DRE: Generating Recommendation Explanations by Aligning Large Language Models at Data-level

Yifan Wang[1], Shen Gao[1]✉, Jiabao Fang[2], Lisi Chen[1], Peng Han[1], and Shuo Shang[1]✉

[1] University of Electronic Science and Technology of China, Chengdu, China
202422081324@std.uestc.edu.cn, {shengao,chenlisi,penghan}@uestc.edu.cn, jedi.shang@gmail.com
[2] Shandong University, Qingdao, China
jiabaofang@mail.sdu.edu.cn

Abstract. Recommendation systems play a crucial role in various domains, suggesting items based on user behavior. And the lack of transparency in presenting recommendations can lead to user confusion. Thus, recommendation explanation methods are proposed to generate natural language explanations for users, which usually require intermediary representations of the recommendation model or need to conduct latent alignment training to the recommendation model. However, this additional training step usually causes potential performance issues due to the different training objectives between the recommendation task and the explanation task. In this paper, we introduce **D**ata-level **R**ecommendation **E**xplanation (DRE), a non-intrusive explanation framework for black-box recommendation models. We propose a data-level alignment method, leveraging large language models to reason relationships between user data and recommended items, without any additional training or intermediary representations for the recommendation model. Additionally, we also address the challenge of enriching the details of the explanation by introducing target-aware user preference distillation, utilizing item reviews. Experimental results on several benchmark datasets demonstrate the effectiveness of the DRE in providing accurate and user-centric explanations, enhancing user engagement with recommended items.

Keywords: Recommendation Explanation · Large language models · In-context learning.

1 Introduction

Recommendation systems (RecSys) play a pivotal role in learning user preferences and interests by analyzing historical user behavior data [11, 12]. Subsequently, the RecSys recommends relevant items from extensive databases, which are widely used in diverse domains such as e-commerce, news portals, and short video applications [4,6,13]. However, the direct presentation of recommended items may inadvertently confuse users, as they may not always comprehend the rationale

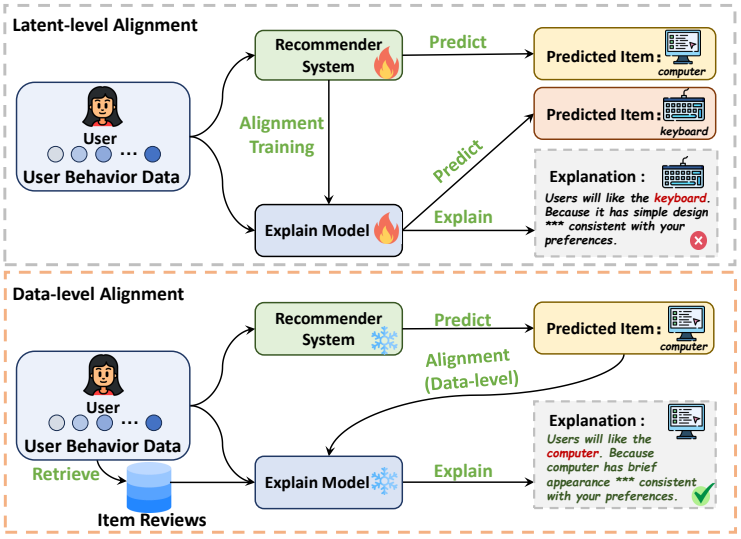


Fig. 1: Comparison between existing latent-level alignment and our data-level alignment recommendation explanation method.

behind a particular recommendation [7, 15]. This lack of transparency impedes users' inclination to explore the recommended item further [9, 28]. Consequently, interpreting the recommendation results of a black-box recommender model logically has always been an important research direction [2, 20]. Most of the existing methods [25, 27] usually focus on how to employ an additional explanation module to align with the recommendation system, subsequently generating natural language explanations.

However, there are two key challenges of these methods: (1) Existing methods [3, 26] often involve intrusion into the latent representations within the recommendation model, necessitating modifications to align the explanation and recommendation modules. Considering the different training objectives of these two modules, it could adversely affect the performance of both language generation and item recommendation. Moreover, although these methods aim to align two modules through training, they still cannot guarantee that the recommendation predictions of the two modules are consistent. Thus the discrepancies between the explained and recommended items may lead to user confusion. Additionally, in real-world applications, modifying the online serving recommendation model is very difficult. It also increases the overall system complexity, leading to a deep coupling between the recommendation and explanation modules. This does not align with the design principle of "low in coupling and high in cohesion" in software design. (2) The recommendation system based on ItemID models the co-occurrence relationships among items [8, 23], lacking an understanding of the specific semantic information about the items, such as the specific purposes of the products or the particular scenarios in which users use them. Thus, simply

aligning the explanation module with the recommendation module cannot provide rich detailed semantic information about the item. However, to generate helpful explanations, the explanation module requires comprehensive and diverse information to avoid generating explanations with hallucination information.

