AuriSRec: Adversarial User Intention Learning in Sequential Recommendation

Junjie Zhang[1,2], RuoBing Xie[3], Wenqi Sun[1,2], Leyu Lin[3], Wayne Xin Zhao[1,2]* and **Ji-Rong Wen[1,2]**,
[1]Gaoling School of Artificial Intelligence, Renmin University of China,
[2]Beijing Key Laboratory of Big Data Management and Analysis Methods.
[3]Tencent
junjie.zhang@ruc.edu.cn, batmanfly@gmail.com

Abstract

With recommender systems broadly deployed in various online platforms, many efforts have been devoted to learning user preferences and building effective sequential recommenders. However, existing work mainly focuses on capturing user *implicit* preferences from historical interactions and simply matching them with the next behavior, instead of predicting user *explicit intentions*. This may lead to inappropriate recommendations. In light of this issue, we propose the adversarial user intention learning approach for sequential recommendation, named **AuriSRec**. The major novelty of our approach is to explicitly predict user current intentions when making recommendations, by inferring their decision-making process as explained in target reviews (reviews written after interacting with the ground-truth item). Specifically, AuriSRec conducts adversarial learning between an intention generator and a discriminator. The generator predicts user intentions by taking their historical reviews and behavioral sequences as inputs, while target reviews provide guidance. Beyond typical sequential modeling methods in the field of natural language process (NLP), a decoupling-based review encoder and a hybrid attention fusion mechanism are introduced to filter noise and enhance the generation capacity. On the other hand, the discriminator determines whether the intention is generated or real based on their matching degree to the target item, thereby guiding the generator to produce gradually improved intentions. Extensive experiments on five datasets demonstrate the effectiveness of our approach.

1 Introduction

Nowadays, recommender systems have become increasingly prevalent in various online platforms. Since user behaviors are dynamically evolving over time, the task of sequential recommendation (SR) has received significant attention in the literature (Wang et al., 2019; Sun et al., 2019). Representative approaches include early Markov Chain methods (Rendle et al., 2010), and recent advances based on Transformer architecture (Kang and McAuley, 2018), which have greatly improved the performance of sequential recommendations.

In general, to interact with an item, a user is initially motivated by an intrinsic intention, which contains demands for certain types of items with personal reasons. They then follow the intentions to determine the behavior for subsequent interactions. Therefore, an ideal sequential recommender should make recommendations by explicitly inferring user current intentions. However, existing SR models (Kang and McAuley, 2018) mainly focus on capturing sequential patterns from user historical interactions and simply match them with the next interacted item, without analyzing user *explicit intention* underlying their interaction behaviors. We argue that this may lead to sub-optimal recommendations. While it is appealing to align recommendation models with the real decision-making process of users, there remains a lack of exploration in this area. One of the main challenges is the difficulty in gathering user explicit intention data for training purposes, as user may not actively disclose this personal information. Notably, many e-commerce platforms like Amazon and Yelp encourage users to write reviews on purchased products, explicitly explaining their experiences in interactions.

To make personalized recommendations, we aim to develop *explicit intention learning* in recommender systems, by employing advanced NLP techniques to analyze user personal information in reviews. Specifically, given a target item (*i.e.,* the ground-truth item to recommend), the associated review (called *target review*) contains explicit evidence of user current intentions, which can guide explicit intention learning. In addition, user historical reviews also reflect their evolving preferences, providing valuable information to predict user in-

* Corresponding author.

Findings of the Association for Computational Linguistics: EACL 2024, pages 12580–12592

tentions. Despite its promise, there are three main challenges. Firstly, predicting user fine-grained intentions is more complex than predicting potential items, posing a challenge to the model's generation capability. Secondly, to extract user explicit intention from reviews, existing methods often directly encode review sequences using universal pretrained language models (PLMs) like BERT (Devlin et al., 2019). However, we argue that reviews contain complex information, including personalized user *preferences* [1], general item *characteristics*, and even noise data (Chen et al., 2018). A sophisticated review encoder is needed to purify these semantics in reviews for effective user intention learning. Lastly, both user reviews and behavioral sequences are crucial for intention prediction, but their heterogeneous nature makes it challenging to integrate them directly.

To this end, in this paper, we present the proposed adversarial user intention learning approach for sequential recommendation, named **AuriSRec**. Our approach takes user historical reviews and behavioral sequences as input, and leverages target reviews as guidance, aiming to explicitly infer user intentions via adversarial learning. To achieve this, we focus on two key points: (1) integrating user historical reviews and behavioral sequences for intention prediction, and (2) aligning the predicted intention with user real intention. For the first point, we design an intention generator by introducing a hybrid attention fusion mechanism. Specifically, instead of a universal language model, we equip the generator with a decoupling-based review encoder, to purify user preferences and item characteristics from reviews. Then, the generator employs purified user preferences as prompts to enhance behavior sequence encoding and infers preference-enhanced intentions. For the second point, we develop an adversarial learning framework, where user intentions derived from target reviews are treated as real intentions. A discriminator is introduced to evaluate the match between intentions (either real or generated) and target items, guiding the generator to improve intention predictions and recommendations.

To our knowledge, this is the first work that explicitly predicts user intention for next interaction. To evaluate the proposed approach, we conduct extensive experiments on five real-world recommendation datasets. The results demonstrate that AuriSRec outperforms several competing baselines.

[1]Here, we define **preference** to distinguish it from **intention**. User preferences are learned from *historical* reviews and reflect their *intrinsic* tastes. While intentions are the motivations behind decisions with *target* items and more immediate.

2 Related Work

Sequential Recommendation. This field predicts potential items by capturing user preferences from their behavioral sequences. Deep neural networks like Transformers have been introduced to enhance the modeling capacity (Kang and McAuley, 2018). Recent focus (Wang et al., 2022; Zhang et al., 2023a,b; Hou et al., 2023) includes incorporating modality features (*e.g.,* text and images) to build transferable and universal recommendation models. However, existing methods mainly capture user implicit preferences from their historical interactions without explicitly analyzing underlying intentions, leading to sub-optimal performance.

Intention Learning for Recommendation. This field focuses on inferring user intentions to improve recommendations (Chen et al., 2022; Cai et al., 2021). Most existing studies predict user intentions in the latent space. Especially, DSSRec (Ma et al., 2020) introduces an intent variable to capture mutual information between a user's historical interactions and multiple future behaviors. ICLRec (Chen et al., 2022) clusters user behavior representations and uses the centroids as intention representations. ICSRec (Qin et al., 2023) splits user behavioral sequences and applies intention learning among the sub-sequences. Nevertheless, these studies may introduce biases, as the captured intention is simulated by models implicitly rather than exposed by users.

Review-based Recommendation. As traditional collaborative filtering methods struggle with sparse data (He et al., 2020), researchers have explored incorporating reviews to improve modeling (Zheng et al., 2017; Chin et al., 2018). For sequential recommendation, RNS (Li et al., 2019) integrates user historical reviews and behavioral sequences to facilitate recommendations. However, this approach employs a universal review encoder to encode reviews, disregarding their coupled user preferences and item characteristics. Moreover, it merely depends on historical reviews to capture user preferences, neglecting user intention expressed in target reviews. In contrast, our work purifies semantics in reviews and explicitly predicts user current intentions, guided by real intentions expressed in target reviews through adversarial learning.

