A Related Work

Our content recommendation task follows the paradigm of sequential recommendation (SRec) (Kang & McAuley, 2018; Wang et al., 2019). Accordingly, our work is closely aligned with the research on sequential recommendation and diffusion model (DM)-based sequential recommendation. In this section, we review the studies on these two topics in detail.

A.1 Sequential Recommendation

Sequential recommendation (SRec) has been widely studied in RSs, owing to the natural temporal order of users' behaviors (Wang et al., 2019; 2021). SRec can be technically divided into two categories: traditional sequential models and deep learning-based models. Traditional sequential models generally leverage sequential pattern mining (Yap et al., 2012) or Markov chain models (He & McAuley, 2016) to model the item dependencies in users' interaction sequences. Traditional sequential methods can only capture simple interaction patterns or short-term dependencies, thereby cannot achieve satisfactory recommendation performance. To overcome these limitations, deep learning-based sequential recommendation methods are proposed to model complex and long-term dependencies in users' behaviors. Among this category, one research line focuses on designing the effective sequence encoders and backbone networks to encode users' interaction sequence, including GRU (Hidasi et al., 2015), CNNs (Tang & Wang, 2018), Transformer (Kang & McAuley, 2018), and Mamba (Liu et al., 2025b). Building upon these, another research line further introduces advanced models, such as Graph Neural Networks (GNNs) (Chang et al., 2021) and generative models (Deldjoo et al., 2024). Among them, generative models have recently attracted significant attention. In particular, DMs Liu et al. (2025a) and large language models (LLMs) (Sheng et al., 2025) have emerged as the two most prominent approaches. DM-based methods will be discussed in detail in Section A.2. LLM-based methods focus on leveraging the open-world knowledge encoded in LLMs to enhance sequential recommendation performance (Harte et al., 2023).

A.2 Diffusion Models for Sequential Recommendation

In recent years, owing to the strong capability to model complex distributions of user behaviors and item content, diffusion models (DMs) have been widely applied in recommendation scenarios (Wei & Fang, 2025; Lin et al., 2024), including top-K recommendation (Wang et al., 2023b; Zhao et al., 2024) and multimodal recommendation (Ma et al., 2024c; Li et al., 2025a). In SRec, DM-based recommendation methods can be broadly categorized into two types: next item generation-based methods, and data augmentation-based methods. The former generally employ sequence encoders (e.g., GRU and Transformer) to encode users' context items into condition embeddings, which then guide the generation of next items (Yang et al., 2023b; Liu et al., 2025a; Li et al., 2025b; Cai et al., 2025; Hu et al., 2024; Li et al., 2025b; Ma et al., 2024b; Wang et al., 2024b; Xie et al., 2024). For example, (Yang et al., 2023b; Liu et al., 2025a) utilize Transformer to learn condition embeddings from users' historical interactions, which are then utilized to guide the next-item generation process. The latter category leverages DMs to generate additional interaction data in order to enrich users' interaction sequences and alleviate sequence sparsity. For instance, (Liu et al., 2023; Ma et al., 2024a; Wu et al., 2023) propose generating pseudo interaction sequences with DMs to mitigate the sequence sparsity problem. Additionally, several methods integrate contrastive learning with diffusion models to generate augmented views, thereby enhancing the training of DM-based recommendation methods (Cui et al., 2024b;a; Qu & Nobuhara, 2025).

Although these methods have achieved remarkable success, they pose a significant risk of generating uncredible content recommendations (e.g., fake news (Wang et al., 2022; 2024a; Ma et al., 2025), misinformation (Pathak et al., 2023; Fernandez et al., 2024)), which can severely harm both user experience and societal well-being. While (Ma et al., 2025) attempts to leverage DMs to mitigate fake news, its effectiveness is limited under the challenge of scarce labeled data. This limitation motivates us to steer DMs towards credible content recommendation while simultaneously addressing the challenge of learning from only limited annotated data.

B More Experimental Details

B.1 Datasets

In our task setting, we require users' chronological interaction sequences with content items, together with labels indicating whether each item contains uncredible content. However, only a limited number of public datasets fulfill these requirements. In this paper, we utilize three datasets: PolitiFact, GossipCop, and MHMisinfo.

The PolitiFact and GossipCop datasets are derived from the FakeNewsNet repository[1] , which collects data from two well-known fact-checking websites: PolitiFact and GossipCop. These datasets provide user–news interaction sequences along with labels that indicate whether each news article is fake or true. The MHMisinfo dataset is collected from a video-based mental health misinformation dataset[2] , containing users' interaction sequences with videos annotated by whether the videos contain mental health misinformation. Although this dataset records user–video interactions, the original video and image contents are not provided. Therefore, we represent the items using their video descriptions instead of visual features.

Given the high sparsity of these datasets, we adopt a data augmentation strategy following common practice (Yang et al., 2023b;a). Specifically, for each user, we transform their interaction sequence into multiple sub-sequences by treating each item as the target item and the items preceding it as historical context. This transformation increases the number of user–item interaction sequences and enriches the training data. The statistics of these datasets are reported in Table 1. After augmentation, the datasets have more sequences, thereby the recommendation performances of Rec4Mit and HDInt are different from the results reported in (Wang et al., 2022) and (Wang et al., 2024a).

Table 1: The statistics of the three used datasets after preprocessing.

Datasets	PolitiFact	GossipCop	MHMisinfo
# Content items	616	9,529	3,160
# Credible content items	306	6,792	2,815
# Uncredible content items	310	2,737	345
# Training sequences	103,335	510,149	38,083
# Test sequences	21,490	68,002	8,060

B.2 Baseline Descriptions

In this section, we introduce the baseline methods used in our comparison.

Traditional sequential recommendation methods:

- **GRU4Rec** (Hidasi et al., 2015) utilizes the Gated Recurrent Unit (GRU) to model the temporal dependencies of items in users' interaction sequences.
- **SASRec** (Kang & McAuley, 2018) employs the Transformer architecture to model the item dependencies in users' interaction sequences. This is one of the most representative sequential recommendation methods.
- **Bert4Rec** (Sun et al., 2019) replaces SASRec's unidirectional Transformer with a bidirectional Transformer architecture to model complex item dependencies. It also introduces a cloze task paradigm for sequential recommendation.
- **LRU4Rec** (Yue et al., 2024) designs linear recurrent units for sequential recommendation. It decomposes linear recurrence operations and proposes recursive parallelization, reducing model size and enabling efficient parallel training.

Contrastive learning-based sequential recommendation methods:

- **CL4SRec** (Xie et al., 2022) uses contrastive learning to address the data sparsity problem in sequential recommendation. It designs three sequence augmentation operations for contrastive

[1]https://github.com/KaiDMML/FakeNewsNet
[2]https://zenodo.org/records/13191247

learning: item cropping, item masking, and item reordering. Transformer is used as the sequential encoder of CL4SRec.

- **ContraRec** (Wang et al., 2023a) proposes two types of contrastive perspectives to enhance the performance of contrastive learning-based sequential recommendation: context-target contrast and context-context contrast. Transformer is used as the sequential encoder of ContraRec.

Sequential recommendation methods for mitigating uncredible content:

- **Rec4Mit** (Wang et al., 2022) first utilizes a disentangler to extract event- and veracity-aware information, respectively. Thereafter, the event embeddings are utilized to derive users' genuine preferences and predict the next items users may be interested in.
- **HDInt** (Wang et al., 2024a). Similar to Rec4Mit, HDInt is also dedicated to mitigating fake news in recommender systems. HDInt also considers the political bias. We omit this part, since it requires additional data and the political bias is not considered in our task.
- **PRISM** (Ma et al., 2025) proposes a protection-enhanced news recommendation method based on interest-aware sequential modeling. It utilizes DMs' controllable ability to learn user interest and mitigate fake news. However, it assumes all the labels of fake news are fully available, which does not hold in the real world. It is also a DM-based sequential recommendation method.

DM-based recommendation methods:

- **DreamRec** (Yang et al., 2023b) assumes that each user has an "oracle" item in mind and selects items that match his ideal item. It uses a Transformer to learn users' preferences, which then serve as the condition for generating the oracle item for each user.
- **DiffuRec** (Li et al., 2023) employs a diffusion model to represent item embeddings in a distribution space and then feeds the embeddings into an approximator to generate target item representations. It argues that the standard objective function of DMs is unsuitable for recommendation tasks and uses cross-entropy loss to optimize model parameters.
- **PreferDiff** (Liu et al., 2025a) proposes a surrogate optimization objective which extend BPR recommendation loss (Rendle et al., 2009) to variational format. Meanwhile, this surrogate optimization objective can also be extended to multiple negative items.

