik, and Mohan S. Kankanhalli. 2010. Multimodal fusion for multimedia analysis: a survey. *Multim. Syst.* 16, 6 (2010), 345–379.

[2] Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency. 2019. Multimodal Machine Learning: A Survey and Taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.* 41, 2 (2019), 423–443.

[3] Gregory A. Baxes. 1994. *Digital image processing - principles and applications.*

[4] James W. Cooley and John W. Tukey. 1965. An Algorithm for the Machine Calculation of Complex Fourier Series. *Mathematics of computation* 19, 90 (1965), 297–301.

[5] Xinyu Du, Huanhuan Yuan, Pengpeng Zhao, Junhua Fang, Guanfeng Liu, Yanchi Liu, Victor S. Sheng, and Xiaofang Zhou. 2023. Contrastive Enhanced Slide Filter Mixer for Sequential Recommendation. In *39th IEEE International Conference on Data Engineering, ICDE 2023, Anaheim, CA, USA, April 3-7, 2023*. 2673–2685.

[6] Xinyu Du, Huanhuan Yuan, Pengpeng Zhao, Jianfeng Qu, Fuzhen Zhuang, Guanfeng Liu, Yanchi Liu, and Victor S. Sheng. 2023. Frequency Enhanced Hybrid Attention Network for Sequential Recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*. 78–88.

[7] Hui Fang, Danning Zhang, Yiheng Shu, and Guibing Guo. 2020. Deep Learning for Sequential Recommendation: Algorithms, Influential Factors, and Evaluations. *ACM Trans. Inf. Syst.* 39, 1 (2020), 10:1–10:42.

[8] Matteo Frigo and Steven G. Johnson. 2005. The Design and Implementation of FFTW3. *Proc. IEEE* 93, 2 (2005), 216–231.

[9] Priyanka Gupta, Diksha Garg, Pankaj Malhotra, Lovekesh Vig, and Gautam M. Shroff. 2019. NISER: Normalized Item and Session Representations with Graph Neural Networks. *CoRR* (2019).

[10] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based Recommendations with Recurrent Neural Networks. In *ICLR*.

[11] Wang-Cheng Kang and Julian J. McAuley. 2018. Self-Attentive Sequential Recommendation. In *ICDM*. 197–206.

[12] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

[13] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. 2017. Neural Attentive Session-based Recommendation. In *CIKM*. 1419–1428.

[14] Xiaolin Lin, Jinwei Luo, Junwei Pan, Weike Pan, Zhong Ming, Xun Liu, Shudong Huang, and Jie Jiang. 2024. Multi-Sequence Attentive User Representation Learning for Side-information Integrated Sequential Recommendation. In *WSDM*. 414–423.

[15] Chang Liu, Xiaoguang Li, Guohao Cai, Zhenhua Dong, Hong Zhu, and Lifeng Shang. 2021. Non-invasive Self-attention for Side Information Fusion in Sequential Recommendation. In *AAAI*. 4249–4256.

[16] Haibo Liu, Zhixiang Deng, Liang Wang, Jinjia Peng, and Shi Feng. 2023. Distribution-based Learnable Filters with Side Information for Sequential Recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*.

[17] Chen Ma, Peng Kang, and Xue Liu. 2019. Hierarchical Gating Networks for Sequential Recommendation. In *KDD*. 825–833.

[18] Julian J. McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. 2015. Image-Based Recommendations on Styles and Substitutes. In *SIGIR*. 43–52.

[19] Judicaël Menant, Jean-François Nezan, Luce Morin, and Muriel Pressigout. 2017. A comparison of stereo matching algorithms on multi-core Digital Signal Processor platform. In *3D Image Processing, Measurement (3DIPM), and Applications 2017, Burlingame, CA, USA, January 29 - February 2, 2017*. 49–54.

[20] Ruihong Qiu, Zi Huang, Hongzhi Yin, and Zijian Wang. 2022. Contrastive Learning for Representation Degeneration Problem in Sequential Recommendation. In *WSDM*. 813–823.

[21] Yehjin Shin, Jeongwhan Choi, Hyowon Wi, and Noseong Park. 2024. An Attentive Inductive Bias for Sequential Recommendation beyond the Self-Attention. In *AAAI*. 8984–8992.

[22] Samir S Soliman and Mandyam D Srinath. 1990. Continuous and discrete signals and systems. *Englewood Cliffs* (1990).

[23] Zheng-Ang Su, Juan Zhang, Zhijun Fang, and Yongbin Gao. 2024. Enhanced side information fusion framework for sequential recommendation. *International Journal of Machine Learning and Cybernetics* (2024).

[24] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In *CIKM*. 1441–1450.

[25] Jiaxi Tang and Ke Wang. 2018. Personalized Top-N Sequential Recommendation via Convolutional Sequence Embedding. In *WSDM*. 565–573.

[26] Shoujin Wang, Liang Hu, Yan Wang, Longbing Cao, Quan Z. Sheng, and Mehmet A. Orgun. 2019. Sequential Recommender Systems: Challenges, Progress and Prospects. In *IJCAI*. 6332–6338.

[27] Shuhan Wang, Bin Shen, Xu Min, Yong He, Xiaolu Zhang, Liang Zhang, Jun Zhou, and Linjian Mo. 2024. Aligned Side Information Fusion Method for Sequential Recommendation. In *WWW Companion*. 112–120.

[28] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-Based Recommendation with Graph Neural Networks. In *AAAI*. 346–353.

[29] Yueqi Xie, Peilin Zhou, and Sunghun Kim. 2022. Decoupled Side Information Fusion for Sequential Recommendation. In *SIGIR*. 1611–1621.

[30] Xu Yuan, Dongsheng Duan, Lingling Tong, Lei Shi, and Cheng Zhang. 2021. ICAI-SR: Item Categorical Attribute Integrated Sequential Recommendation. In *SIGIR*. 1687–1691.

[31] Tingting Zhang, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Deqing Wang, Guanfeng Liu, and Xiaofang Zhou. 2019. Feature-level Deeper Self-Attention Network for Sequential Recommendation. In *IJCAI*. 4320–4326.

[32] Wayne Xin Zhao, Shanlei Mu, Yupeng Hou, Zihan Lin, Yushuo Chen, Xingyu Pan, Kaiyuan Li, Yujie Lu, Hui Wang, Changxin Tian, Yingqian Min, Zhichao Feng, Xinyan Fan, Xu Chen, Pengfei Wang, Wendi Ji, Yaliang Li, Xiaoling Wang, and Ji-Rong Wen. 2021. RecBole: Towards a Unified, Comprehensive and Efficient Framework for Recommendation Algorithms. In *CIKM*. 4653–4664.

[33] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-Rec: Self-Supervised Learning for Sequential Recommendation with Mutual Information Maximization. In *CIKM*. 1893–1902.

[34] Kun Zhou, Hui Yu, Wayne Xin Zhao, and Ji-Rong Wen. 2022. Filter-enhanced MLP is All You Need for Sequential Recommendation. In *WWW*. 2388–2399.