3 Problem Definition

In this section, we introduce the notations and define the task. Formally, given a set of users $\mathcal{U}$ and a set of items $\mathcal{I}$ in a recommender system, we can obtain the interaction records for each user $u \in \mathcal{U}$, and organize them into a chronologically ordered item sequence $S_u = \{i_1, i_2, \cdots, i_n\}$ with length n. Each interaction likely has a corresponding review r written by user u for item i (we also consider cases where reviews are only written after a few interactions). We organize user u's historical reviews into a chronological sequence $R_u = r_1, r_2, \ldots, r_n$. Especially, we propose inferring user potential intentions by jointly modeling user reviews and historical interaction sequences. To improve their alignments, instead of using item IDs, we acquire item textual representations by employing BERT (Devlin et al., 2019) to encode item description text (*e.g.,* title, category, and brand), following a similar approach to (Hou et al., 2022):

$$\boldsymbol{x}_i = \text{BERT}([\texttt{[CLS]}; w_1, \ldots, w_m]), \quad (1)$$

Here we define the task of sequential recommendation. Given historical reviews R_u and behavioral sequence S_u of user u, we aim to recommend a top-K ranking list of items as the potential next items by predicting $p(i_{n+1}|R_u, S_u)$. Since the prediction is performed for the $(n+1)$-th step, we refer to the ground-truth item i_{n+1} and its associated review r_{n+1} as *target item* and *target review*, respectively.

4 Methodology

In this section, we present **AuriSRec**, an approach for explicit intention learning. We first describe our overall framework, then discuss two key parts (*i.e.,* review decoupling and intention generation), and finally present the optimization process. Figure 1 illustrates the overall architecture.

4.1 Overall Framework

Our model focuses on explicitly forecasting user intentions when making recommendations, by aligning with real user decision-making processes as expressed in target reviews. To achieve this, we develop an adversarial learning framework between an *intention generator* G and a *discriminator* D. Guided by target reviews, the generator takes both user historical reviews and behavioral sequences as input, and predicts potential intentions as follows:

$$\boldsymbol{e}^g = G\big(\underbrace{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n}_{\text{past preferences}}, \underbrace{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n}_{\text{interaction}}\big), \quad (2)$$

where $\boldsymbol{e}^g$ is the predicted intention, $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n$ are user preferences learned from historical reviews and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are user behavioral sequence.

In the proposed framework, our approach introduces three main technical advancements. Specifically, to purify intricate semantics coupled in reviews, we design a dual-perspective contrastive learning task, thereby pre-training a decoupling-based review encoder (Section 4.2). To integrate the extracted user preferences with behavioral sequences for intention predictions, we propose a hybrid attention fusion mechanism, where user preferences act as prompts to guide the encoding of behavioral sequences (Section 4.3). To align the predicted intentions with user real intentions, we develop an adversarial learning approach. In this process, we consider intentions expressed in user target reviews as real intentions. A discriminator is designed to discriminate whether the input intention is generated or real according to their matching degree to the target item. Through a minimax game, the discriminator guides our model (*i.e.,* generator) to match the distribution of real user intention (Section 4.4). Finally, we leverage the inferred user intention to predict the next item as follows:

$$P\left(i_{n+1} \mid S_u, R_u\right) = \text{Softmax}\left(\boldsymbol{e}^g \cdot \boldsymbol{x}_{i_{n+1}}\right). \quad (3)$$

In what follows, we provide a detailed introduction.

4.2 User Review Decoupling

To infer a user's current intention, it is crucial to consider insights from their historical reviews. However, since user reviews are freely written, they blend personalized *user preferences*, and general *item characteristics*. This poses challenges in extracting personal semantics within reviews. Unlike existing NLP methods that use a universal language model, we design a decoupling-based review encoder pre-trained by a dual-perspective contrastive task, aiming to purify coupled semantics.

4.2.1 Decoupling-based Review Encoder.

In addition to taking PLMs as the review encoder to encode the universal semantics, we integrate a *user head* and an *item head* on its basis to purify user preferences and item characteristics from reviews, respectively. Both the user head and item head are lightweight transformer encoders. Formally, given the review r_j and its words $\{w_1, w_2, \cdots, w_l\}$, similar to the textual encoding technique of item texts (*i.e.,* $\boldsymbol{x}_i$ in Eq. (1)), we first employ BERT model to extract the universal semantics $\boldsymbol{H}_j$ in review. We

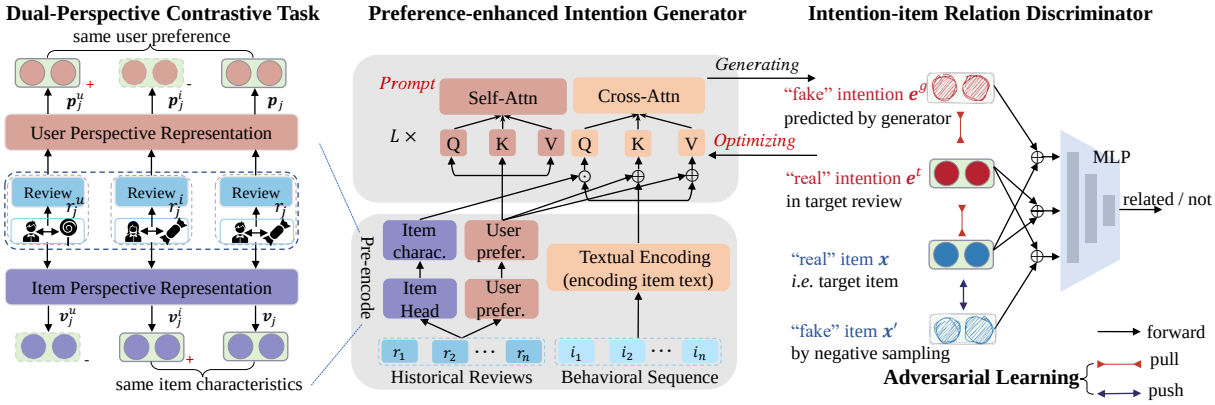

Figure 1: The overall framework of AuriSRec. It includes an adversarial learning framework and essential components for explicit intention predictions and recommendations.

then input $\boldsymbol{H}_j$ into the user head and item head to acquire the representations of user preference $\boldsymbol{p}_j$ and item characteristics $\boldsymbol{v}_j$ as follows:

$$\boldsymbol{p}_j = \text{User-head}(\boldsymbol{H}_j \cdot \boldsymbol{W}_p + \boldsymbol{b}_p), \tag{4}$$
$$\boldsymbol{v}_j = \text{Item-head}(\boldsymbol{H}_j \cdot \boldsymbol{W}_v + \boldsymbol{b}_v). \tag{5}$$

where $\boldsymbol{p}_j, \boldsymbol{v}_j \in \mathbb{R}^{d_V}$ are the sum of last layer hidden states of the input.