B.3 Evaluation Metrics

HR@K and NDCG@K are two commonly used metrics to evaluate the recommendation accuracy, thereby we do not make further introduction for them. Credible Rate (CR@K) is a metric to measure the credibility of a recommendation model. Specifically, it calculates the average rate of the credible content items in the recommendation lists:

$$\mathrm{CR@K} = \frac{1}{|\mathcal{S}_{test}|} \sum_{s \in \mathcal{S}_{test}} \frac{K - |\mathcal{R}_s \cap \mathcal{I}_{unc}^{Ground-truth}|}{K}, \tag{1}$$

where $\mathcal{S}_{test}$ is the test set of sequences. $\mathcal{R}_s$ is the recommendation list for sequence s. $\mathcal{I}_{unc}^{Ground-truth}$ denotes the ground-truth set of uncredible items. $|\mathcal{R}_s \cap \mathcal{I}_{neg}^{Ground-truth}|$ calculates the number of uncredible items in the recommendation list. The higher value of CR@K means the better performance in delivering credible recommendations.

In addition, we test how our methods perform in terms of both accurate and credible recommendations, we design a combined metric HC@K (i.e., combining HR@K and CR@K). Formally, HC@K is calculated as follows:

$$\mathrm{HC@K} = \frac{2 \times \mathrm{HR@K} \times (\mathrm{CR@K}/2)}{\mathrm{HR@K} + (\mathrm{CR@K}/2)}. \tag{2}$$

This combined metric is inspired by the F1-score, which combines precision rate and recall rate. To note that, since the values of HR@K and CR@K are not on the same scale, we divide CR@K with a factor of 2 to rescale it into a similar value level with HR@K. This adjustment ensures a fair combination; otherwise, the metric with a much smaller magnitude would disproportionately dominate the combined score.

B.4 Implementation Details

In this paper, we consider a more challenging and realistic scenario in which only a small proportion of uncredible items are verified. To simulate this setting, we randomly select 20% of the uncredible items with available labels during the training process, while the labels of the remaining items are treated as unknown. It is similar to the semi-supervised setting. In contrast, during the testing stage, all content labels are provided to enable an accurate evaluation.

The items in PolitiFact and GossipCop are news articles, and we use their textual descriptions as item content. In MHMisinfo, although the items are videos, only textual descriptions are available; thus, we can only rely on the textual descriptions for content representation. We encode these textual descriptions into language embeddings using LLaMA2-7B (Touvron et al., 2023), and further project them into a lower dimension through an MLP. Following (Liu et al., 2025a), we fix the transformed embedding dimension at 3072 for all DM-based methods, as they exhibit strong performance only with higher embedding sizes. For other methods, the embedding size is set to 64. We also experimented with larger embedding sizes for these methods, but observed little or no performance gain, and even performance drops for some methods, consistent with the findings in (Liu et al., 2025a).

In our implementation, we select Transformer as our sequence encoder. Following the standard configuration (Vaswani et al., 2017), the Transformer architecture in our implementation includes multi-head attention, position-wise feed-forward network, layer normalization, and dropout.

For our method `Disco`, the hyperparameter w is tuned within $\{0.5, 1, 1.5, 2, 5\}$. We fix m at 10,000 and tune γ within $\{0.1, 0.2, 0.3, 0.4, 0.5\}$ to control the maximum selection ratio as well as the growth rate of the current selection ratio. The maximum number of diffusion steps is fixed at 2,000 and the DDIM step is set to 100, following the settings of (Liu et al., 2025a). For all DM-based methods, we utilize a linear schedule for β_t in range [0.0001, 0.02]. In our implementation, we do not use a classifier-free guidance (Ho & Salimans), since we found it does not influence much to the performance of `Disco`. In our implementation, we found that the singular values in Λ are relatively large; therefore, the threshold for constructing the null space of uncredible features is fixed at 3 for all datasets in our experiments. We search learning rate in range $\{$1e-5, 5e-5, 1e-4, 5e-4, 1e-3$\}$. The batch size is searched in $\{2048 \times 2^i\}_{i=0,1,2,3}$. The model parameters are initialized using normal initialization and optimized by AdamW (Loshchilov & Hutter, 2017). The hyperparameter settings of baseline methods are reported in Table 2. All experiments are conducted on an NVIDIA A40 GPU with 48 GB of memory. Each method is run five times, and we report the average performance along with the standard deviation.

Table 2: The hyperparameter settings of baseline methods.

Methods	Hyperparameter searching space
GRU4Rec	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0
SASRec	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0
Bert4Rec	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0, mask probability$\sim\{0.2, 0.4, 0.6, 0.8\}$
LRURec	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0, dropout rate$\sim\{0.2, 0.4, 0.6, 0.8\}$
CL4SRec	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0, mask/reorder/crop proportion$\sim\{0.2, 0.4, 0.6, 0.8\}$, $\lambda\sim\{0.1, 0.3, ..., 0.9\}$
ContraRec	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0, mask/reorder/crop proportion$\sim\{0.2, 0.4, 0.6, 0.8\}$, $\tau_1, \tau_2 \sim\{0.1, 0.2, ..., 1\}$, $\gamma\sim\{0, 0.01, 0.1, 1, 5, 10\}$
Rec4Mit	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0, $k\sim\{2, 4, ..., 20\}$
HDInt	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0, $\lambda\sim\{1, 2, ..., 10\}$, $\gamma\sim\{2, 4, 6, 8, 10\}$
PRISM	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0, $T\sim\{500, 1000, 1500, 2000\}$, $w\sim\{0, 2, 4, 6, 8\}$, $\lambda_{OT}, \lambda_c, \lambda_r, \lambda_{rec}\sim\{0.2, 0.4, 0.6, 0.8, 1\}$, embedding size$\sim\{64, 128, 256, 512, 1024, 2048, 3072\}$
DreamRec	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0, $T\sim\{500, 1000, 1500, 2000\}$, $w\sim\{0, 2, 4, 6, 8\}$, embedding size$\sim\{64, 128, 256, 512, 1024, 2048, 3072\}$
DiffuRec	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0, $T\sim\{16, 32, 64, 128\}$, δ=0.001, embedding size$\sim\{64, 128, 256, 512, 1024, 2048, 3072\}$
PreferDiff	lr$\sim\{$1e-2, 5e-2, 1e-3, 5e-3, 1e-4$\}$, weight decay=0, $T\sim\{500, 1000, 1500, 2000\}$, $w\sim\{0, 2, 4, 6, 8\}$, $\lambda\sim\{0.2, 0.4, 0.6, 0.8\}$, embedding size$\sim\{64, 128, 256, 512, 1024, 2048, 3072\}$

B.5 More Hyperparameter Experiments

The hyperparameter γ controls the selection ratio of potential uncredible items. We evaluate the performance of `Disco` (using combined metric HC@5) under different values of γ in range $\{$0.1, 0.2, 0.3, 0.4, 0.5$\}$. As shown in Figure 1, `Disco` achieves the best performance when fixing $w = 0.1$ on PolitiFact and GossipCop, and $w = 0.4$ on MHMisinfo. Lower values prevent the model

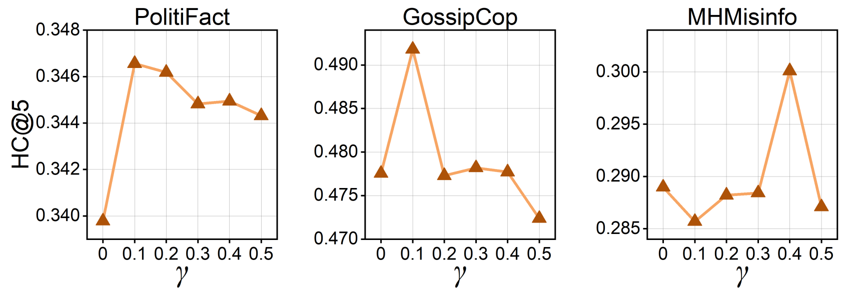

Figure 1: Effect of γ on `Disco`.

from effectively capturing potentially uncredible items, while higher values may introduce excessive noise, both of which degrade model performance.

B.6 Time Efficiency Analysis

We conduct experiments to evaluate the training and inference cost (in seconds) of our model `Disco` and four DM-based methods under the same batch size. As shown in Table 3, the training cost of `Disco` is relatively higher than DreamRec and PreferDiff, mainly due to our additional designs for credible content recommendation, including content disentanglement and credible subspace projection. This is acceptable due to the higher recommendation accuracy and credibility of our proposed method. Our training cost is much lower than that of DiffuRec an PRISM. As for inference cost, our proposed method `Disco` demonstrates the highest efficiency. This is because we adopt DDIM (Song et al., 2021) as our generation strategy, which is more efficient than the DDPM (Ho et al., 2020) paradigm employed by DreamRec, DiffuRec and PRISM. Even compared with PreferDiff, which also adopts DDIM, `Disco` also exhibits higher efficiency. It is because we do not employ classifier-free guidance in our implementation, since it has limited influence on our model while incurring additional time consumption. Although `Disco` requires disentangling item embeddings first in the inference stage, it's inference cost remains comparable to PreferDiff on GossipCop dataset, which contain large number of items.