In this paper, we propose the **D**ata-level **R**ecommendation **E**xplanation (DRE) which can be applied to any black-box recommendation model without accessing intermediate representations or modifying the model. To avoid modifying the recommendation system, we propose a *data-level alignment method* to align the explanation module and the recommendation model. Figure 1 shows the comparison between our proposed paradigm and existing methods. Since the large language models (LLMs) have shown strong reasoning capability in many tasks [10, 19, 29], we propose to employ the LLM to reason the relationships between the user's historical data and recommended items [22]. Specifically, we feed the input user historical behavior data used by the recommendation model and the recommended item to the LLM. And we leverage the internal knowledge of LLM to find a reasonable relationship between the user preference and the attributes of the recommended item. This data-level alignment method can align these two modules without requiring any internal representation or intermediate result of the recommendation model, and it can easily be plugged into any RecSys.

For the second challenge, due to the limited detailed information of item descriptions, relying solely on item descriptions for inferring relationships between items can sometimes be challenging in uncovering implicit relationship information. Therefore, we propose utilizing the reviews of the items purchased by users and the reviews of the target recommended items to enhance the explanation module's understanding of user preferences and the semantics of target items. Since there is a lengthy of reviews for items that users have purchased, extracting relevant information from these reviews and generating explanations that better align with user preferences is a challenge. Thus, we introduce the *target-aware user preference distillation* method, which leverages the understanding and reasoning capabilities of LLM, employing semantic matching to extract target-aware information from reviews on items previously purchased by users. Finally, by incorporating the extracted target-aware information, we generate explanations for the recommended target items. Experiments conducted on several benchmark datasets from recommendation systems demonstrate that our proposed DRE generates explanations accurately describing aspects that users care about, thereby enhancing user interest in recommended items.

Our contributions are as follows:

• We propose DRE, an LLM-based non-intrusive explanation framework for recommendation systems. DRE generates explanations without accessing intermediate representations or modifying the recommendation model.

• We propose a data-level alignment method to align the explanation module and the recommendation module, leveraging LLMs to reason relationships between user data and recommended items.

• We introduce a target-aware user preference distillation method to distill user-related information from item reviews. Since not all the information is helpful in

reviews, DRE extracts product features that the user cares about.
• Experimental results on several benchmark datasets illustrate the advantage of DRE in terms of the accuracy of explanation.

2 Related Work

Explaining the black box of recommender systems has long been a prominent research direction in the field of recommender systems. Current research can be mainly divided into two categories. The first category focuses on identifying the most critical factors influencing recommendation results [5, 18]. [21] formulate an optimization problem to generate minimal changes to item aspects, thereby altering the recommended result. These aspects can be viewed as the composition of an explanation detailing why the original item is recommended. [14, 30] define information-based measures to identify the attributes that the model utilizes from the input to generate explanations. The second category mainly focuses on training a surrogate model to explain the target model. For example, [24] propose a reinforcement learning framework that gets rewards from the environment and modifies recommendation explanation. [16] propose a framework for generating explanations based on the knowledge graph. [15] employ LLMs as surrogate models, aiming to mimic and understand target recommender models by leveraging both natural language and latent spaces. After alignment, LLMs can generate target items and provide recommendation explanations.

However, existing methods either rely solely on a few entity words or keywords as explanations or employ complex fine-tuning approaches to generate natural language explanations. It makes the explanations not natural or complex to use, which requires fine-tuning or modification of existing recommendation systems.

3 DRE Methodology

In this section, we detail the **D**ata-level **R**ecommendation **E**xplanation (DRE). An overview of DRE is shown in Figure 2. It consists of three main components: Data-level Alignment, Target-aware User Preference Distillation and Explanation Generation.

The **Data-level Alignment** module first aligns the behavior of the explanation module and recommendation module at the data level, ensuring the explanation module can generate natural language explanations consistent with the recommendation results. The **Target-aware User Preference Distillation** module extracts target-aware information from reviews on items previously purchased by users, ensuring that the extracted product features align with the aspects user cares about. By incorporating the extracted target-aware information, the **Explanation Generation** module generates logically reasonable recommendation explanation for the recommended target item.