4.2.2 Dual-Perspective Contrastive Task.

Given the decoupling-based review encoder mentioned above, we pre-train the user and item heads to focus on user preferences and item characteristics in reviews, respectively. Since there are no explicit labels for these different semantics, we aim to achieve this purpose through unsupervised learning. Notably, reviews by the same user may express their personal preferences, including intrinsic factors unrelated to the item being commented. Similarly, reviews for the same item may highlight consistent item features like color and price. With this in mind, our idea is to encourage the user head to extract similar preferences from the reviews written by the same user, while simultaneously enforcing the item head to extract similar item characteristics from the reviews about the same item.

To implement this, we design a dual-perspective contrastive learning task, where positive and negative samples are selected in dual perspectives. Specifically, on one hand, reviews by the same user, reflecting similar preferences, are positive samples to be pulled together from a user perspective. Conversely, since these reviews are for different items, they contain distinct item characteristics and should serve as negative samples to be pushed away from an item perspective. Therefore, even the same review acts as opposite polarities under different perspectives, enabling the encoder to decouple user preferences and item characteristics. Formally, for a review r_j written by a user u to an item i, we sample r_j^u from reviews written by the same user but for distinct items, and r_j^i from reviews received by the same item but from different users. We first conduct a user perspective contrastive learning and use the user head to capture the representations of user preferences (*e.g.*, $\boldsymbol{p}_j$ in Eq. (4)). Then we pull together $\langle \boldsymbol{p}_j, \boldsymbol{p}_j^u \rangle$, which reflects similar user preferences, while pushing apart $\langle \boldsymbol{p}_j, \boldsymbol{p}_j^i \rangle$ and other in-batch negatives as follows:

$$\ell_{\text{user}} = -\sum_{j=1}^{B} \log \frac{\exp\left(\boldsymbol{p}_j \cdot \boldsymbol{p}_j^u / \tau\right)}{\sum_{j'=1}^{B} \left[\exp\left(\boldsymbol{p}_j \cdot \boldsymbol{p}_{j'}^u / \tau\right) + \exp\left(\boldsymbol{p}_j \cdot \boldsymbol{p}_{j'}^i / \tau\right)\right]}, \tag{6}$$

where preferences of different users $\boldsymbol{p}_{j'}^u$ (in-batch negatives) and $\boldsymbol{p}_{j'}^i$ (other users who interacted with the same item) are regarded as negative samples. The item perspective contrastive loss is symmetric:

$$\ell_{\text{item}} = -\sum_{j=1}^{B} \log \frac{\exp\left(\boldsymbol{v}_j \cdot \boldsymbol{v}_j^i / \tau\right)}{\sum_{j'=1}^{B} \left[\exp\left(\boldsymbol{v}_j \cdot \boldsymbol{v}_{j'}^i / \tau\right) + \exp\left(\boldsymbol{v}_j \cdot \boldsymbol{v}_{j'}^u / \tau\right)\right]}, \tag{7}$$

The overall loss of dual-perspective contrastive learning can be formalized by combining the two perspective losses mentioned above as follows:

$$\mathcal{L}_{\text{pre-train}} = \ell_{\text{user}} + \ell_{\text{item}}. \tag{8}$$

Overall, the proposed decoupling-based review encoder can outperform universal encoders (*e.g.*, BERT), by decoupling user preferences and item

characteristics within reviews respectively. To enhance the efficiency, we pre-encode and cache each review, allowing extracted representations to be directly used in downstream modules. This significantly reduces inference latency. We showcase the intricate semantics in reviews and demonstrate the efficacy of our review encoder in Appendix D.

4.3 Preference-enhanced Intention Generator

Predicting user intentions is more challenging than predicting the next interaction, due to its fine-grained nature. Therefore, we aim to develop a preference-enhanced intention generator G_θ that considers both user historical reviews and behavioral sequences. Especially, we build the intention generator by extending the widely-used Transformer architecture with a hybrid attention fusion mechanism, to integrate these intricate signals.

4.3.1 Hybrid Attention Fusion Mechanism.

We employ user behavioral sequences and historical reviews to predict their intentions. As mentioned in Section 4.2, user review sequences can be decoupled into user preference sequences and item characteristic sequences, by Eq. (4) and Eq. (5). User behavioral sequences can be encoded to their textual representation, using Eq. (1). To integrate these signals, we consider user preferences as the *personalized prompts* in modeling user behavioral sequences, and employ item characteristics to bridge the semantic gap between them.

Specifically, given the user preference sequence and behavior sequence, we first concatenate these two sequences and input them to the generator with different attention mechanisms. For user preferences, since they reflect user intrinsic tastes, we employ bidirectional attention to capture their global preference features. For behavioral sequences, to capture the evolving sequential patterns, we adopt the left-to-right attention encoding. Notably, since user behaviors are driven by their relevant intrinsic preferences, we extend the attention context in behavior modeling by incorporating both the leftward context of the behavioral sequence and the user preferences through cross-attention. This allows user preferences to act as *personalized prompts* (Liu et al., 2023), guiding the behavioral sequence encoding and enhancing intention generation. Additionally, we use item characteristic sequences to enhance the *query* in cross-attention between user preferences and behavioral sequences, to bridge the gap between them. This is because item characteristics extracted from reviews correlate with user preferences (decoupled from the same review) and align with behavioral sequences (reflecting general item characteristics). The intention generator can be formalized as follows:

$$\boldsymbol{F}^0 = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \tag{9}$$

$$\boldsymbol{P}^{l+1} = \mathrm{FFN}(\text{Self-Attn}(\boldsymbol{P}^l, \boldsymbol{P}^l, \boldsymbol{P}^l)), \tag{10}$$

$$\boldsymbol{F}^{l+1} = \mathrm{FFN}(\text{Cross-Attn}([\boldsymbol{V}+\boldsymbol{F}^h],[\boldsymbol{P}^l;\boldsymbol{F}^h],[\boldsymbol{P}^l;\boldsymbol{F}^h])), \tag{11}$$

where $\boldsymbol{F}^l = [\boldsymbol{f}_0^l; \ldots; \boldsymbol{f}_n^l]$ denoted hidden representations of behavioral sequence in the l-th layer, $\boldsymbol{P}^l = [\boldsymbol{p}_0^l; \ldots; \boldsymbol{p}_n^l]$ denotes user preferences and $\boldsymbol{V} = [\boldsymbol{v}_0; \ldots; \boldsymbol{v}_n]$ are the item characteristics extracted from historical reviews. To obtain the output of the intention generator (Eq. (2)), we take the final hidden vector of the extended sequence $\boldsymbol{f}_n^L$ as the generated user intention, denoted as $\boldsymbol{e}^g$.

4.4 Intention Learning and Recommendation

To improve generation effectiveness, we employ an adversarial learning framework. This involves aligning the generator's estimated intentions with real intentions expressed in target reviews, based on the guidance of discriminator. In what follows, we introduce the discriminator architecture, summarize the adversarial learning process, and present recommendation optimization and inference.