Table 3: Time cost (s) of different models on PolitiFact, GossipCop, and MHMisinfo.

Datasets	Time cost (s)	DreamRec	DiffuRec	PRISM	PreferDiff	`Disco`
PolitiFact	Training/epoch	9.4	74.8	25.3	11.3	12.6
	Inference	278.2	224.3	994.3	15.2	10.5
GossipCop	Training/epoch	43.3	376.9	137.6	58.8	73.1
	Inference	956.8	783.1	3223.6	127.9	133.5
MHMisinfo	Training/epoch	3.3	26.9	8.9	3.9	4.2
	Inference	107.6	87.1	372.9	8.7	7.7

C Why DMs pose a danger of generating uncredible content recommendation?

In this section, we empirically and theoretically analyze why existing DM-based recommendation methods risk generating uncredible recommendations.

C.1 Empirical Findings

In DM-based recommendation methods, the condition and diffusion target are two critical factors. In this section, we conduct experiments to examine how they influence the recommendation credibility of DM-based methods. Specifically, we divide the training dataset into four subsets based on whether the context items or the diffusion target (i.e., target items) contain uncredible content. We use ✓ to denote that context items or target items contain uncredible content, and ✗ to denote

Table 4: Performance comparison of DreamRec and PreferDiff under different settings of uncredible content items in condition and diffusion target on PolitiFact and GossipCop datasets. Best results are highlighted in bold.

Methods	Whether contain uncredible content items? condition	diffusion target	Politi HR@10	NDCG@10	CR@10	HC@10	Gossip HR@10	NDCG@10	CR@10	HC@10
DreamRec	Training with complete dataset		**0.3287**	**0.2047**	0.8437	**0.3661**	**0.5501**	**0.3704**	0.8336	<u>0.4742</u>
	✗	✗	<u>0.2674</u>	<u>0.1571</u>	**0.9935**	<u>0.3477</u>	<u>0.4658</u>	<u>0.3160</u>	**0.9771**	**0.4769**
	✗	✓	0.0577	0.0409	0.1888	0.0716	0.0372	0.0284	0.0522	0.0307
	✓	✗	0.2671	0.1541	<u>0.9875</u>	0.3467	0.1927	0.1368	<u>0.9340</u>	0.2728
	✓	✓	0.0684	0.0413	0.0806	0.0507	0.0539	0.0404	0.0450	0.0317
PreferDiff	Training with complete dataset		**0.3554**	**0.2147**	<u>0.8981</u>	**0.3968**	**0.6022**	**0.3999**	0.8228	**0.4887**
	✗	✗	<u>0.3035</u>	<u>0.1915</u>	**0.9591**	<u>0.3717</u>	<u>0.5036</u>	<u>0.3657</u>	**0.9315**	<u>0.4839</u>
	✗	✓	0.0557	0.0410	0.1073	0.0547	0.0407	0.0304	0.0833	0.0412
	✓	✗	0.2625	0.1553	0.8561	0.3254	0.2074	0.1454	<u>0.9076</u>	0.2847
	✓	✓	0.0568	0.0385	0.0837	0.0482	0.0573	0.0421	0.0254	0.0208

the opposite. After this dataset division, we train two representative DM-based recommendation methods (DreamRec (Yang et al., 2023b) and PreferDiff (Liu et al., 2025a)) on each subset. From the results reported in Table 4, we can find that these two factors indeed affect the recommendation credibility of DM-based methods. We refer to these two factors as uncredible condition and uncredible diffusion target.

- **Uncredible condition.** When controlling the diffusion target, if the context items contain uncredible content that leads to an uncredible condition, the credibility metric CR@10 (i.e., credible Rate) decreases to some extent for both DreamRec and PreferDiff across the PolitiFact and GossipCop datasets. This finding indicates that the uncredible condition is a factor contributing to the risk of DMs generating uncredible recommendation results.

- **Uncredible diffusion target.** When controlling the condition, if the diffusion target is an uncredible item (i.e., an uncredible diffusion target), CR@10 drops to an extremely low level. This further emphasizes that the uncredible diffusion target is another key contributing factor.

Apart from these two findings, we also observe that training with the complete datasets yields worse recommendation credibility compared to the subset where neither the condition nor the diffusion target contains uncredible items. This further validates that the uncredible condition and the uncredible diffusion target are indeed the key contributing factors that place DM-based recommendation methods at risk of generating uncredible recommendation results.

Moreover, **although simply removing uncredible items from the datasets can improve recommendation credibility, it significantly deteriorates recommendation accuracy**. This is because uncredible items may also reflect users' genuine preferences, thereby discarding them restricts the model's ability to accurately learn users' true interests. Therefore, it is crucial to design advanced models that can mitigate the recommendation of uncredible content while simultaneously preserving high recommendation accuracy. This is the motivation and research significance of our proposed model, `Disco`.

C.2 Theoretical Analysis

Proof: Uncredible condition can enhance DM's generation of uncredible results

The training of a conditional DM is to maximize $\mathbb{E}_{p_{data}(\mathbf{e}_n,\mathbf{c})}[\log p_\theta(\mathbf{e}_n|\mathbf{c})]$, where $\mathbf{e}_n$ is the diffusion target (i.e., the last item in a user's interaction sequence) and $\mathbf{c}$ is the condition. This training objective pushes the generation toward the real data distribution.

When an uncredible content-related condition $\mathbf{c}^{unc}$ is utilized to guide the generation process, the model aims to maximize $\mathbb{E}_{p_{data}(\mathbf{e}_n,\mathbf{c}^{unc})}\left[\log p_\theta(\mathbf{e}_n|\mathbf{c}^{unc})\right]$. Then, we have:

$$
\begin{aligned}
\mathbb{E}_{p_{data}(\mathbf{e}_n,\mathbf{c}^{unc})}\left[\log p_\theta(\mathbf{e}_n|\mathbf{c}^{unc})\right] =&\mathbb{E}_{p_{data}(\mathbf{c}^{unc})}\mathbb{E}_{p_{data}(\mathbf{e}_n|\mathbf{c}^{unc})}\left[\log p_\theta(\mathbf{e}_n|\mathbf{c}^{unc})\right]\\
=&\mathbb{E}_{p_{data}(\mathbf{c}^{unc})}\left[\int_{\mathbf{e}_n} p_{data}(\mathbf{e}_n|\mathbf{c}^{unc})\log p_\theta(\mathbf{e}_n|\mathbf{c}^{unc})d\mathbf{e}_n\right]\\
=&\mathbb{E}_{p_{data}(\mathbf{c}^{unc})}\left[-H(p_{data}(\mathbf{e}_n^*|\mathbf{c}^{unc}),p_\theta(\mathbf{e}_n^*|\mathbf{c}^{unc})\right]\\
=&\mathbb{E}_{p_{data}(\mathbf{c}^{unc})}\left[-H(p_{data}(\mathbf{e}_n^*|\mathbf{c}^{unc}))\right]\\
&-\mathbb{E}_{p_{data}(\mathbf{c}^{unc})}\left[D_{\mathrm{KL}}(p_{data}(\mathbf{e}_n^*|\mathbf{c}^{unc})\|p_\theta(\mathbf{e}_n^*|\mathbf{c}^{unc}))\right]\\
=&-H_{p_{data}}(\mathcal{E}|\mathcal{C}^{unc})\\
&-\mathbb{E}_{p_{data}(\mathbf{c}^{unc})}\left[D_{\mathrm{KL}}(p_{data}(\mathbf{e}_n^*|\mathbf{c}^{unc})\|p_\theta(\mathbf{e}_n^*|\mathbf{c}^{unc}))\right],
\end{aligned} \tag{3}
$$

where $\mathcal{E}$ represents the whole generation space and $\mathbf{e}_n^* \in \mathcal{E}$. $\mathcal{C}^{unc}$ is the whole space of uncredible condition $\mathbf{c}^{unc}$. $H(\cdot,\cdot)$ is the entropy between two variables or distributions. According to the above derivation, we have:

$$
\begin{aligned}
H_{p_{data}}(\mathcal{E}|\mathcal{C}^{unc}) =&-\mathbb{E}_{p_{data}(\mathbf{e}_n,\mathbf{c}^{unc})}\left[\log p_\theta(\mathbf{e}_n|\mathbf{c}^{unc})\right]\\
&-\mathbb{E}_{p_{data}(\mathbf{c}^{unc})}\left[D_{\mathrm{KL}}(p_{data}(\mathbf{e}_n^*|\mathbf{c}^{unc})\|p_\theta(\mathbf{e}_n^*|\mathbf{c}^{unc}))\right]\\
&\leq -\mathbb{E}_{p_{data}(\mathbf{e}_n,\mathbf{c}^{unc})}\left[\log p_\theta(\mathbf{e}_n|\mathbf{c}^{unc})\right].
\end{aligned} \tag{4}
$$