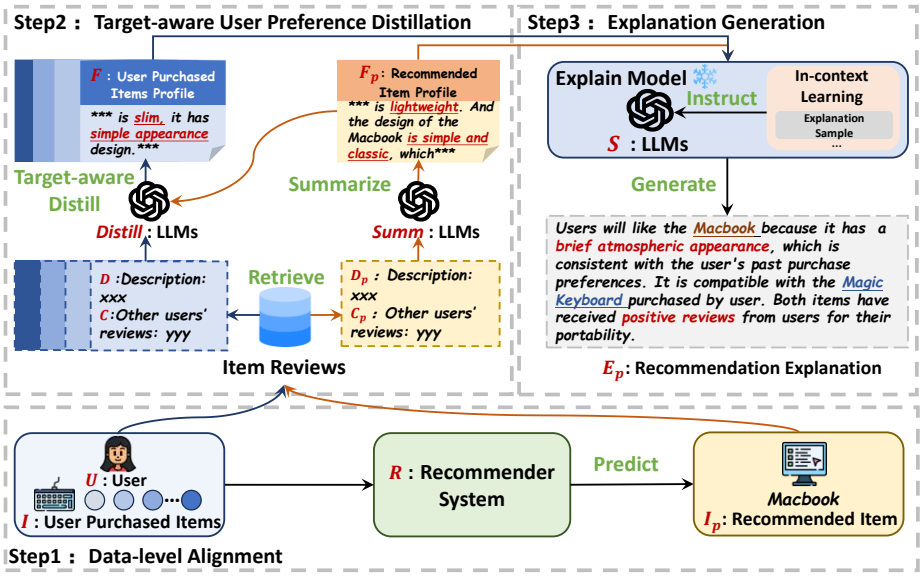


Fig. 2: Overview of DRE, which firstly align the explanation module and recommender with **Data-level Alignment**, and then generate the explanation by incorporating details of target from **Target-aware User Preference Distillation**.

3.1 Data-level Alignment

In order to generate precise explanations for recommended results, we propose a data-level alignment method to achieve behavioral consistency between the recommendation module and the explanation module. Given a list of items $I = \{I_1, I_2, \ldots, I_N\}$ which is purchased by the user U, the recommendation model R predicts items I_p that the user U might find interesting. To achieve alignment between the recommendation module and the explanation module, previous methods typically fine-tune the explanation module to perform the recommendation prediction task as well, generating items I_p consistent with the predictions of the recommendation model R. However, this approach inevitably reduces the text generation capability of the explanation module due to changes in its model structure and parameters. In this paper, we propose leveraging the in-context learning and reasoning abilities of LLM to align the explanation module with the recommendation module. Given inputs I and outputs I_p that are consistent with the recommendation model R, LLM can learn this prediction pattern in the context and explore the associated relationships to generate natural language explanations.

3.2 Target-aware User Preference Distillation

Relying solely on item IDs and item descriptions for recommendation explanations may fail to capture the details or user actual experiences of the item, which are crucial for users. Therefore, we propose to incorporate the reviews of user-purchased items I and the target item I_p predicted by the recommendation model R to assist the explanation model in obtaining more item detail information. Given a purchased item I_i ($I_i \in I, 1 \leq i \leq N$) of user U, we retrieve M reviews $C^i = \{C_1^i, C_2^i, \ldots, C_M^i\}$ of item I_i written by *other* users from the database, where each C_1^i represents a paragraph of natural language product review. Then, we can retrieve M user reviews for each purchased item I_i of user U, and then obtain a review set $C = \{C^1, C^2, \ldots, C^N\}$ which contains $M \times N$ reviews of other users. Similarly, we can also retrieve M reviews for the target item I_p denoted as $C^p = \{C_1^p, C_2^p, \ldots, C_M^p\}$ which is also written by other users. In this paper, we assume that the item characteristics described in the review set C are the key features that user U cares about, since the user U has bought these items. Therefore, we need to perform semantic matching between C and C^p to extract those item features that are both of interest to the user in the past purchased items and possessed by the target product I_p. We propose the *target-aware user preference distillation* method, which involves matching the target item reviews C^p with C to extract valuable information for generating recommendation explanations.

Since the description and reviews of items are usually quite long, and not all the information is helpful for generating recommendation explanations. For the target item I_p, we first construct an overview item profile F_p to distill the useful item features. We use the product description D_p and reviews information $C^p = \{C_1^p, C_2^p, \ldots, C_M^p\}$ of I_p as input and prompt the LLM to generate an item profile F_p:

$$F_p = \text{Summ}\left(\{C_1^p, C_2^p, \ldots, C_M^p\}, D_p\right), \tag{1}$$

where F_p contains both the basic information of the target item and user usage experiences and Summ is an LLM-based module that is prompted by the following instructions:

Instruction: You are given item's description and reviews. Response item profile using the following format: item:{item name}, description: {item description}, other users' reviews: {item reviews}. Extract key features from reviews.
Input: Target item description $<D_p>$, target item's reviews $<C^p>$.
Output: Target item profile F_p.

However, not all the product features mentioned in F_p may be of concern to the user U. Therefore, it is essential to extract the product features that user U care about from $C = \{C^1, C^2, \ldots, C^N\}$ associated with user behavior. We leverage LLM to extract product features by analyzing the relationships between the target item and the items previously purchased by the user. Specifically, we use the item profile F_p of the target item to filter reviews in set C^i of item I_i:

$$F_i = \text{Distill}\left(F_p, \{C_1^i, C_2^i, \ldots, C_M^i\}, D_i\right), \tag{2}$$

where D_i is the item description of item I_i, and Distill is an LLM-based module that is prompted by the following instructions:

Instruction: Finish history item profile using relevant features with recommended item, strictly adhere to the following format when responding:history item: {item name}, genre: {item genre}, relevant information: {item information}, other users' reviews: {reviews}. Relevant information mainly describes similarities between history item and recommended item, and summarize other users' reviews.
Input: History item $<D_i>$ and other users' reviews for this item $<C_1^i, C_2^i, \ldots, C_M^i>$; target item profile $<F_p>$. Find the similar features between the target item and the history item.
Output: Target-aware item profile F_i.