4.4.1 Intention-Item Relation Discriminator.

As previously stated, user next behavior is driven by their current intentions. We hypothesize that a close connection exists between a perfectly generated intention and the target item. Therefore, unlike other adversarial learning methods (Goodfellow et al., 2020) that directly distinguish the generated data from the real one, we introduce an *intention-item relation discriminator* D (parameterized by ϕ). It captures the consistency between user intentions and target items, evaluating their compatibility as a matched pair. Specifically, given the intention representation $\boldsymbol{e}$ (Eq. (2)) and item textual representation $\boldsymbol{x}$ (Eq. (1)), we discriminate their matching relation as:

$$D_\phi(\boldsymbol{e}, \boldsymbol{x}) = \mathrm{Sigmoid}(\mathrm{MLP}(\boldsymbol{e} \oplus \boldsymbol{x})), \tag{12}$$

where $\oplus$ denotes concatenation, and MLP is a 3-layer neural network.

4.4.2 Adversarial Learning.

For adversarial learning, given a target item, we consider the intention predicted by the generator as a *mismatched* representation (*i.e., fake intention*)

and the intention learned from the target review as a *matched* representation (*i.e., real intention*). This is because the target review contains explicit explanations from users about their interaction decisions, making it a surrogate of real intention. Especially, to obtain real intention representation $\boldsymbol{e}^t$, we follow the similar generation process as depicted in Eq. (2), while taking user target reviews and historical interactions as input:

$$\boldsymbol{e}^t = \mathrm{G}(\underline{\boldsymbol{H}_{n+1}}; \boldsymbol{x}_1, \cdots, \boldsymbol{x}_n), \tag{13}$$

where $\boldsymbol{H}_{n+1}$ is the encoded target review. The consistent fake and real intention generation process empirically improves the training stability.

In addition to focusing on the intention side, we also conduct adversarial learning on the item side, by sampling negative items and discriminating their matching degree to real intention. Overall, through iterative minimax optimization, the discriminator guides the intention generator to produce intentions that better match real user intentions and are more consistent with target items. Formally, the adversarial learning process is as follows:

$$\begin{aligned}\ell_{\mathrm{adl}}(\theta,\phi) = \min_{\theta}\max_{\phi}\sum_{j=1}^{B}(\\ E_{\boldsymbol{e}^t=G(\boldsymbol{H}_{n+1};\boldsymbol{S}_u),\boldsymbol{x}_j\sim p_{pos}}[\log(D_\phi(\boldsymbol{e}^t,\boldsymbol{x}_j))] \\ +E_{\boldsymbol{e}^t=G(\boldsymbol{H}_{n+1};\boldsymbol{S}_u),\boldsymbol{x}'_j\sim p_{neg}}[1-\log(D_\phi(\boldsymbol{e}^t,\boldsymbol{x}'_j))] \\ +E_{\boldsymbol{e}^g=G(\boldsymbol{P};\boldsymbol{S}_u),\boldsymbol{x}_j\sim p_{pos}}[1-\log(D_\phi(\boldsymbol{e}^g,\boldsymbol{x}_j))]),\end{aligned} \tag{14}$$

where $\boldsymbol{e}^t$ and $\boldsymbol{e}^g$ indicate real and fake intention learned from target review (Eq. (13)) and generated by Eq. (2), respectively. $\boldsymbol{x}_j \sim p_{pos}$ and $\boldsymbol{x}'_j \sim p_{neg}$ denote target item and fake items obtained by in-batch negative sampling, respectively.

4.4.3 Recommendation optimization and Inference.

Via adversarial learning, our model (*i.e.,* the generator) can effectively predict the current intention of a user for the next interaction. To further improve the recommendation efficacy, we optimize the widely used cross-entropy loss as follows:

$$\ell_{\mathrm{rec}}(\theta) = -\sum_{j=1}^{|\mathcal{I}|}\log\frac{\exp\left(\boldsymbol{e}^g_j\cdot\boldsymbol{x}_j/\tau\right)}{\sum_{j'=1}^{|\mathcal{I}|}\exp\left(\boldsymbol{e}^g_j\cdot\boldsymbol{x}'_j/\tau\right)}, \tag{15}$$

where $\boldsymbol{e}^g$ is the predicted intention by our generator. In general, our approach is trained by jointly optimizing the adversarial loss in Eq. (14) and recommendation loss in Eq. (15), with a weight ratio of $1:1$. During inference, we make recommendations with the predicted intention by Eq. (3). We analyze the overall complexity in Section 4.5.2.

4.5 Discussion

4.5.1 Comparison with existing methods

Here, we discuss the relationships between our proposed InstructRec and other methods.

(1) *Traditional sequential recommendation models* (Kang and McAuley, 2018; Sun et al., 2019) typically rely on user historical interactions to capture implicit preferences. However, these implicit patterns may not align with real user intentions, leading to suboptimal recommendations. Our approach aims to explicitly predict user current intentions, by aligning with their real decision-making process expressed in target reviews.

(2) *Intention-enhanced methods* (Chen et al., 2022; Qin et al., 2023; Tanjim et al., 2020; Cai et al., 2021) focus on inferring user intentions when making recommendations, by employing user behavior types and item category information. Although effective, we argue that this data is coarse-grained and cannot fully reflect user fine-grained needs. In contrast, our method derives explicit user intentions from personalized reviews, which provide clear evidence of user intentions.

(3) *Universal sequence representation methods* like UniSRec (Hou et al., 2022) involve learning item textual representation using PLMs. However, these methods heavily focus on modeling modal features. In contrast, our proposal integrates semantic information from reviews and user behaviors through a hybrid-attention fusion mechanism, balancing content and collaborative-based modeling.

(4) *Review-based sequential recommenders* such as RNS (Li et al., 2019) encodes historical reviews using a unified review encoder. However, they fail to consider the distinct semantics related to users and items within reviews, resulting in less representative embeddings. Moreover, none of these methods explore using user target reviews to predict their current intentions. Our work introduces a decoupling-based review encoder to purify intricate semantics in reviews, and employ target reviews as guidance for user intention prediction.

(5) *Adversarial learning approaches* (Xie et al., 2021; Zhao et al., 2020) employ adversarial learning to predict next items. In contrast, Our method focuses on capturing fine-grained intention distribution, making more suitable recommendations. We also introduce user and item two-sided adversarial learning to capture user-item relations better.

4.5.2 Complexity analysis

In this part, we analyze the complexity of our proposed InstructRec. Specifically, for dual-perspective contrastive learning, it takes a time of $\mathcal{O}(BKM^2D)$, where K is the number of layers of review encoder, M is the average length of reviews and D is the dimension of hidden representation. Due to the high cost, we pre-encode reviews and cache the results. For the generator, since it takes both historical reviews and behavioral sequence as input, the time complexity is $\mathcal{O}(2BLN^2D)$, where L is the number of layers of the generator and N is the average length of behavioral sequences. For the discriminator, since we conduct adversarial learning on both the intention and item side, the time complexity is $\mathcal{O}((2+B)BD)$. Thus, the computation cost of adversarial learning is dominated by the generator, which is twice as complex as the classic Transformer-based SASRec (Kang and McAuley, 2018). For inference, the target review and discriminator are discarded. After generating the predicted intention e^g, the complexity is the same as standard MF methods for evaluating the candidates ($\mathcal{O}(D|\mathcal{I}|)$).