Ideally, when the model is optimally trained, the D_{KL} term will approach zero, indicating that the conditional generation distribution approaches the real data distribution. Therefore, the mutual information between the whole conditional generation space $\mathcal{E}$ and the whole uncredible condition space $\mathcal{C}^{unc}$ can be calculated as:

$$
\begin{aligned}
I_{p_\theta}(\mathcal{E},\mathcal{C}^{unc}) =&I_{p_{data}}(\mathcal{E},\mathcal{C}^{unc})\\
=&H_{p_{data}}(\mathcal{E}) - H_{p_{data}}(\mathcal{E}|\mathcal{C}^{unc})\\
\geq&H_{p_{data}}(\mathcal{E}) + \mathbb{E}_{p_{data}(\mathbf{e}_n,\mathbf{c}^{unc})}\left[\log p_\theta(\mathbf{e}_n|\mathbf{c}^{unc})\right].
\end{aligned} \tag{5}
$$

The second equation is derived according to the property of mutual information: $I(X,Y) = H(X) - H(X|Y)$. As the training goes on, the second term becomes larger. At the same time, $H_{p_{data}}(\mathcal{E})$ is a constant based on the real data distribution p_{data}. Hence, the lower bound of $I_{p_\theta}(\mathcal{E},\mathcal{C}^{unc})$ also becomes larger. Based on this, we can conclude that training the diffusion model with uncredible conditions increases the mutual information between the generation space and the uncredible condition space. This indicates that the generation space increasingly contains uncredible features reflected in the uncredible conditions.

Proof: Uncredible diffusion target can enhance DM's generation of uncredible results

The optimization loss of existing DM-based recommendation methods can be formulated as:

$$
\mathcal{L} = \mathbb{E}_{t\sim U(0,T)}[\|\mathbf{e}_n^0 - f_\theta\left(\mathbf{e}_n^t,\mathbf{c},t\right)\|_2^2]. \tag{6}
$$

When an uncredible item embedding $\mathbf{e}_j$ ($j \in \mathcal{I}_{unc}$) is used as the diffusion target (i.e., uncredible diffusion target) during training, the diffusion loss encourages the prediction direction of the diffusion network f_θ to move closer to $\mathbf{e}_j$. Specifically, the MSE distance between the diffusion target and the output of f_θ will be smaller, indicating higher similarity.

In the inference stage, the generation process of diffusion recommenders can be expressed as:

$$
\mathbf{e}_n^{t-1} = w_1 f_\theta(\mathbf{e}_n^t,\mathbf{c},t) + w_2\mathbf{e}_n^t + w_3\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I}), \tag{7}
$$

where $w_1 = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}$, $w_2 = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}$, and $w_3 = \sqrt{\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}(1-\alpha_t)}$. This generation process is performed step by step, and the final embedding $\mathbf{e}_n^0$ is taken as the generation result, which then serves as the reference for item prediction and recommendations.

Let $\mathbf{e}_n^{t-1}$ denote the generated embedding at step $t-1$ without using uncredible diffusion target $\mathbf{e}_j$ during training. In such case, the parameters of the diffusion network are denoted as θ. Similarly, let $\hat{\mathbf{e}}_n^{t-1}$ denote the generated embedding at step $t-1$ with $\mathbf{e}_j$ as the uncredible diffusion target during training. In this case, the diffusion parameters are denoted as $\hat{\theta}$. We then calculate the difference in similarity between the normalized $\mathbf{e}_j$ and the normalized generated embeddings $\hat{\mathbf{e}}_n^{t-1}$ and $\mathbf{e}_n^{t-1}$ at step $t-1$ as follows:

$$\begin{aligned}
\Delta^{t-1} &= \mathrm{sim}(\mathbf{e}_j, \hat{\mathbf{e}}_n^{t-1}) - \mathrm{sim}(\mathbf{e}_j, \mathbf{e}_n^{t-1}) \\
&= \left[w_1 \left(f_{\hat{\theta}}(\hat{\mathbf{e}}_n^t, \mathbf{c}, t) - f_\theta(\mathbf{e}_n^t, \mathbf{c}, t)\right) + w_2\left(\hat{\mathbf{e}}_n^t - \mathbf{e}_n^t\right) + \cancel{w_3(\boldsymbol{\epsilon}^t - \boldsymbol{\epsilon}^t)}\right] \cdot \mathbf{e}_j^\top \\
&= w_1 \left(f_{\hat{\theta}}(\hat{\mathbf{e}}_n^t, \mathbf{c}, t) - f_\theta(\mathbf{e}_n^t, \mathbf{c}, t)\right) \cdot \mathbf{e}_j^\top + w_2\left(\hat{\mathbf{e}}_n^t - \mathbf{e}_n^t\right) \cdot \mathbf{e}_j^\top .
\end{aligned} \tag{8}$$

Here, we utilize the dot product to calculate the similarity. To avoid the interference from sampled noise, we use $\boldsymbol{\epsilon}^t$ to denote the sample noise $\boldsymbol{\epsilon}$ in step $t-1$, and use it for both generation processes to control this variable.

When $t = T$, we have:

$$\begin{aligned}
\Delta^{T-1} &= \mathrm{sim}(\mathbf{e}_j, \hat{\mathbf{e}}_n^{T-1}) - \mathrm{sim}(\mathbf{e}_j, \mathbf{e}_n^{T-1}) \\
&= w_1 \left(f_{\hat{\theta}}(\hat{\mathbf{e}}_n^T, \mathbf{c}, T) - f_\theta(\mathbf{e}_n^T, \mathbf{c}, T)\right) \cdot \mathbf{e}_j^\top + w_2\left(\hat{\mathbf{e}}_n^T - \mathbf{e}_n^T\right) \cdot \mathbf{e}_j^\top \\
&= w_1 \left(f_{\hat{\theta}}(\boldsymbol{\epsilon}^T, \mathbf{c}, T) - f_\theta(\boldsymbol{\epsilon}^T, \mathbf{c}, T)\right) \cdot \mathbf{e}_j^\top + \cancel{w_2\left(\boldsymbol{\epsilon}^T - \boldsymbol{\epsilon}^T\right) \cdot \mathbf{e}_j^\top} \\
&= w_1 \left(f_{\hat{\theta}}(\boldsymbol{\epsilon}^T, \mathbf{c}, T) \cdot \mathbf{e}_j^\top - f_\theta(\boldsymbol{\epsilon}^T, \mathbf{c}, T) \cdot \mathbf{e}_j^\top\right) \\
&= \underbrace{w_1}_{>0} \underbrace{\left(\mathrm{sim}\left(\mathbf{e}_j, f_{\hat{\theta}}(\boldsymbol{\epsilon}^T, \mathbf{c}, T)\right) - \mathrm{sim}\left(\mathbf{e}_j, f_\theta\left(\boldsymbol{\epsilon}^T, \mathbf{c}, T\right)\right)\right)}_{>0} > 0.
\end{aligned} \tag{9}$$

We control the process of two generations start from the same point $\hat{\mathbf{e}}_n^T = \mathbf{e}_n^T = \boldsymbol{\epsilon}^T$ for fair comparison. As mentioned earlier, the prediction direction of $f_{\hat{\theta}}$ is closer to $\mathbf{e}_j$ than that of f_θ. Hence, the MSE distance between the output of $f_{\hat{\theta}}$ and to $\mathbf{e}_j$ is smaller than that between the output of f_θ and $\mathbf{e}_j$. When the embeddings are normalized, a smaller MSE distance corresponds to a higher dot product similarity. Consequently, $\mathrm{sim}\left(\mathbf{e}_j, f_{\hat{\theta}}(\boldsymbol{\epsilon}^T, \mathbf{c}, T)\right) - \mathrm{sim}(\mathbf{e}_j, f_\theta\left(\boldsymbol{\epsilon}^T, \mathbf{c}, T\right)) > 0$. At the same time, $w_1 > 0$, therefore $\Delta^{T-1} > 0$. This indicates that, when starting from the same initial point, the generation result at step $T-1$ produced by model $f_{\hat{\theta}}$, which has been trained with an uncredible diffusion target, will be more similar to this uncredible diffusion target.