By integrating these two parts of information, we obtain the target-aware item profiles $F = \{F_1, F_2, \ldots, F_N\}$ for the items the user U has purchased.

3.3 Explanation Generation

Finally, we integrate the item profile F_p of the target item with the item profiles $F = \{F_1, F_2, \ldots, F_N\}$ of the items the user has purchased. We employ an in-context learning approach and instruct the LLM as follows to generate a logically coherent recommendation explanation that aligns with the recommendation system R and corresponds to user attention preferences:

$$E_p = S\left(F_p, \{F_1, F_2, \ldots, F_N\}\right), \tag{3}$$

where S is an LLM-based module to generate the recommendation explanation which is instructed by the following instructions:

Instruction: Now you are a recommendation assistant, combined with history relevant items, write an explanation of the recommended item. The format of response is as below: item: {target item} recommend reason: {reason}.
input: History items profiles $<F_i>$, target item profile $<F_p>$. Give the target item a detailed recommendation explanation by combining history items.
output: Recommendation explanation E_p.

For each recommendation explanation sample, we use the relevant values of the sample to fill in the fields of the prompt's angle brackets, constructing a sample-specific prompt.

4 Experimental Setup

4.1 Implementation Details

In our experiments, all DRE-C variants and the ChatGPT baseline use the gpt-3.5-turbo version, and the DRE-M variant and `Mistral` baseline use the Mistral

Table 1: Statistics of datasets.

Dataset	**#User**	**#Item**	**#Review**
Cell Phones & Accessories	12,467	6,977	38,729
Home & Kitchen	16,102	1,590	20,277
Clothing Shoes & Jewelry	19,310	3,746	24,712
Yelp	12,377	4,446	14,453

$8 \times 7B$ version which is open-sourced. And we update the memory modules of agents in DRE after each turn, meaning that only the suggestions and experiences from the previous turn are retained.

4.2 Evaluation Metrics

To quantitatively measure the performance of DRE, we propose two evaluation metrics in our paper: (1)**Aspect Score**: We assume that the aspects mentioned in the review C_U^p of the target item I_p written by user U are crucial to the user. We use the review C_U^p as a reference of the explanation E_p. We first employ the LLM to extract aspects of the review C_U^p. Subsequently, we measure the alignment between recommendation explanations E_p and user preferences by calculating the extent of the aspect overlap between E_p and C_U^p:

$$\text{Aspect\_Score} = \frac{1}{N_a}\sum_{i=1}^{N_a} hit(i) \in [0,1], \tag{4}$$

where N_a is the number of aspects in the user review C_U^p. To capture the user's detailed intent, we set N_a=7. And when the aspect i in the explanation is semantically the same as the aspect in the recommendation explanations E_p then $hit(i) = 1$, otherwise, $hit(i) = 0$. (2)**Rating Score**: Following [15], to directly evaluate the quality of the generated explanation, we implement a three-level scoring criteria to quantitatively evaluate the explanation generated by models: (i) RATING-1: Poor Explanation, using chunks of original sentence from provided data. (ii) RATING-2: Acceptable Explanation, consider only one aspect of user history and reviews, explaining unrelated items together. (iii) RATING-3: Satisfactory Explanation. We employ the LLM to evaluate the generated explanation according to these criteria and calculate the average rating score over all the test set.

4.3 Dataset

In this paper, we employ two commonly used recommendation datasets in the experiments: Amazon [17] and Yelp [3].The statistics of these datasets are shown

[3] https://www.yelp.com/dataset

in Table 1. In the Amazon dataset, we employ several categories, including Cell Phones & Accessories, Clothing Shoes & Jewelry, and Home & Kitchen. Intuitively, in order to better capture user preferences, we model user preferences only using positive user reviews. To construct the user purchase history, we limit the items sequence to a minimum of 4 items on Clothing Shoes & Jewelry, Home & Kitchen, and a minimum of 3 items on Cell Phones & Accessories. The last item is then used as the prediction target item. We select 100 samples in each category as testset and each item has associated reviews. We filtered the data by removing the sample of items with fewer than 2 user-purchased items and no accompanying reviews from users. In the Yelp dataset, we utilize attributes and categories associated with item as descriptions. We also select 100 samples from the Yelp dataset as the test set and filter the data with a length of historical data of less than 3 or at least 1 review.

4.4 Comparison Methods

We compare DRE to a state-of-the-art LLM-based recommendation explanation method and several LLMs, including: (i) `RecExplainer` [15] introduces an explanation approach by leveraging LLM, which employs three methods - behavior alignment, intention alignment, and hybrid alignment - in the latent spaces. (ii) `ChatGPT` [4] is a closed-source LLM from OpenAI. We use the version gpt-3.5-turbo-0613. We conduct recommendation explanation as a prompt learning method that uses a single instruction with the same input data as our DRE. (iii) `Mistral` [1] is an open-source LLM and we use the mixture-of-experts version with 8×70 billion parameters, and use the same prompt as `ChatGPT`.