5 Experiments

5.1 Experimental Setup

5.1.1 Datasets.

We experiments with five categories from Amazon review dataset (Ni et al., 2019): "*Industrial and Scientific*", "*Prime Pantry*", "*Musical Instruments*", "*Arts, Crafts and Sewing*" and "*Office Products*". Interactions are grouped by users and sorted by timestamp. Dataset statistics are in Table 5.

5.1.2 Approaches for Comparison.

We consider the following four types of baseline methods for performance comparison: (1) Traditional sequential recommendation methods: SASRec (Kang and McAuley, 2018) and BERT4Rec (Sun et al., 2019); (2) Intention-enhanced methods: SSE-PT (Wu et al., 2020) and ICLRec (Chen et al., 2022); (3) Universal text-enhanced methods: S^3-Rec (Zhou et al., 2020), ZESRec (Ding et al., 2021), and UniSRec (Hou et al., 2022); (4) Review-enhanced methods: RNS (Li et al., 2019). We give a detailed description of each baseline in Appendix B.

5.1.3 Implementation Details.

Following prior work (Hou et al., 2022), we set the maximum user behavioral sequence length to 50. We optimize all the baselines by searching the hyper-parameters. For AuriSRec, we pre-train the review encoder for 100 epochs. Using Adam optimizer with a batch size of 2,048, we tune the learning rate in {0.0007, 0.001, 0.007, 0.01}. We evaluate the next-item recommendation performance using hit ratio (HR) and normalized discounted cumulative gain (NDCG) metrics.

5.2 Overall Performance

We compare the performance of different methods in Table 1. In general, AuriSRec outperforms the baselines on nearly all datasets.

Traditional sequential recommenders (*e.g.,* SASRec and BERT4Rec) do not perform well. This indicates that capturing user implicit preferences from historical interactions and matching them with target items is sub-optimal. Although intention-enhanced methods (*e.g.,* SSE-PT and ICLRec) improve these methods by introducing user intention-related variables, these implicit representation struggles to infer fine-grained intentions. Our method employs reviews for explicit intention learning, leading to better performance.

Text-based methods (*e.g.,* S^3-Rec, ZESRec, and UniSRes) outperform traditional recommenders, indicating the remarkable language modeling capability of PLMs. However, their performance heavily relies on modal features. In contrast, our approach not only captures user preferences from reviews using PLMs but also employs them as prompts to guide behavioral sequence encoding. This integration of semantic content and collaborative information improves upon these methods.

For review-based methods, RNS performs poorly. This may be because it employs a universal review encoder and struggles to discriminate intricate information in reviews, leading to sub-optimal representations. Additionally, RNS ignores target reviews, missing out on valuable insights into current user needs. In contrast, our approach designs a decoupling-based review encoder to purify intricate semantics in reviews. We employ adversarial learning to explicitly predict user intentions based on target reviews, providing more personalized recommendations. Notably, users in real-world systems may not write reviews after interactions, making it challenging to analyze their intentions. We pro-

Table 1: Overall performance comparison. Best and second-best methods marked in bold and underlined. "Improv." denotes the improvement over the best baseline. "*" denotes significant improvements (t-test with $p < 0.05$).

Dataset	Metric	SASRec	BERT4Rec	SSE-PT	ICLRec	S^3-Rec	ZESRec	UniSRec	RNS	AuriSRec	Improv.
Scientific	HR@5	0.0759	0.0315	0.0776	0.0742	0.0734	0.0770	0.0767	0.0331	**0.0834***	+7.47%
	NDCG@5	0.0471	0.0190	0.0470	0.0519	0.0483	0.0445	0.0496	0.0212	**0.0533***	+2.70%
	HR@10	0.1020	0.0521	0.1041	0.1027	0.0999	0.1039	0.1138	0.0723	**0.1198***	+5.27%
	NDCG@10	0.0555	0.0257	0.0555	0.0621	0.0568	0.0556	0.0593	0.0376	**0.0649***	+4.51%
Prime	HR@5	0.0283	0.0174	0.0252	0.0158	0.0279	0.0251	0.0327	0.0273	**0.0376***	+14.98%
	NDCG@5	0.0150	0.0103	0.0132	0.0095	0.0147	0.0152	0.0206	0.0163	**0.0224***	+8.74%
	HR@10	0.0482	0.0287	0.0450	0.0256	0.0462	0.0395	0.0563	0.0503	**0.0640***	+13.68%
	NDCG@10	0.0214	0.0139	0.0196	0.0126	0.0206	0.0198	0.0282	0.0236	**0.0310***	+9.93%
Instruments	HR@5	0.0810	0.0602	0.0830	0.0816	0.0803	0.0703	**0.0935**	0.0642	0.0903	–
	NDCG@5	0.0537	0.0394	0.0528	0.0624	0.0540	0.0473	0.0616	0.0461	**0.0645***	+3.37%
	HR@10	0.1102	0.0783	0.1088	0.1083	0.1039	0.0909	0.1112	0.0926	**0.1167***	+5.27%
	NDCG@10	0.0621	0.0452	0.0611	0.0711	0.0616	0.0539	0.0709	0.0553	**0.0751***	+5.63%
Arts	HR@5	0.0802	0.0692	0.0784	0.0748	0.0820	0.0593	0.0789	0.0704	**0.0839***	+2.32%
	NDCG@5	0.0492	0.0308	0.0490	0.0554	0.0509	0.0381	0.0520	0.0422	**0.0580***	+4.69%
	HR@10	0.1070	0.0715	0.1046	0.0974	0.1078	0.0798	0.1089	0.0820	**0.1131***	+3.86%
	NDCG@10	0.0578	0.0382	0.0574	0.0646	0.0592	0.0447	0.0616	0.0496	**0.0674***	+4.33%
Office	HR@5	0.0850	0.0560	0.0866	0.0801	0.0827	0.0591	0.0844	0.0569	**0.0920***	+6.24%
	NDCG@5	0.0587	0.0378	0.0581	**0.0682**	0.0592	0.0406	0.0599	0.0461	0.0671	–
	HR@10	0.1090	0.0736	0.1100	0.0948	0.1085	0.0736	0.1059	0.0926	**0.1152***	+4.73%
	NDCG@10	0.0652	0.0435	0.0645	0.0719	0.0655	0.0452	0.0668	0.0553	**0.0746***	+3.76%

Table 2: Ablation analysis of AuriSRec and its variants on "Scientific" and "Prime" datasets. "AL." is the abbreviation of "Adversarial Learning".

Variants	Scientific		Prime	
	HR@10	NDCG@10	HR@10	NDCG@10
AuriSRec	**0.1198**	**0.0649**	**0.0640**	**0.0310**
w/o User Taste	0.1117	0.0603	0.0587	0.0288
w/o Decoupling	0.1132	0.0619	0.0599	0.0291
w/o AL.	0.1102	0.0586	0.0532	0.0274
Directly Fitting	0.1164	0.0624	0.0614	0.0288
w/o Item-side AL,	0.1145	0.0614	0.0618	0.0300

pose several methods to solve this and evaluate their performance in Appendix C.

5.3 Further Analysis

5.3.1 Ablation Study.

Here, we assess the impact of each proposed component on final performance. As shown in Table 2, removing any component degrades performance:

(1) w/o *User Preference*: In this variant, the generator only models behavioral sequences to predict intentions. The performance drop indicates that user preferences in historical reviews provide valuable insights for predicting intentions and guiding user behaviors encoding as personalized prompts.