When $t = T-1$, we have:

$$\begin{aligned}
\Delta^{T-2} = \;& w_1 \left(f_{\hat{\theta}}(\hat{\mathbf{e}}_n^{T-1}, \mathbf{c}, T-1) - f_\theta(\mathbf{e}_n^{T-1}, \mathbf{c}, T-1)\right) \cdot \mathbf{e}_j^\top + w_2\left(\hat{\mathbf{e}}_n^{T-1} - \mathbf{e}_n^{T-1}\right) \cdot \mathbf{e}_j^\top \\
= \;& w_1 \mathrm{C}_{T-1}^+ + w_2\left(\mathrm{sim}(\mathbf{e}_j, \hat{\mathbf{e}}_n^{T-1}) - \mathrm{sim}(\mathbf{e}_j, \mathbf{e}_n^{T-1})\right) \\
= \;& w_1 \mathrm{C}_{T-1}^+ + w_2 \Delta^{T-1}.
\end{aligned} \tag{10}$$

As mentioned before, when a uncredible item embedding $\mathbf{e}_j$ is taken for training, the diffusion loss will encourage the prediction direction of $f_{\hat{\theta}}$ closer to $\mathbf{e}_j$. At the same time, $\hat{\mathbf{e}}_n^{T-1}$ is closer to $\mathbf{e}_j$, as compared to that of $\mathbf{e}_n^{T-1}$. This further enforces $f_{\hat{\theta}}(\hat{\mathbf{e}}_n^{T-1}, \mathbf{c}, T-1)$ more similar to $\mathbf{e}_j$, than that of $f_\theta(\mathbf{e}_n^{T-1}, \mathbf{c}, T-1)$. Hence, the first term is a positive constant, and we denote it by C_{T-1}^+.

Similarly, when $t < T-1$, we have:

$$\begin{aligned}
\Delta^{T-3} &= w_1 \mathrm{C}_{T-2}^+ + w_2 \Delta^{T-2} \\
&= w_1 \mathrm{C}_{T-2}^+ + w_2(w_1 \mathrm{C}_{T-1}^+ + w_2 \Delta^{T-1}) \\
&= w_1 \mathrm{C}_{T-2}^+ + w_1 w_2 \mathrm{C}_{T-1}^+ + w_2^2 \Delta^{T-1}.
\end{aligned} \tag{11}$$

$$\begin{aligned}
\Delta^{T-4} &= w_1 \mathrm{C}_{T-3}^+ + w_2 \Delta^{T-3} \\
&= w_1 \mathrm{C}_{T-3}^+ + w_2(w_1 \mathrm{C}_{T-2}^+ + w_1 w_2 \mathrm{C}_{T-1}^+ + w_2^2 \Delta^{T-1}) \\
&= w_1 \mathrm{C}_{T-3}^+ + w_1 w_2 \mathrm{C}_{T-2}^+ + w_1 w_2^2 \mathrm{C}_{T-1}^+ + w_2^3 \Delta^{T-1}.
\end{aligned} \tag{12}$$

$$\cdots$$

$$\begin{aligned}
\Delta^0 &= w_1 \mathrm{C}_1^+ + w_1 w_2 \mathrm{C}_2^+ + \cdots + w_1 w_2^{T-2} \mathrm{C}_{T-1}^+ + w_2^{T-1} \Delta^{T-1} \\
&= \underbrace{\sum_m^{T-1} w_1 w_2^{m-1} \mathrm{C}_m^+}_{>0} + \underbrace{w_2^{T-1} \Delta^{T-1}}_{>0} \\
&= \mathrm{sim}(\mathbf{e}_j, \hat{\mathbf{e}}_n^0) - \mathrm{sim}(\mathbf{e}_j, \hat{\mathbf{e}}_n^0) \\
&> 0.
\end{aligned} \tag{13}$$

According to above analysis, the final generated result $\hat{\mathbf{e}}_n^0$ using diffusion network $f_{\hat{\theta}}$ is more similar with uncredible item embedding $\mathbf{e}_j$, as compared to the final generated result $\mathbf{e}_n^0$ using diffusion network f_θ. This indicates that when uncredible items are used as the diffusion targets during training, the model tends to generate outputs that carry more uncredible features, i.e., embeddings that are more similar to uncredible items.

Algorithm 1 Training of `Disco`

1: **Input:** Training dataset $\mathcal{S}_{train} = \{(\mathbf{e}_n, \mathbf{e}_{neg}, s^{pre}, s^{unc})\}_{s=1}^{|\mathcal{S}_{train}|}$, available uncredible item set $\mathcal{I}_{unc}$, trainable parameters Θ, total diffusion steps T, learning rate η, variance schedules $\{\alpha_t\}_{t=0}^T$.
2: **Output:** Optimized parameters Θ.
3: $\mathbf{F} = \mathrm{Stack}(\{\mathbf{e}_i^{unc}\}_{i \in \mathcal{I}_{unc}})$ ▷ Construct uncredible feature matrix
4: **repeat**
5: $(\mathbf{e}_n, \mathbf{e}_{neg}, s^{pre}, s^{unc}) \sim \mathcal{S}_{train}$ ▷ Sample training data
6: $\mathbf{c}^{pre} = \mathrm{Tramsformer}(s^{pre})$ ▷ Obtain preference-related condition
7: $\mathbf{c}^{unc} = \mathrm{Mean}(s^{unc})$ ▷ Obtain uncredible content-related condition
8: **Update F** by Algorithm 3 ▷ Progressive uncredible feature matrix enhancement
9: $[\mathbf{U}_1; \mathbf{U}_2], \mathbf{\Lambda}, \mathbf{V} = \mathrm{SVD}(\mathbf{F}^\top)$ ▷ Construct null space of uncredible feature matrix
10: $\tilde{\mathbf{e}}_n = \mathbf{e}_n \mathbf{U}_2 \mathbf{U}_2^\top$ ▷ Credible subspace projection for diffusion target $\mathbf{e}_n$
11: $\tilde{\mathbf{e}}_n = (\tilde{\mathbf{e}}_n + \mathbf{e}_n)/2$ ▷ Residual connection
12: $t \sim \mathrm{Uniform}(1, T)$ ▷ Sample diffusion step
13: $\tilde{\mathbf{e}}_n^t = \sqrt{\bar{\alpha}_t}\tilde{\mathbf{e}}_n^0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$ ▷ Add noise to the embedding of diffusion target
14: $\mathbf{e}_{neg}^t = \sqrt{\bar{\alpha}_t}\mathbf{e}_{neg}^0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$ ▷ Add noise to the embedding of negative preference item
15: $\Theta \leftarrow \Theta - \eta \nabla_\Theta \mathcal{L}_{\texttt{Disco}}(\tilde{\mathbf{e}}_n, \mathbf{e}_{neg}, \mathbf{c}^{pre}, \mathbf{c}^{unc}, t, \Theta)$ ▷ Update parameters
16: **until** convergence
17: **return** Θ

Algorithm 2 Inference of `Disco`

1: **Input:** Test dataset $\mathcal{S}_{test} = \{s^{pre}\}_{s=1}^{|\mathcal{S}_{test}|}$, trained diffusion network parameters $\theta \in \Theta$, total reverse steps T, DDIM steps T', variance schedules $\{\alpha_t\}_{t=0}^T$.
2: **Output:** A recommendation list for each user/sequence.
3: $s^{pre} \sim \mathcal{S}_{test}$ ▷ Sample test sequence
4: $\mathbf{c}^{pre} = \mathrm{Transformer}(s^{pre})$ ▷ Obtain preference-related condition
5: **for** $t' = T', \cdots, 1$ **do**
6: $t = \lfloor t' \times (T/T') \rfloor$ ▷ Calculate DDIM denoising step
7: $\mathbf{e}_n^T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ▷ Start from Gaussian noise
8: $\mathbf{e}_n^{t-1} = \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t} f_\theta(\mathbf{e}_n^t, \mathbf{c}^{pre}, t) + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{e}_n^t + \sqrt{\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}(1-\alpha_t)}\boldsymbol{\epsilon}$ ▷ Step-by-step generation
9: **end for**
10: $\hat{y}_i = \mathbf{e}_n^0 \cdot \mathbf{e}_i^\top$ ▷ Calculate the matching score between a user/sequence and a candidate item $\mathbf{e}_i$
11: $\mathcal{R} = \{i | \mathrm{TopK}(\hat{y}_i), i \in \mathcal{I}\}$ ▷ Select top K items with highest matching scores
12: **return** $\mathcal{R}$

Algorithm 3 Progressive enhancement of uncredible feature matrix

1: **Input:** Original uncredible feature matrix **F**, available uncredible item set $\mathcal{I}_{unc}$, current iteration j, maximum selection ratio γ, maximum iteration m to reach γ.
2: **Output:** Updated uncredible feature matrix **F**.
3: $\mathrm{UD}(i) = \frac{1}{|\mathcal{I}_{unc}|}\sum_{i' \in \mathcal{I}_{unc}} \cos(\mathbf{e}_i^{unc}, \mathbf{e}_{i'}^{unc})$ ▷ Calculate uncredible degree of items in $\mathcal{I} \setminus \mathcal{I}_{unc}$
4: $ratio(j) = \min(\gamma, \frac{j}{m}\gamma)$ ▷ Calculate the selection ratio at current iteration
5: select $\lfloor |\mathcal{I} \setminus \mathcal{I}_{unc}| \cdot ratio(j) \rfloor$ items with highest uncredible degree ▷ Select potential uncredible items
6: Add potential uncredible items into $\mathcal{I}_{unc}$ ▷ Extension of uncredible item set
7: $\mathbf{F} = \mathrm{Stack}(\{\mathbf{e}_i^{unc}\}_{i \in \mathcal{I}_{unc}})$ ▷ Enhancement of uncredible feature matrix
8: **return F**