We also employ two variants of DRE: **DRE-C** and **DRE-M** which use `ChatGPT` and `Mistral` as the LLM backbone respectively. To verify the effectiveness of each module in DRE, we also employ several ablation models: (i) `DRE-C w/o Rev.`: We remove all the reviews in our model and only use the description as input. (ii) `DRE-C w/o Dist.`: We directly summarize the description and reviews for the user-purchased item using Equation 1 without using the Distill method in Equation 2. (iii) `DRE-C w/o Dist.`+F_p: Based on `DRE w/o Dist.`, we also directly utilize the description and reviews of the target item without using the Summ method in Equation 1. (iv) `DRE-C w/` F_p: We directly generate the explanation by using the F_p as input to LLM, without using any information from user-purchased items. All the ablation studies are conducted based on **DRE-C**.

5 Experimental results

5.1 Main Results

Table 2 shows the performance of our proposed DRE and baselines in terms of two metrics. We can find that DRE shows superior performance in terms of all metrics compared to the state-of-the-art recommendation explanation

[4] https://chat.openai.com/

Table 2: Recommendation explanation performance comparison. ‡ indicates significant improvement over `ChatGPT` with $p \leq 0.01$ according to a Student's t test.

Method	Home & Kitchen		Clothing Shoes & Jewelry		Cell Phones & Accessories		Yelp	
	Aspect (↑)	**Rating** (↑)	**Aspect** (↑)	**Rating** (↑)	**Aspect** (↑)	**Rating** (↑)	**Aspect** (↑)	**Rating** (↑)
`RecExplainer` [15]	0.6057	2.64	0.5628	2.68	0.6028	2.64	0.3238	2.86
`Mistral` [1]	0.7028	2.65	0.5757	2.79	0.6571	2.00	0.4642	2.65
`ChatGPT` [4]	0.6971	2.51	0.6362	2.86	0.6229	2.67	0.4200	2.79
DRE-M	0.7142	2.68	0.6485	2.89	0.6857	2.57	0.5542	2.82
DRE-C	**0.7714**‡	**2.88**†	**0.6728**‡	**2.94**‡	**0.7400**‡	**2.90**‡	**0.5600**‡	**2.91**‡
`DRE-C w/o Rev.`	0.6914	2.64	0.6400	2.65	0.6542	2.66	0.4242	2.83
`DRE-C w/o Dist.`	0.6278	2.79	0.5714	2.77	0.6057	2.89	0.5542	2.86
`DRE-C w/o Dist.+`F_p	0.5828	2.77	0.5671	2.82	0.5971	2.83	0.5028	2.83
`DRE-C w/` F_p	0.7385	1.64	0.5814	2.06	0.6585	2.03	0.4285	1.50

method `RecExplainer`.This phenomenon indicates that compared to the latent-level alignment, our data-level alignment is capable of generating explanations of higher quality. Since we employ the data-level alignment method between the explanation model and the recommendation model, our DRE not only exhibits high quality, but also does not require any data for model training. This significantly enhances the applicability of the method, making it usable in scenarios without labeled data, and also reduces the issue of domain transfer caused by the labeled datasets.

We can also find our proposed DRE achieves superior performance compared with its LLM backbones respectively. Although the LLM backbones (*e.g.,* `Mistral` and `ChatGPT`) use the same input data as our proposed DRE, they cannot generate a high-quality recommendation explanation. Since LLMs can only reveal a limited relationship between user-purchased items and target item based solely on descriptions. This phenomenon demonstrates that our proposed target-aware user preference distillation method can assist the model in capturing more user preference information.

5.2 Ablation Study

To evaluate the effectiveness of each module in DRE, we also conduct ablation studies with model DRE-C, and the results are shown in Table 2. We found that the `DRE-C w/o Rev.` method achieves lower scores compared to other ablation models, indicating the effectiveness of integrating review information in our approach. Due to the complexity of information in reviews, generating meaningful explanations requires extracting target-aware information. Therefore, `DRE-C w/o Dist.` also exhibited lower performance after removal Distill module from DRE.

Additionally, since descriptions and reviews are usually quite long, extracting helpful information about recommended item requires distilling useful features from description and reviews. Therefore, `DRE-C w/o Dist.+`F_p method exhibited lower performance after removal Summ module from `DRE-C w/o Dist.`

Table 3: Human evaluation results for two datasets.

	Clothing Shoes & Jewelry	Cell Phones & Accessories
`RexExplainer` [15]	1.80	1.80
`Mistral` [1]	1.60	1.87
`ChatGPT` [4]	1.87	1.60
DRE-M	2.60	2.53
DRE-C	2.67	2.73

5.3 Human Evaluation

In previous experiments, we used LLM to assess recommendation explanation quality. In this section, we employ two well-educated human annotators evaluate it directly. We use the same evaluation criteria as the rating score as shown in § 4.2. We conducted human evaluation on 60 randomly selected recommendation explanation samples from the Clothing Shoes & Jewelry and Cell Phones & Accessories dataset respectively. From Table 3, we can find that although the scores from the human evaluation and LLM scores (as shown in Table 2) do not fully align, the rankings among the baselines are consistent. To validate LLM-based evaluations, we assessed consistency with human evaluation using Cohen's kappa. The kappa value of 0.463 indicates moderate agreement, further supporting the consistency between LLM and human evaluations.