(2) w/o *Decoupling*: Instead of decoupling semantics in reviews, we directly use universal BERT to encode reviews. The performance gap suggests that reviews have complex semantics, which can be clarified with our decoupling-based encoder.

The above two experiments indicate that our method can integrate signals from reviews and behavioral sequences to improve recommendations.

(4) w/o *Adversarial Learning*: Without adversarial learning (Eq. (14)), this variant only optimizes the model with the recommendation loss (Eq. (15)), degrading to a prompt-enhanced sequential model that lacks guidance from target reviews for intention learning. The result indicates the efficacy of explicitly predicting user intentions.

(5) *Directly Fitting*: Here, we directly optimize the generator to fit user intentions in target reviews, without adversarial learning. However, this variant shows a significant performance drop and overfitting, highlighting the effect of adversarial learning in capturing real intention distribution.

(6) w/o *Item-side Adversary*: In this variant, adversarial learning is only conducted on the intention side. The result shows that sampling negative items for adversarial learning helps the discriminator capture the relations between user intentions and items, enhancing the learning of the generator.

The above three variants demonstrate the effectiveness of adversarial learning in enabling the generator to infer user current intentions.

5.3.2 Effect of Target Reviews.

We employ target reviews to guide intention learning. Here, we explore this by considering two questions: *(1) what information is in target reviews*, and *(2) what can our model learn from target reviews*.

For the first question, we test the result of directly employing semantics extracted from target reviews for recommendations. The upper part of Table 3 shows that semantics extracted from target reviews with both user head and item head can make satisfying recommendations. This confirms

Table 3: Performance analysis w.r.t. encoding of target reviews. "BERT", "User Head" and "Item Head" mean different review encoders to encode target review.

Target Review Encoder	Scientific		Prime	
	HR@10	NDCG@10	HR@10	NDCG@10
Employ semantic of target review for recommendation				
BERT	0.5294	0.3620	0.6635	0.5168
User Head	0.1421	0.0802	0.3051	0.2041
Item Head	0.8717	0.6700	0.8949	0.7248
Employ semantic of target review as adversarial learning guidance				
BERT (Our method)	**0.1198**	**0.0649**	**0.0640**	**0.0310**
User Head	0.1168	0.0645	0.0618	0.0310
Item Head	0.1162	0.0627	0.0608	0.0294

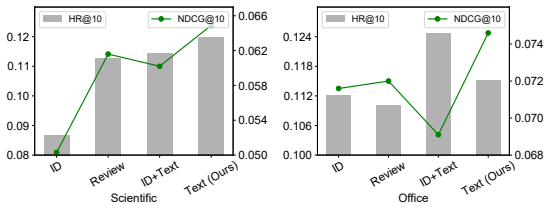

Figure 2: Performance comparison w.r.t. item representations on "Scientific" and "Office".

that target reviews contain specific information related to user preferences and target item characteristics, thereby demonstrating real user intentions. For the second question, we evaluate how different semantics from target reviews guide the generator's training. The bottom part of Table 3 reveals that using universal semantics from target reviews as guidance yields the best performance. It helps the generator grasp user preferences and aligns generated intentions with item characteristics. Overall, target reviews are essential for intention learning.

5.3.3 Effect of Item Representations.

Our approach aligns the semantic space of user historical reviews and behavioral sequences by learning textual item representations (Eq. (1)). Here, we explore the effects of different item representation methods on recommendations. Figure 2 shows that encoding item descriptions (title, category, and brand) performs best, indicating that the textual representation approach helps to integrate diverse input signals. In the "Office" dataset, incorporating ID embeddings with textual representation improves HR@10 but reduces NDCG@10. This may be due to the large dataset containing many items with similar descriptions, limiting the discriminative of textual encoding. However, coarse-grained ID representations compromise ranking metrics that emphasize detailed comparisons. Additionally, representing items using reviews causes overfitting, suggesting reviews lack distinctive item information. Therefore, we should choose representation methods based on specific requirements.

6 Conclusion and Future Work

In this paper, we introduce *AuriSRec*, a method for explicit user intentions learning in sequential recommendations. Our approach conducts adversarial learning between an intention generator and a discriminator, with user historical reviews and behavioral sequences as inputs, and target reviews as guidance. Unlike previous work that takes universal language models for review encoding, we design a decoupling-based review encoder to purify user preferences and item characteristics from historical reviews. The extracted preferences serve as personalized prompts to guide the encoding of user behavioral sequences, thereby generating preference-enhanced intentions. Experimental results demonstrate the superiority of our approach. Future work will involve using additional data types that reflect user intentions (*e.g.,* ratings), to improve intention predictions.

7 Limitations

Our study focuses on learning user explicit intentions for sequential recommendations, by employing their personalized reviews. However, it still has several limitations. First, our work only utilizes the valuable information in reviews, without including other data types that can indicate user intentions, such as ratings and clicked images. Second, we fine-tune a review encoder based on the BERT model. We do not use powerful large language models, such as LLaMA (Touvron et al., 2023), as review encoders due to efficiency considerations.

Acknowledgement

This work was partially supported by National Natural Science Foundation of China under Grant No. 62222215, Beijing Natural Science Foundation under Grant No. L233008 and 4222027, Young Elite Scientists Sponsorship Program by CAST under Grant No. 2023QNRC001. Xin Zhao is the corresponding author.

References

Renqin Cai, Jibang Wu, Aidan San, Chong Wang, and Hongning Wang. 2021. Category-aware collaborative sequential recommendation. In *SIGIR '21: The*

44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021, pages 388–397. ACM.

Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. 2018. Neural attentional rating regression with review-level explanations. In *WWW*.

Yongjun Chen, Zhiwei Liu, Jia Li, Julian McAuley, and Caiming Xiong. 2022. Intent contrastive learning for sequential recommendation. In *WWW*.

Jin Yao Chin, Kaiqi Zhao, Shafiq Joty, and Gao Cong. 2018. Anr: Aspect-based neural recommender. In *CIKM*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Hao Ding, Yifei Ma, Anoop Deoras, Yuyang Wang, and Hao Wang. 2021. Zero-shot recommender systems. *arxiv*.

Wenqi Fan, Zihuai Zhao, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang, Jiliang Tang, and Qing Li. 2023. Recommender systems in the era of large language models (llms). *CoRR*.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Communications of the ACM*.

Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yong-Dong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *SIGIR*.

Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. 2022. Towards universal sequence representation learning for recommender systems. In *KDD*.

Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. 2023. Large language models are zero-shot rankers for recommender systems. *arXiv preprint arXiv:2305.08845*.

Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *ICDM*.

Chenliang Li, Xichuan Niu, Xiangyang Luo, Zhenzhong Chen, and Cong Quan. 2019. A review-driven neural model for sequential recommendation. In *IJCAI*.

Jianghao Lin, Xinyi Dai, Yunjia Xi, Weiwen Liu, Bo Chen, Xiangyang Li, Chenxu Zhu, Huifeng Guo, Yong Yu, Ruiming Tang, and Weinan Zhang. 2023. How can recommender systems benefit from large language models: A survey. *CoRR*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Jianxin Ma, Chang Zhou, Hongxia Yang, Peng Cui, Xin Wang, and Wenwu Zhu. 2020. Disentangled self-supervision in sequential recommenders. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 483–491. ACM.