D Derivation of Equation 5

In this section, we provide the derivation of Equation 5. For simplicity, we only need to derive the first term, since the derivation of the second term follows the same procedure. The detailed derivation is as follows:

$$
\begin{aligned}
-\mathbb{E}_q\left[\log \frac{p_\theta\left(\mathbf{e}_n^{0:T}|\mathbf{c}^{pre}\right)}{q(\mathbf{e}_n^{1:T}|\mathbf{e}_n^0)}\right] &\overset{①}{=} -\mathbb{E}_q\left[\log \frac{p(\mathbf{e}_n^T|\mathbf{c}^{pre})p_\theta(\mathbf{e}_n^0|\mathbf{e}_n^1,\mathbf{c}^{pre})\prod_{t>1}^{T}p_\theta\left(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{c}^{pre}\right)}{q(\mathbf{e}_n^1|\mathbf{e}_n^0)\prod_{t>1}^{T}q(\mathbf{e}_n^t|\mathbf{e}_n^{t-1},\mathbf{e}_n^0)}\right] \\
&\overset{②}{=} -\mathbb{E}_q\left[\log \frac{p(\mathbf{e}_n^T|\mathbf{c}^{pre})p_\theta(\mathbf{e}_n^0|\mathbf{e}_n^1,\mathbf{c}^{pre})\prod_{t>1}^{T}p_\theta\left(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{c}^{pre}\right)}{q(\mathbf{e}_n^1|\mathbf{e}_n^0)\prod_{t>1}^{T}\frac{q(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{e}_n^0)q(\mathbf{e}_n^t|\mathbf{e}_n^0)}{q(\mathbf{e}_n^{t-1}|\mathbf{e}_n^0)}}\right] \\
&\overset{③}{=} -\mathbb{E}_q\left[\log \frac{p(\mathbf{e}_n^T|\mathbf{c}^{pre})p_\theta(\mathbf{e}_n^0|\mathbf{e}_n^1,\mathbf{c}^{pre})\prod_{t>1}^{T}p_\theta\left(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{c}^{pre}\right)}{\cancel{q(\mathbf{e}_n^1|\mathbf{e}_n^0)}\frac{\cancel{q(\mathbf{e}_n^2|\mathbf{e}_n^0)}}{\cancel{q(\mathbf{e}_n^1|\mathbf{e}_n^0)}}\cdots\frac{q(\mathbf{e}_n^T|\mathbf{e}_n^0)}{\cancel{q(\mathbf{e}_n^{T-1}|\mathbf{e}_n^0)}}\prod_{t>1}^{T}q(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{e}_n^0)}\right] \\
&\overset{④}{=} -\mathbb{E}_q\left[\log p(\mathbf{e}_n^0|\mathbf{e}_n^1,\mathbf{c}^{pre}) + \log\frac{p_\theta(\mathbf{e}_n^T)}{q(\mathbf{e}_n^T|\mathbf{e}_n^0)} + \log\frac{\prod_{t>1}^{T}p_\theta\left(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{c}^{pre}\right)}{\prod_{t>1}^{T}q(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{e}_n^0)}\right] \\
&\overset{⑤}{=} -\mathbb{E}_q\left[\log p(\mathbf{e}_n^0|\mathbf{e}_n^1,\mathbf{c}^{pre})\right] - \mathbb{E}_q\left[\log\frac{p_\theta(\mathbf{e}_n^T)}{q(\mathbf{e}_n^T|\mathbf{e}_n^0)}\right] \\
&\qquad -\sum_{t>1}^{T}\mathbb{E}_q\left[\log\frac{p_\theta\left(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{c}^{pre}\right)}{q(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{e}_n^0)}\right] \\
&\overset{⑥}{=} -\underbrace{\mathbb{E}_q\left[\log p(\mathbf{e}_n^0|\mathbf{e}_n^1,\mathbf{c}^{pre})\right]}_{\text{reconstruction term}} + \underbrace{D_{\mathrm{KL}}\left(q(\mathbf{e}_n^T|\mathbf{e}_n^0)\|p_\theta(\mathbf{e}_n^T)\right)}_{\text{prior matching term}} \\
&\qquad + \underbrace{\sum_{t>1}^{T}\mathbb{E}_q\left[D_{\mathrm{KL}}\left(q(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{e}_n^0)\|p_\theta\left(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{c}^{pre}\right)\right)\right]}_{\text{denoising matching term}}.
\end{aligned}
\tag{14}
$$

Equation ② is derived through Bayes rule: $q(\mathbf{e}_n^t|\mathbf{e}_n^{t-1},\mathbf{e}_n^0) = \frac{q(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{e}_n^0)q(\mathbf{e}_n^t|\mathbf{e}_n^0)}{q(\mathbf{e}_n^{t-1}|\mathbf{e}_n^0)}$. Equation ④ is obtained since $p(\mathbf{e}_n^T|\mathbf{c}^{pre}) = p(\mathbf{e}_n^T)$ given $\mathbf{e}_n^T \sim \mathcal{N}(\mathbf{0},\mathbf{I})$, which is independent with condition $\mathbf{c}^{pre}$. DMs generally optimize the denoising matching term $D_{\mathrm{KL}}\left(q(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{e}_n^0)\|p_\theta\left(\mathbf{e}_n^{t-1}|\mathbf{e}_n^t,\mathbf{c}^{pre}\right)\right)$ instead of the whole variational bound. Then, this denoising matching term can be derived into the optimization loss $\mathcal{L} = \mathbb{E}_{\mathbf{e}_n^0,\mathbf{c}^{pre},t}\left[\frac{1}{2\sigma_t^2}\left|\left|\boldsymbol{\mu}_q(\mathbf{e}_n^t,\mathbf{e}_n^0) - \boldsymbol{\mu}_\theta(\mathbf{e}_n^t,\mathbf{c}^{pre},t)\right|\right|_2^2\right]$, by adding the condition $\mathbf{c}^{pre}$ into $\boldsymbol{\mu}_\theta(\mathbf{e}_n^t,t)$ in (Ho et al., 2020). Similar with (Pathak et al., 2023), $\boldsymbol{\mu}_q(\mathbf{e}_n^t,\mathbf{e}_n^0)$ is defined as (Pathak et al., 2023):

$$
\boldsymbol{\mu}_q(\mathbf{e}_n^t,\mathbf{e}_n^0) = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})\mathbf{e}_n^t + \sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)\mathbf{e}_n^0}{1-\bar{\alpha}_t}. \tag{15}
$$

In our model, $\boldsymbol{\mu}_\theta(\mathbf{e}_n^t,\mathbf{c}^{pre},t)$ is defined as:

$$
\boldsymbol{\mu}_\theta(\mathbf{e}_n^t,\mathbf{c}^{pre},t) = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})\mathbf{e}_n^t + \sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)f_\theta(\mathbf{e}_n^t,\mathbf{c}^{pre},t)}{1-\bar{\alpha}_t}, \tag{16}
$$

where $f_\theta(\mathbf{e}_n^t,\mathbf{c}^{pre},t)$ is the predicted $\mathbf{e}_n^0$ using the diffusion network f_θ.

Then, the optimization term can be rewritten as:

$$\begin{aligned}\mathcal{L} =&\mathbb{E}_{\mathbf{e}_n^0,\mathbf{c}^{pre},t}\left[\frac{1}{2\sigma_t^2}\left\|\boldsymbol{\mu}_q(\mathbf{e}_n^t,\mathbf{e}_n^0)-\boldsymbol{\mu}_\theta(\mathbf{e}_n^t,t)\right\|_2^2\right]\\ =&\mathbb{E}_{\mathbf{e}_n^0,\mathbf{c}^{pre},t}\left[\frac{1}{2\sigma_t^2}\left\|\frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})\mathbf{e}_n^t+\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)\mathbf{e}_n^0}{1-\bar{\alpha}_t}\right.\right.\\ &\left.\left.-\frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})\mathbf{e}_n^t+\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)f_\theta(\mathbf{e}_n^t,\mathbf{c}^{pre},t)}{1-\bar{\alpha}_t}\right\|_2^2\right]\\ =&\mathbb{E}_{\mathbf{e}_n^0,\mathbf{c}^{pre},t}\left[\frac{1}{2\sigma_q^2(t)}\left\|\frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}\mathbf{e}_n^0-\frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}f_\theta(\mathbf{e}_n^t,\mathbf{c}^{pre},t)\right\|_2^2\right]\\ =&\mathbb{E}_{\mathbf{e}_n^0,\mathbf{c}^{pre},t}\left[\frac{1}{2\sigma_q^2(t)}\left(\frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}\right)^2\left\|\mathbf{e}_n^0-f_\theta(\mathbf{e}_n^t,\mathbf{c}^{pre},t)\right\|_2^2\right].\end{aligned} \tag{17}$$

