Agent4ExpRec: Towards Generating and Evaluating User-Centric Recommendation Explanation

Yayu Hou, Dongjin Yu*, Dongjing Wang, Changgeng Li
Hangzhou Dianzi University
{houyy, yudj, dongjing.wang, cgli}@hdu.edu.cn

Abstract

Explainable Recommendation (ER) aims to improve the transparency of Recommender Systems (RS) by generating natural language explanations. However, existing methods primarily optimize textual similarity to reference explanations, which often leads to item-centric explanations focusing on product attributes rather than users' personalized preferences. Moreover, they usually do not care how the generated explanations influence the behavior of users. To address these limitations, we introduce Agent4ExpRec, a closed-loop user-centric pipeline designed for generating and evaluating recommendation explanation. Based on the original explanations, it enables the peer agent to analyze the user's requirements from multiple aspects and provides feedback, and automatically plans the refinement process to generate explanations focusing on user preferences. During the evaluation stage, we analyze how different explanations influence user behavior, providing evidence that our generated explanations are more consistent with users' personalized preferences. Since the actual user behavior on large-scale explanation text is difficult to obtain in practice, we use LLM-powered agents equipped with user profile to simulate user interactions and decision-making in ER scenarios. Extensive experiments demonstrate that the generated explanations consistently align with user preferences, highlighting the potential of user-centric refinement for explainable recommendation tasks. Our code is available at https://github.com/yy-Hou/Agent4ExpRec.

1 Introduction

Recommender Systems (RS) are capable of analyzing user preferences and intentions to generate personalized recommendations for individual users (Lei et al., 2024). However, these systems often function as 'black boxes', making it difficult for users to understand how or why a particular item was recommended (Gedikli et al., 2014). Explainable Recommendations (ER) have been proposed to enhance explainability by providing users with human-interpretable explanations that justify how the recommended item aligns with their preferences (Yang et al., 2024; Li et al., 2021). This not only increases users' trust in the system, but also facilitates their decision-making (Ma et al., 2024).

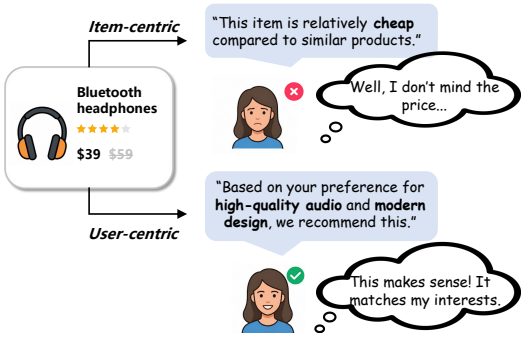


Figure 1: Comparison of item-centric explanation and user-centric explanation. Users' acceptance of recommendations depends not only on the item itself, but also on whether the recommendation aligns with their preferences.

Existing ER methods have yielded promising results by leveraging ID-level representations (Li et al., 2017a; Dong et al., 2017b) and advanced language models (e.g., Transformer (Li et al., 2021), GPT (Li et al., 2023a) and other LLMs (Kim et al., 2025)). Nonetheless, most existing solutions typically follow the supervised learning paradigm, generating explanation texts which are similar to the ground truth under supervision from textual similarity signals (e.g., BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004)). These frameworks overlook user feedback signals for explanations, producing explanations that fail to reflect users' personalized interests and preferences. As depicted in Figure 1, what users are concerned about is the connection between the item and personal preferences, rather than the price of the product. By prioritizing tex-

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 36200–36215
October 24-29, 2026 ©2026 Association for Computational Linguistics

tual similarity as the primary optimization target, existing studies neglect several critical challenges in ER: How can user-centric explanations be effectively generated? How can we evaluate whether an explanation is satisfactory to users? How can explanations influence user behavior in recommendation scenarios?

Several studies employ human studies and LLM-based evaluation, showing that LLMs can serve as reliable and reproducible evaluators for explanation assessment (Lu et al., 2023; Zhang et al., 2024b). However, these works mainly focus on evaluating rather than improving explanation generation. HF4Rec (Tang et al., 2025) explores the use of LLM feedback as multi-perspective supervisory signals for retraining ER models. Nevertheless, its reliance on limited corpora restricts its ability to generalize to diverse real-world user preferences. Furthermore, an explanation refinement framework has been proposed to optimize explanations according to explicit user goals (Zhang et al., 2025), but its heuristic refinement strategy remains insufficient for handling large-scale personalized recommendation scenarios.