Additionally, to directly compare the differences in recommendation explanations generated by DRE and `ChatGPT`, we asked data annotators to directly compare the results from the two models. Specifically, we presented the data annotators with recommendation explanations generated by DRE and `ChatGPT` in random order and classified them according to the following criteria: (i) No significant difference between the two explanations; (ii) DRE better aligns with user preferences; (iii) `ChatGPT` better aligns with user preferences; The final results showed that 18.88% of samples are classified as category I, 61.11% as category II, and 20% as category III. This demonstrates that our proposed method offers significant advantages over directly prompting `ChatGPT`.

5.4 Case Study

Figure 3 shows an example of recommendation explanations generated by `ChatGPT`, `RecExplainer`, and DRE-C based on information about user-purchased items and recommended item. The blue and yellow highlighted text in the explanation indicates the recommended item and user-purchased items. We use the text in red to illustrate the shortcomings of the explanation, which is not generated by the model. The text in green shows target-aware information generated by the model. The text in blue represents the consistent information of reviews from user U for user-purchased items and recommended item. From this case, we can find that `ChatGPT` fails to establish convincing and reasonable relationships between recommended items and user preferences. Although `RecExplainer` employs the

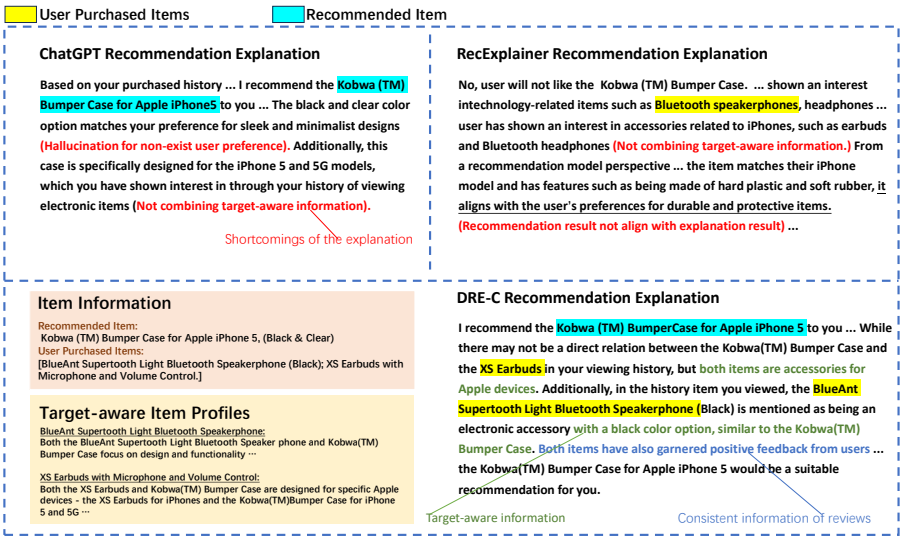
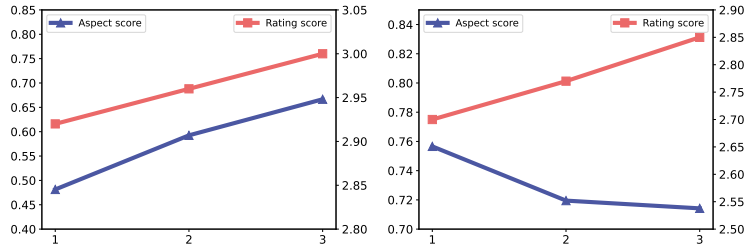


Fig. 3: Examples of the generated recommendation explanation of two baselines and DRE.

complicated alignment training step for the recommendation module, the generated explanation still fails to align with the recommendation result (as shown in the red text in the bracket). And DRE provides target-aware information that is persuasive and aligns with user preferences. This observation demonstrates that our proposed target-aware user preference distillation can effectively filter target-aware information from reviews and descriptions.

5.5 Analysis of Different Input



(a) Performance of using different numbers of user-purchased items.

(b) Performance of using different numbers of reviews.

Fig. 4: Performance analysis of using different numbers of user history and reviews.

To verify the impact of the quantity of product reviews and the amount of user-purchased items on the performance, we measured the change in model performance under different input data settings. Figure 4(a) shows the effect of the amount of user-purchased items on the model's performance, From this figure, we can observe an upward trend in both aspect and rating scores, which demonstrates that incorporating more user historical purchase items into the model helps the model to more comprehensively understand user preferences.

Figure 4(b) shows the trend in model performance as the number of input reviews changes. As the number of item reviews a user has increased, the model pays more attention to these reviews, resulting in a focus on analyzing other user reviews of the item and a reduction in the description of item features. Since the aspect score focuses more on evaluating the description of the item features, this leads to a decrease in the score as shown in Figure 4(b). However, this decrease does not indicate a decline in the quality of the recommendation explanation. If the user prioritizes the consistent reviews from other users between the target item and user-purchased items, more user reviews will help the model capture the consensus of user opinions on the item. If the user focuses more on the consistent product features between the target item and user-purchased items, the number of reviews can be appropriately reduced. Therefore, the number of product reviews can be adjusted according to the user's preference to achieve the desired recommendation explanation.