In practice, the coefficient $\frac{1}{2\sigma_q^2(t)}\left(\frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}\right)^2$ is generally omitted (Ho et al., 2020). Hence, the optimization loss of our preference-related condition guided generation can be rewritten as $\mathcal{L}_{pre}=\mathbb{E}_{\mathbf{e}_n^0,\mathbf{c}^{pre},t}\left[\left\|\mathbf{e}_n^0-f_\theta(\mathbf{e}_n^t,\mathbf{c}^{pre},t)\right\|_2^2\right]$. Similarly, the optimization loss of uncredible content-related condition guided generation is: $\mathcal{L}_{unc}=\mathbb{E}_{\mathbf{e}_n^0,\mathbf{c}^{unc},t}\left[\left\|\mathbf{e}_n^0-f_\theta(\mathbf{e}_n^t,\mathbf{c}^{unc},t)\right\|_2^2\right]$. Our `Disco` model aims to encourage the generation guided by preference-related condition and discourage the generation guided by uncredible content-related condition. To achieve this goal, the optimization objective is formulated as shown in Equation **??**:

$$\mathcal{L}=\mathcal{L}_{pre}-\mathcal{L}_{unc}=\mathbb{E}_{\mathbf{e}_n^0,\mathbf{c}^{pre},t}\left[\left\|\mathbf{e}_n^0-f_\theta(\mathbf{e}_n^t,\mathbf{c}^{pre},t)\right\|_2^2\right]-\mathbb{E}_{\mathbf{e}_n^0,\mathbf{c}^{unc},t}\left[\left\|\mathbf{e}_n^0-f_\theta(\mathbf{e}_n^t,\mathbf{c}^{unc},t)\right\|_2^2\right]. \tag{18}$$

E CASE STUDY

In this section, we conduct a case study using the GossipCop dataset to evaluate the effectiveness of `Disco`. The GossipCop dataset contains users' interaction sequences with news articles, including both true news (i.e., credible items) and fake news (i.e., uncredible items). Specifically, we present the historical interaction sequences and recommendation lists for five users. The credible content items are marked in green, while uncredible items are marked in red. In addition, to illustrate the semantic relevance between content items, we utilize the same background color to highlight content with similar or related topics. From Table 5, we have the following observations:

- `Disco` demonstrates strong capability in delivering credible recommendations. Specifically, although all these users have interacted with uncredible items in their historical interaction sequences, the recommendation lists generated by `Disco` contain no uncredible content.
- `Disco` is capable of mitigating uncredible content while still preserving high recommendation accuracy. This is achieved by removing uncredible features while retaining users' genuine preference-related information. For example, taking User4 as an example, this user had historically interacted with some news (including fake news) about the death of celebrities (highlighted in yellow). `Disco` can effectively capture this user's genuine preference and recommend some content also in such topics. It is worth noting that User4 had interacted with fake news about the death of "Tom Petty", and `Disco` recommends this user with a credible news article about the same event. This plays an important role in countering misinformation, as it helps users correct false impressions formed through prior exposure to uncredible content.

F USAGE CLAIM OF LARGE LANGUAGE MODELS

We only utilize ChatGPT for polishing the academic writing, with the prompt "Proofread the grammar and polish the writing of the given sentences".

Table 5: Five cases showcasing the historical interaction sequences and the recommendation lists of five users sampled from GossipCop dataset. Credible refers to credible content (i.e., true news) and Uncredible refers to uncredible content (i.e., fake news). In a user sample, the texts marked by the same-color background refer to similar topics. "**ground truth**" means the corresponding recommended content items have been actually read by the user in the test set.

User1	Historical sequence	Credible: Justin Timberlake, Chris Stapleton release 'Say Something' song, video.	Uncredible: Nicole Kidman, Keith Urban: Secrets to a Successful Relationship.	Uncredible:Kendall Jenner Shades Scott Disick Over Photo With Sofia Richie and His Kids.	Uncredible: Grammy winners 2018: the complete list.	
	Recommendations	Credible 2018 Latin GRAMMY Awards Complete Winners List.	Credible: Weinstein Company Files for Bankruptcy and Revokes Nondisclosure Agreements.	Credible: Oscars: The Complete Winners List.	Credible: Pop superstar Lady Gaga has officially landed her first Las Vegas residency.	Credible (**ground truth**): TV News Roundup: Netflix Reveals Fuller House Season 4 Premiere Date
User2	Historical sequence	Credible: 13 Nights Of Halloween 2017 Schedule: Full List of Movies.	Uncredible: Taylor Swift will reportedly keep her new album off streaming services like Spotify and Apple Music for a week.	Uncredible: Former NBC interviewer lashes out at Trump in an NYT op-ed for reportedly casting doubt on the authenticity of the infamous tape.	Credible: 'Big Little Lies' Season 2 News, Premiere Date & Cast.	
	Recommendations	Credible (**ground truth**): Justin Timberlake Announces New Album Man of the Woods.	Credible: Seven-time and defending champion says she isn't quite ready to return after giving birth to daughter in September.	Credible: Pop superstar Lady Gaga has officially landed her first Las Vegas residency.	Credible: Jamie Lynn Spears' second child on the way will join big sister Maddie Briann.	Credible: "Good morning baby of mine, John Stamos' fiance Caitlin McHugh wrote as she debuted her baby bump...
User3	Historical sequence	Credible: Hugh Grant and Anna Eberstein's baby on the way joins their daughter.	Uncredible: The cancellation of the third Sex and the City film came with headline-making fallout something Sarah Jessica Parker struggled with	Uncredible: Selena Gomez has completed her treatment for depression and anxiety and is reported feeling	Credible: Congratulations are in order for Rachel McAdams the 39-year-old actress is reportedly going to be a first-time mom! Though she has not personally confirmed the baby news	
	Recommendations	Credible: All Chicago West Baby Photos Timeline.	Credible: Demi Lovato Says She Contemplated Suicide at Age 7.	Credible: 'Black Panther' is the most tweeted about movie ever.	Credible (**ground truth**): His wife Faith Hill said the country star had been suffering from dehydration.	Credible: Tisha Campbell-Martin Files For Divorce From Husband of 21 Years
User4	Historical sequence	Uncredible: Caitlyn Jenner told Diane Sawyer that she had undergone the final surgery in her gender reassignment procedures on Friday night's 20/20 special.	Credible: Indiana police found the actress unresponsive after responding to a 911 call Saturday.	Credible: Roger Ailes, Former Fox News CEO, Dies At 77.	Uncredible: Tom Petty Dead: Celebrities React on Social Media Variety.	
	Recommendations	Credible: An emotional Celine Dion returned to the stage in Las Vegas on Tuesday night.	Credible (**ground truth**): Rocker Tom Petty died Monday after being rushed to a Los Angeles hospital.	Credible: Hugh Hefner's death certificate from the Los Angeles County Department of Public Health.	Credible: The final season of Netflix's "House of Cards" keeps the secret of how Frank Underwood died until the very end.	Credible: Pauley Perrette announces she's leaving "NCIS" after 15 seasons.
User5	Historical sequence	Credible: Benjamin Glaze had never kissed a girl before Katy Perry tricked him during the ABC reboot of American Idol.	Uncredible: During her chat with Ryan Murphy Friday (March 16) for the opening night of PaleyFest in Los Angeles.	Credible: A longtime aerialist for the famed Cirque Du Soleil plummeted to his death in front of a horrified crowd in Florida on Saturday night while trying out a new act...	Uncredible: Justin Bieber's struggling with his split from Selena Gomez as she's all smiles on her Australian vacation. Here's how the Biebs is coping with his...	
	Recommendations	Credible (**ground truth**): Justin Bieber Wants to Be With Selena Gomez But Is Hanging With Baskin Champion.	Credible: The singer covered Ariana Grande's 'Just a Little Bit of Your Heart' in the arena where her concert was attacked...	Credible: Trevorrow helmed the rebooted franchise's first installment.	Credible: Voting closes at 5pm PT today (June 29) for this year's News' TV Scoop Awards...	Credible: Blake Shelton Gets His Palms Read With Jimmy Fallon, Jokes About Having Too Much Sex.

REFERENCES

Zhuo Cai, Shoujin Wang, Victor W Chu, Usman Naseem, Yang Wang, and Fang Chen. Unleashing the potential of diffusion models towards diversified sequential recommendations. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1476–1486, 2025.

Jianxin Chang, Chen Gao, Yu Zheng, Yiqun Hui, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. Sequential recommendation with graph neural networks. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 378–387, 2021.

Ziqiang Cui, Haolun Wu, Bowei He, Ji Cheng, and Chen Ma. Context matters: Enhancing sequential recommendation with context-aware diffusion-based contrastive learning. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pp. 404–414, 2024a.

Ziqiang Cui, Haolun Wu, Bowei He, Ji Cheng, and Chen Ma. Diffusion-based contrastive learning for sequential recommendation. *arXiv preprint arXiv:2405.09369*, 2024b.