Jianmo Ni, Jiacheng Li, and Julian J. McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *EMNLP-IJCNLP*, pages 188–197.

Xiuyuan Qin, Huanhuan Yuan, Pengpeng Zhao, Guanfeng Liu, Fuzhen Zhuang, and Victor S. Sheng. 2023. Intent contrastive learning with cross subsequences for sequential recommendation. *CoRR*.

Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized markov chains for next-basket recommendation. In *WWW*.

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*.

Md. Mehrab Tanjim, Congzhe Su, Ethan Benjamin, Diane Hu, Liangjie Hong, and Julian J. McAuley. 2020. Attentive sequential models of latent intent for next item recommendation. In *WWW*, pages 2528–2534. ACM / IW3C2.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Jie Wang, Fajie Yuan, Mingyue Cheng, Joemon M Jose, Chenyun Yu, Beibei Kong, Zhijin Wang, Bo Hu, and Zang Li. 2022. Transrec: Learning transferable recommendation from mixture-of-modality feedback. *arXiv*.

Shoujin Wang, Liang Hu, Yan Wang, Longbing Cao, Quan Z Sheng, and Mehmet Orgun. 2019. Sequential recommender systems: challenges, progress and prospects. In *IJCAI*.

Libing Wu, Cong Quan, Chenliang Li, and Donghong Ji. 2018. PARL: let strangers speak out what you like. In *CIKM*.

Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, Hui Xiong, and Enhong Chen. 2023. A survey on large language models for recommendation. *CoRR*.

Table 4: Notation table.

Notation	Description
(u, i, t)	the user u interacted with item i at timestamp t
$\boldsymbol{x}$	the textual item representation
$\boldsymbol{h}$	review representation encoded by universal BERT
$\boldsymbol{p}$	the user preference extracted from reviews
$\boldsymbol{v}$	the item characteristics extracted from reviews
$\boldsymbol{e}_g$	the predicted intention
$\boldsymbol{e}_t$	the real intention

Table 5: Statistics of the datasets after preprocessing.

Dataset	#Users	#Items	#Inters	Sparsity	Avg.len
Scientific	8,442	4,385	59,427	99.970%	7.04
Prime	13,101	4,898	126,962	99.802%	9.69
Instruments	24,962	9,964	208,926	99.916%	8.37
Arts	45,486	21,019	395,150	99.959%	8.69
Office	87,436	25,986	684,837	99.970%	7.84

Liwei Wu, Shuqing Li, Cho-Jui Hsieh, and James Sharpnack. 2020. Sse-pt: Sequential recommendation via personalized transformer. In *RecSys*.

Zhe Xie, Chengxuan Liu, Yichi Zhang, Hongtao Lu, Dong Wang, and Yue Ding. 2021. Adversarial and contrastive variational autoencoder for sequential recommendation. In *WWW*.

Junjie Zhang, Yupeng Hou, Ruobing Xie, Wenqi Sun, Julian McAuley, Wayne Xin Zhao, Leyu Lin, and Ji-Rong Wen. 2023a. Agentcf: Collaborative learning with autonomous language agents for recommender systems. *arXiv preprint arXiv:2310.09233*.

Junjie Zhang, Ruobing Xie, Yupeng Hou, Wayne Xin Zhao, Leyu Lin, and Ji-Rong Wen. 2023b. Recommendation as instruction following: A large language model empowered recommendation approach. *arXiv preprint arXiv:2305.07001*.

Pengyu Zhao, Tianxiao Shui, Yuanxing Zhang, Kecheng Xiao, and Kaigui Bian. 2020. Adversarial oracular seq2seq learning for sequential recommendation. In *IJCAI*.

Lei Zheng, Vahid Noroozi, and Philip S Yu. 2017. Joint deep modeling of users and items using reviews for recommendation. In *WSDM*.

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*.

A Datasets

Table 5 shows the statistics of the used datasets after preprocessing.

B Baselines

We consider the following three types of baseline methods for performance comparison: (1) Traditional sequential recommendation methods: SASRec (Kang and McAuley, 2018) and BERT4Rec (Sun et al., 2019); (2) Intention-enhanced methods: SSE-PT (Wu et al., 2020) and ICLRec (Chen et al., 2022); (3) Universal text-enhanced methods: S^3-Rec (Zhou et al., 2020), ZESRec (Ding et al., 2021), and UniSRec (Hou et al., 2022); (4) Review-enhanced methods: RNS (Li et al., 2019). We give a detailed description of each baseline:

• **SASRec** (Kang and McAuley, 2018) is a transformer-encoder based sequential recommendation model, which employs a multi-head self-attention mechanism to capture sequential patterns.

• **BERT4Rec** (Sun et al., 2019) is a bidirectional self-attention recommendation model, where a cloze objective is designed to encode user behavioral sequences.

• **SSE-PT** (Wu et al., 2020) proposes to enhance the original transformer architecture, which relatively lacks personalization, by introducing supplementary user embeddings. The stochastic shared embeddings method is utilized for regularization.

• **ICLRec** (Chen et al., 2022) first encodes user behavioral representations using Transformer-based endoer. Then, these representations are clustered, and the cluster centroids are used as the user intention representation. An expectation-maximization (EM) framework is employed to repeat this process and obtain better intention representation.

• **S^3-Rec** (Zhou et al., 2020) adopts four auxiliary self-supervised objectives to learn the correlations among attributes, items, subsequences, and sequences via mutual information maximization.

• **ZESRec** (Ding et al., 2021) views pre-trained BERT representations as item representations for zero-shot recommendation. The encoding of textual representations allows for the incorporation of universal item description information.

• **UniSRec** (Hou et al., 2022) proposes to learn universal item and sequential representations. Two contrastive learning tasks with different sampling methods are designed to fuse semantics from different domains and learn universal sequential representation. It also introduces the MoE-enhanced adapter to adapt textual item representation.

• **RNS** (Li et al., 2019) conducts sequential rec-

ommendations with reviews, where both users and items are represented by the aspects extracted from their associated reviews. It integrates both user long-term preferences derived from reviews and short-term sequential patterns for recommendation.

C Performance Comparison w.r.t. Review Sparsity

Previous experiments show that our model can employ user reviews to estimate intentions. However, users in real-world systems may not always write reviews after interactions, making it challenging to analyze their intentions explicitly. To alleviate this issue, we propose three supplementary strategies. We report their performance with different review sparsity in Figure 3. Overall, even with 60% review sparsity, most of the proposed supplementary methods can outperform the competitive baseline UniSRec(Hou et al., 2022), which acquires item textual representation without using reviews to simulate intentions. This confirms the robustness of our proposed AuriSRec.

(1) *Replace with Item Description*: In this case, we replace missing reviews with relevant item descriptions. This helps simulate item features that might be mentioned in reviews. However, it does not consider user preference simulation. The performance drop highlights the effectiveness of user preference modeling.