5.6 Analysis of Different Hyper-parameters

The temperature parameter in the transformer-based language model controls the randomness and diversity of text generation, and higher temperature results in generating more diverse text [5]. To assess the influence of temperature setting on the DRE, we conducted experiments using different temperature configurations on the Home & Kitchen dataset. Since the recommendation explanation task requires both diverse explanations and fidelity to product attributes and user reviews, from Figure 5, we can find that both too high and too low temperature parameter can lead to a decrease in model performance.

5.7 Computational Cost

Since our proposed DRE is a multi-module method based on prompting LLM, we provide statistics on the total token consumption of DRE and the token consumption of each module separately. Table 4 compares the token consumption of our proposed method with several baseline methods.

Firstly, from the results, it can be seen that the Distill module in our proposed DRE consumes the most tokens compared to the other two modules. Since the Distill module is responsible for generating target-aware items profiles F_N, which requires using a large amount of item information as input and analyzing product associations, it consumes a significant number of tokens. Furthermore, as shown

[5] https://platform.openai.com/docs/guides/text-generation/completions-api

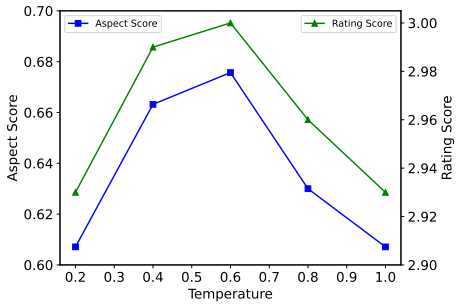


Fig. 5: Performance of using different temperature settings in DRE.

Table 4: Statistics of token consumption for baselines. We show the token consumption of each module in DRE-C in the last three rows. The number in the bracket represents the percentage of tokens consumed by the module relative to the total token consumption of the model.

	Home & Kitchen	**Clothing Shoes & Jewelry**	**Cell Phones & Accessories**	**Yelp**
`DRE-C`	13641	20374	11819	11578
`ChatGPT`	3331	2227	3096	2850
Sub-modules in DRE				
`Summ`	2059 (15.09%)	3138 (15.40%)	2530 (21.41%)	1438 (12.42%)
`Distill`	9046 (66.31%)	12752 (62.59%)	7055 (59.69%)	7847 (67.78%)
`Explain`	2536 (18.59%)	4484 (22.01%)	2234 (18.90%)	2293 (19.80%)

in the ablation study in Table 2, the Distill module contributes the most to the overall performance improvement in DRE-C (compared between DRE-C and `DRE-C w/o Dist.`). Since `ChatGPT` uses only simple instructions as prompts to directly generate recommendation explanations, its token consumption is lower than our method. However, the quality of the explanation generated by `ChatGPT` is significantly lower than those produced by DRE-C as shown in Table 2.

6 Conclusion

In this paper, we introduced **D**ata-level **R**ecommendation **E**xplanation (DRE), a non-intrusive explanation framework for black-box recommendation models. We propose a data-level alignment method to align the explanation module and the recommendation model without additional parameter training or intermediate representations in recommendation model. Since the detailed information in the item description is limited, we propose the target-aware user preference distillation method to enhance semantic understanding by incorporating item reviews when generating recommendation explanations. Experimental results demonstrate the effectiveness of DRE in providing accurate and user-centric explanations, contributing to the improvement of recommendation system interpretability and user engagement.

Acknowledgment

This work was supported by the National Key R&D Program of China (No. 2023YFC3305600), the Natural Science Foundation of China (T2293773, 62432002, 62406061), and the Natural Science Foundation of Shandong Province (ZR2023QF159).