Yashar Deldjoo, Zhankui He, Julian McAuley, Anton Korikov, Scott Sanner, Arnau Ramisa, René Vidal, Maheswaran Sathiamoorthy, Atoosa Kasirzadeh, and Silvia Milano. A review of modern recommender systems using generative models (gen-recsys). In *Proceedings of the 30th ACM SIGKDD conference on Knowledge Discovery and Data Mining*, pp. 6448–6458, 2024.

Miriam Fernandez, Alejandro Bellogín, and Iván Cantador. Analysing the effect of recommendation algorithms on the spread of misinformation. In *Proceedings of the 16th ACM Web Science Conference*, pp. 159–169, 2024.

Jesse Harte, Wouter Zorgdrager, Panos Louridas, Asterios Katsifodimos, Dietmar Jannach, and Marios Fragkoulis. Leveraging large language models for sequential recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 1096–1102, 2023.

Ruining He and Julian McAuley. Fusing similarity models with markov chains for sparse sequential recommendation. In *IEEE International Conference on Data Mining*, pp. 191–200, 2016.

Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Guoqing Hu, Zhengyi Yang, Zhibo Cai, An Zhang, and Xiang Wang. Generate and instantiate what you prefer: Text-guided diffusion for sequential recommendation. *arXiv preprint arXiv:2410.13428*, 2024.

Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *IEEE International Conference on Data Mining*, pp. 197–206, 2018.

Jin Li, Shoujin Wang, Qi Zhang, Shui Yu, and Fang Chen. Generating with fairness: A modality-diffused counterfactual framework for incomplete multimodal recommendations. In *Proceedings of the ACM on Web Conference 2025*, pp. 2787–2798, 2025a.

Wuchao Li, Rui Huang, Haijun Zhao, Chi Liu, Kai Zheng, Qi Liu, Na Mou, Guorui Zhou, Defu Lian, Yang Song, et al. Dimerec: a unified framework for enhanced sequential recommendation via generative diffusion models. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, pp. 726–734, 2025b.

Zihao Li, Aixin Sun, and Chenliang Li. Diffurec: A diffusion model for sequential recommendation. *ACM Transactions on Information Systems*, 42(3):1–28, 2023.

Jianghao Lin, Jiaqi Liu, Jiachen Zhu, Yunjia Xi, Chengkai Liu, Yangtian Zhang, Yong Yu, and Weinan Zhang. A survey on diffusion models for recommender systems. *arXiv preprint arXiv:2409.05033*, 2024.

Qidong Liu, Fan Yan, Xiangyu Zhao, Zhaocheng Du, Huifeng Guo, Ruiming Tang, and Feng Tian. Diffusion augmentation for sequential recommendation. In *Proceedings of the 32nd ACM International conference on information and knowledge management*, pp. 1576–1586, 2023.

Shuo Liu, An Zhang, Guoqing Hu, Hong Qian, and Tat-seng Chua. Preference diffusion for recommendation. In *ICLR*, 2025a.

Ziwei Liu, Qidong Liu, Yejing Wang, Wanyu Wang, Pengyue Jia, Maolin Wang, Zitao Liu, Yi Chang, and Xiangyu Zhao. Sigma: Selective gated mamba for sequential recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12264–12272, 2025b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Haokai Ma, Ruobing Xie, Lei Meng, Xin Chen, Xu Zhang, Leyu Lin, and Zhanhui Kang. Plug-in diffusion model for sequential recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 8886–8894, 2024a.

Haokai Ma, Ruobing Xie, Lei Meng, Yimeng Yang, Xingwu Sun, and Zhanhui Kang. Seedrec: sememe-based diffusion for sequential recommendation. In *Proceedings of IJCAI*, pp. 1–9, 2024b.

Haokai Ma, Yimeng Yang, Lei Meng, Ruobing Xie, and Xiangxu Meng. Multimodal conditioned diffusion model for recommendation. In *Companion Proceedings of the ACM on Web Conference*, pp. 1733–1740, 2024c.

Zihan Ma, Minnan Luo, Yiran Hao, Zhi Zeng, Xiangzheng Kong, and Jiahao Wang. Bridging interests and truth: Towards mitigating fake news with personalized and truthful recommendations. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 490–503, 2025.

Royal Pathak, Francesca Spezzano, and Maria Soledad Pera. Understanding the contribution of recommendation algorithms on misinformation recommendation and misinformation dissemination on social networks. *ACM Transactions on the Web*, 17(4):1–26, 2023.

Yuanpeng Qu and Hajime Nobuhara. Intent-aware diffusion with contrastive learning for sequential recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1552–1561, 2025.

Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 452–461, 2009.

Leheng Sheng, An Zhang, Yi Zhang, Yuxin Chen, Xiang Wang, and Tat-Seng Chua. Language representations can be what recommenders need: Findings and potentials. In *The Thirteenth International Conference on Learning Representations*, 2025.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *ACM International Conference on Information and Knowledge Management*, pp. 1441–1450, 2019.

Jiaxi Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pp. 565–573, 2018.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

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

Chenyang Wang, Weizhi Ma, Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. Sequential recommendation with multiple contrast signals. *ACM Transactions on Information Systems*, 41 (1):1–27, 2023a.

Shoujin Wang, Liang Hu, Yan Wang, Longbing Cao, Quan Z. Sheng, and Mehmet Orgun. Sequential recommender systems: Challenges, progress and prospects. In *Twenty-Eighth International Joint Conference on Artificial Intelligence* {*IJCAI-19*}. International Joint Conferences on Artificial Intelligence Organization, 2019.

Shoujin Wang, Longbing Cao, Yan Wang, Quan Z Sheng, Mehmet A Orgun, and Defu Lian. A survey on session-based recommender systems. *ACM Computing Surveys*, 54(7):1–38, 2021.

Shoujin Wang, Xiaofei Xu, Xiuzhen Zhang, Yan Wang, and Wenzhuo Song. Veracity-aware and event-driven personalized news recommendation for fake news mitigation. In *Proceedings of the ACM web conference 2022*, pp. 3673–3684, 2022.

Shoujin Wang, Wentao Wang, Xiuzhen Zhang, Yan Wang, Huan Liu, and Fang Chen. A hierarchical and disentangling interest learning framework for unbiased and true news recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3200–3211, 2024a.

Wenjie Wang, Yiyan Xu, Fuli Feng, Xinyu Lin, Xiangnan He, and Tat-Seng Chua. Diffusion recommender model. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 832–841, 2023b.

Yu Wang, Zhiwei Liu, Liangwei Yang, and Philip S Yu. Conditional denoising diffusion for sequential recommendation. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 156–169, 2024b.

Ting-Ruen Wei and Yi Fang. Diffusion models in recommendation systems: A survey. *arXiv preprint arXiv:2501.10548*, 2025.

Zihao Wu, Xin Wang, Hong Chen, Kaidong Li, Yi Han, Lifeng Sun, and Wenwu Zhu. Diff4rec: Sequential recommendation with curriculum-scheduled diffusion augmentation. In *Proceedings of the 31st ACM international conference on multimedia*, pp. 9329–9335, 2023.

Wenjia Xie, Rui Zhou, Hao Wang, Tingjia Shen, and Enhong Chen. Bridging user dynamics: Transforming sequential recommendations with schrödinger bridge and diffusion models. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pp. 2618–2628, 2024.

Xu Xie, Fei Sun, Zhaoyang Liu, Shiwen Wu, Jinyang Gao, Jiandong Zhang, Bolin Ding, and Bin Cui. Contrastive learning for sequential recommendation. In *IEEE International Conference on Data Mining*, pp. 1259–1273, 2022.

Zhengyi Yang, Xiangnan He, Jizhi Zhang, Jiancan Wu, Xin Xin, Jiawei Chen, and Xiang Wang. A generic learning framework for sequential recommendation with distribution shifts. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 331–340, 2023a.

Zhengyi Yang, Jiancan Wu, Zhicai Wang, Xiang Wang, Yancheng Yuan, and Xiangnan He. Generate what you prefer: Reshaping sequential recommendation via guided diffusion. In *Conference on Neural Information Processing Systems*, volume 36, 2023b.

Ghim-Eng Yap, Xiao-Li Li, and Philip S Yu. Effective next-items recommendation via personalized sequential pattern mining. In *International conference on database systems for advanced applications*, pp. 48–64. Springer, 2012.

Zhenrui Yue, Yueqi Wang, Zhankui He, Huimin Zeng, Julian McAuley, and Dong Wang. Linear recurrent units for sequential recommendation. In *ACM International Conference on Web Search and Data Mining*, pp. 930–938, 2024.

Jujia Zhao, Wang Wenjie, Yiyan Xu, Teng Sun, Fuli Feng, and Tat-Seng Chua. Denoising diffusion recommender model. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1370–1379, 2024.