(2) *Average of Other Reviews*: Here, we propose encoding the user's remaining reviews and calculating their average to simulate the representation of the missed review. Although the average representation can provide an estimation of the user's general preferences, it falls short of capturing the unique characteristics of each item, leading to the suboptimal simulation of user intentions.

(3) *Retrieve Other Users' Reviews*: Previous research suggests alleviating review sparsity by estimating a user's review using reviews from similar users (Wu et al., 2018). Here we use SASRec to encode each user's behavioral sequence as their representation. When user u interacts with item i, we simulate the missed review by selecting the review from users who have provided feedback on item i and possess the closet representations to u. Despite the inevitable performance decline, this approach outperforms others by considering both user preferences and item features. Moreover, there has been a growing emphasis on employing Large Language Models (LLMs) to provide recommendations (Wu et al., 2023; Fan et al., 2023; Lin et al., 2023). It is intuitive to leverage the powerful planning and generation capacities of LLMs to infer user intentions and simulate personalized reviews, which will be our future work.

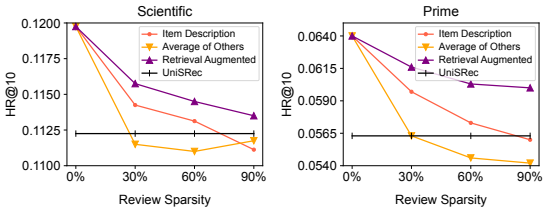

Figure 3: Performance comparison w.r.t. review sparsity on "Scientific" and "Prime" dataset.

D Case Study about Coupled Semantics in Reviews and Review Encoder effects

In this section, we present an illustrative case about coupled semantics (*i.e.,* user preferences and item characteristics) within reviews, shown in Figure 4.

We first sample a review r_{u_1,i_1} written by user u_1 to item i_1 from the Arts dataset. Then, we sample a review r_{u_1,i_2} written by the same user u_1 but for a different item i_2, as well as another review r_{u_2,i_1}, written for the same item i_1 bur by a different user u_2. We highlight the relevant user preferences and item characteristics within these reviews. As we can see, reviews written by the same user reflect similar preferences, while reviews for the same item show consistent characteristics. This phenomenon demonstrates the rationality of our proposed decoupling-based review encoder.

Then, we explore the encoding capacity of the proposed review encoder. Specifically, we encode the sampled reviews with different review encoders (*i.e.,* universal BERT, user head, and item head of decoupling-based review encoder) and compare the encoding similarity. As illustrated in Figure 4, we can find that using a universal language model like BERT as a review encoder makes it challenging to capture the intricate semantics in reviews, leading to similar representations for different reviews. In contrast, our proposed decoupling-based review encoder effectively separates the intertwined semantics within reviews. Specifically, the user head purifies user preferences from the reviews, resulting in similar representations for reviews written by the same user. Meanwhile, the item head purifies item characteristics, making reviews for the same item have similar representations. Overall, com-

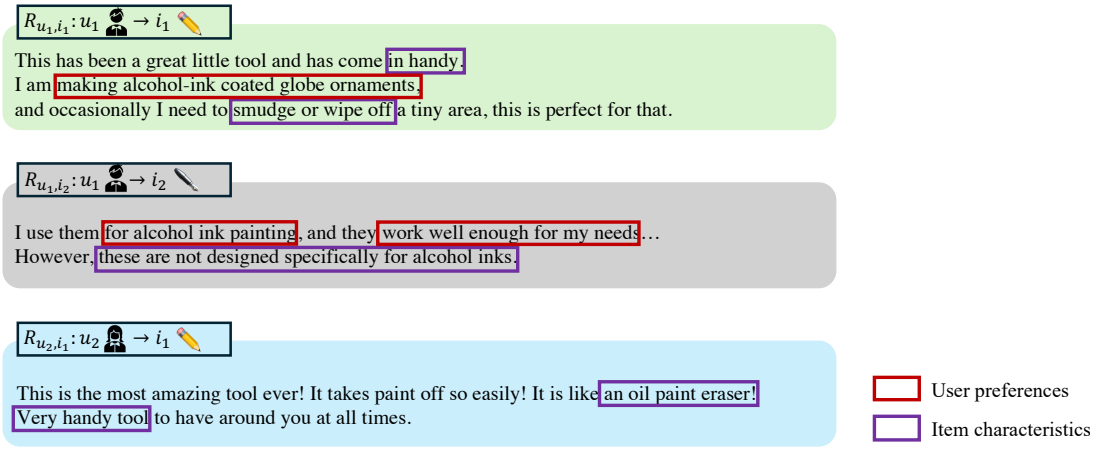

Figure 4: A case about user preferences and item characteristics components in reviews. Reviews written by the same user reflect similar user preferences, while reviews for the same item reflect consistent item characteristics.

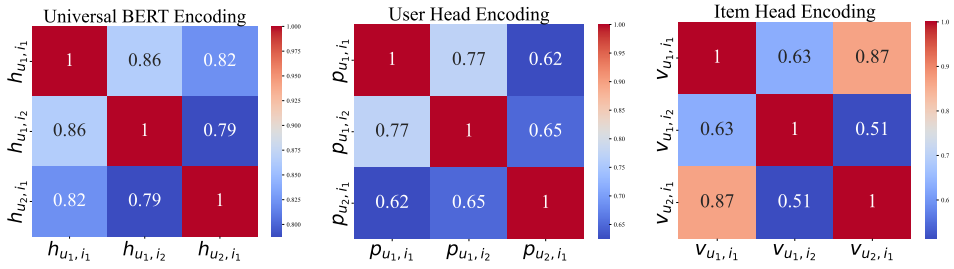

Figure 5: The cosine similarity of different review representations written by different users to different items, when using the universal BERT, user head of decoupling-based encoder, and item head of decoupling-based encoder as review encoder, respectively. Our proposed review encoder effectively purifies coupled semantics within reviews (*i.e.,* user preferences and item characteristics in this paper), resulting in more discriminative review representations.

pared to universal review encoders like BERT used in previous work, our proposed decoupling-based review encoder can generate more discriminative and purified review representations, thereby improving recommendation efficacy.

E Performance Comparison w.r.t. Cold-start Users.

Compared to traditional sequential recommenders relying solely on user behavioral sequences, we further introduce user historical reviews to improve recommendations. Here, we explore whether this rich information can alleviate the issue of user cold-start recommendations. We group the test data based on user behavioral sequence length (*i.e.,* the frequency of interaction), and then compare the improved HR@10 ratio with the baseline SASRec.

The results presented in Figure 6 show that AuriSRec consistently outperforms other baselines, regardless of whether users are extremely cold or active. This benefits from both the encoded textual item representation and the explicit preferences captured from reviews.

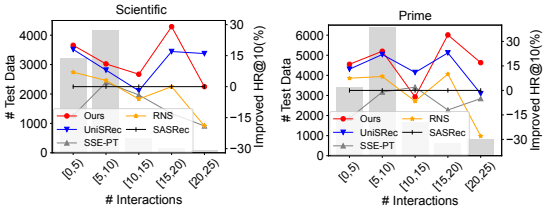

Figure 6: Performance comparison w.r.t. cold-start users on "Scientific" and "Prime" dataset.