References

1. Mixtral of experts: Mixtral-8x7b. https://mistral.ai/news/mixtral-of-experts/, accessed: 2024-02-02
2. Bilgic, M., Mooney, R.J.: Explaining recommendations: Satisfaction vs. promotion. In: Beyond personalization workshop, IUI. vol. 5, p. 153 (2005)
3. Chen, J., Zhang, H., He, X., Nie, L., Liu, W., Chua, T.S.: Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In: Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval. pp. 335–344 (2017)
4. Chen, L., Shang, S.: Region-based message exploration over spatio-temporal data streams. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 873–880 (2019)
5. Chen, X., Qin, Z., Zhang, Y., Xu, T.: Learning to rank features for recommendation over multiple categories. In: Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. pp. 305–314 (2016)
6. Chen, Z., Zhang, D., Feng, S., Chen, K., Chen, L., Han, P., Shang, S.: Kgts: Contrastive trajectory similarity learning over prompt knowledge graph embedding. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 8311–8319 (2024)
7. Cheng, M., Liu, Z., Liu, Q., Ge, S., Chen, E.: Towards automatic discovering of deep hybrid network architecture for sequential recommendation. In: Proceedings of the ACM Web Conference 2022. pp. 1923–1932 (2022)
8. Diao, Q., Qiu, M., Wu, C.Y., Smola, A.J., Jiang, J., Wang, C.: Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 193–202 (2014)
9. Du, J., Zhou, S., Yu, J., Han, P., Shang, S.: Cross-task multimodal reinforcement for long tail next poi recommendation. IEEE Transactions on Multimedia **26**, 1996–2005 (2023)
10. Gao, S., Li, H., Shi, Z., Huang, C., Tu, Q., Shang, S., Tian, Z., Huang, M.: 360rea: Towards a reusable experience accumulation with 360 assessment for multi-agent system. In: Findings of the Association for Computational Linguistics ACL 2024. pp. 13149–13162 (2024)
11. Gao, S., Ren, Z., Zhao, Y.E., Zhao, D., Yin, D., Yan, R.: Product-aware answer generation in e-commerce question-answering. Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (2019), https://api.semanticscholar.org/CorpusID:59158861
12. Guo, H., Tang, R., Ye, Y., Li, Z., He, X.: Deepfm: a factorization-machine based neural network for ctr prediction. arXiv preprint arXiv:1703.04247 (2017)
13. Han, P., Zhou, S., Yu, J., Xu, Z., Chen, L., Shang, S.: Personalized re-ranking for recommendation with mask pretraining. Data Science and Engineering **8**(4), 357–367 (2023)

14. Lakkaraju, H., Kamar, E., Caruana, R., Leskovec, J.: Interpretable & explorable approximations of black box models. arXiv preprint arXiv:1707.01154 (2017)
15. Lei, Y., Lian, J., Yao, J., Huang, X., Lian, D., Xie, X.: Recexplainer: Aligning large language models for recommendation model interpretability. arXiv preprint arXiv:2311.10947 (2023)
16. Ma, W., Zhang, M., Cao, Y., Jin, W., Wang, C., Liu, Y., Ma, S., Ren, X.: Jointly learning explainable rules for recommendation with knowledge graph. In: The world wide web conference. pp. 1210–1221 (2019)
17. Ni, J., Li, J., McAuley, J.: Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In: Conference on Empirical Methods in Natural Language Processing (2019), https://api.semanticscholar.org/CorpusID:202621357
18. Pan, D., Li, X., Li, X., Zhu, D.: Explainable recommendation via interpretable feature mapping and evaluation of explainability. arXiv preprint arXiv:2007.06133 (2020)
19. Shang, S., Yao, Z., Fu, H., Tao, C., Chen, X., Wang, F., Wang, Y., Ren, Z., Gao, S.: Unified multi-scenario summarization evaluation and explanation. IEEE Transactions on Knowledge and Data Engineering (2024)
20. Sharma, A., Cosley, D.: Do social explanations work? studying and modeling the effects of social explanations in recommender systems. In: Proceedings of the 22nd international conference on World Wide Web. pp. 1133–1144 (2013)
21. Tan, J., Xu, S., Ge, Y., Li, Y., Chen, X., Zhang, Y.: Counterfactual explainable recommendation. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. pp. 1784–1793 (2021)
22. Wang, H., Feng, S., Chen, L., Liu, Y., Shang, S.: Simulating individual infection risk over big trajectory data. In: International Conference on Database Systems for Advanced Applications. pp. 136–151. Springer (2024)
23. Wang, X., He, X., Feng, F., Nie, L., Chua, T.S.: Tem: Tree-enhanced embedding model for explainable recommendation. In: Proceedings of the 2018 world wide web conference. pp. 1543–1552 (2018)
24. Wang, X., Chen, Y., Yang, J., Wu, L., Wu, Z., Xie, X.: A reinforcement learning framework for explainable recommendation. In: 2018 IEEE international conference on data mining (ICDM). pp. 587–596. IEEE (2018)
25. Wang, Y., Chu, Z., Ouyang, X., Wang, S., Hao, H., Shen, Y., Gu, J., Xue, S., Zhang, J.Y., Cui, Q., et al.: Enhancing recommender systems with large language model reasoning graphs. arXiv preprint arXiv:2308.10835 (2023)
26. Xu, L., Zhang, J., Li, B., Wang, J., Cai, M., Zhao, W.X., Wen, J.R.: Prompting large language models for recommender systems: A comprehensive framework and empirical analysis. arXiv preprint arXiv:2401.04997 (2024)
27. Xu, W., Cai, D., Zhang, Z., Lam, W., Shi, S.: Reasons to reject? aligning language models with judgments. arXiv preprint arXiv:2312.14591 (2023)
28. Zhang, Y., Chen, X., et al.: Explainable recommendation: A survey and new perspectives. Foundations and Trends® in Information Retrieval **14**(1), 1–101 (2020)
29. Zhao, W.X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., et al.: A survey of large language models. arXiv preprint arXiv:2303.18223 (2023)
30. Zilke, J.R., Loza Mencía, E., Janssen, F.: Deepred–rule extraction from deep neural networks. In: Discovery Science: 19th International Conference, DS 2016, Bari, Italy, October 19–21, 2016, Proceedings 19. pp. 457–473. Springer (2016)