Recently, LLM-powered agents have been widely used for emulating human behavior in various scenes (Park et al., 2023; Dubois et al., 2023). Inspired by Agent4Rec (Zhang et al., 2024a), we posit that LLM-powered agents can be leveraged to simulate users in explainable recommendation scenarios. To this end, we propose a framework that draws on agents to develop a user simulator in ER scenarios and generate user-centric explanations. Our framework incorporates fine-grained refinement among agents that specialize in different skills, consisting of a peer, a planner, and a refiner. Specifically, we employ an explanation generating tool to generate the initial explanation. Then, the peer agent is assigned to analyze the initial explanation text accompanied by user profile. The planner plans the refine path based on the feedback, while the refiner conducts targeted refinement and generates the final explanation presented to the user.

To assess user experience with explanations, we employ user agents to simulate user behavior and collect interaction signals. Specifically, during the recommendation stage, both recommended items and their corresponding explanations are presented to users, and user behaviors are analyzed to evaluate the effectiveness of the explanations. We further demonstrate that the simulated behaviors are consistent with real user behavior. To evaluate the effectiveness of Agent4ExpRec, we conduct comprehensive experiments on three real-world datasets. The results show that our approach effectively generates accurate explanations, improving the click-through rate and satisfaction of users.

The key contributions of this work can be summarized as: (1) We highlight the gap between existing ER methods and user demands of explanations. To bridge this gap, we propose a multi-agent framework to collect feedback and refine explanations. (2) We introduce a closed-loop pipeline integrating explanation generation and evaluation. We incorporate collective preferences to alleviate data noise and bias, and evaluate the alignment capability by measuring its consistency with real user behaviors. (3) We design a novel user simulation platform that make it possible for user behavior collection and analysis in ER. This reveals the influence of recommendation explanations, providing a direction for the evaluation and improvement. (4) Experiments on three diverse, real-world datasets demonstrate that the proposed Agent4ExpRec framework can emulate users in ER and generates high-quality explanations.

2 Related Work

2.1 Explainable Recommendation

Early embedding-based ER methods integrate item-side information into recommendation models to jointly optimize recommendation and explanation generation capabilities (Zhang et al., 2014a; Diao et al., 2014). In recent years, the post-hoc approach has gained significant traction. For instance, some studies employed Recurrent Neural Networks (RNNs), such as NRT (Li et al., 2017b) and Att2Seq (Dong et al., 2017a), while others utilized Transformer-based architectures (Li et al., 2021; Raczyński et al., 2023). Despite their effectiveness, these ID-based methods suffer from limited generalization to cold-start users and items. More recently, with the emergence of large language models (LLMs), approaches such as PEPLER (Li et al., 2023b) and fine-tuned LLM variants (Yang et al., 2024) have demonstrated strong capabilities in generating personalized explanations. However, these frameworks often incur substantial computational overhead.

2.2 LLM-based Agent for RS

Recent advances in LLMs have enabled the development of agent-based recommender systems,

where autonomous agents equipped with capabilities such as profiling, memory, and planning are employed to accomplish complex recommendation tasks (Mao et al., 2025; Chen et al., 2025; Wang et al., 2024; Huang et al., 2025). Existing approaches can be broadly categorized into three directions: recommendation-oriented methods, which improve recommendation through multi-agent collaboration; interaction-oriented methods, which enhance recommendations via user interactions; and simulation-oriented methods, which model realistic user behaviors for recommendation evaluation. Despite their effectiveness, these methods primarily emphasize task completion and collaboration efficiency, while paying limited attention to personalized explanation generation. In contrast, our framework incorporates user modeling and personalized feedback into the explanation generation process through collaborative agents, enabling the generation of explanations that are more faithful to individual preferences.

3 Methodology

We introduce Agent4ExpRec, a user-enhanced explanation generation and evaluation framework that incorporates user simulation in explainable recommendation scenarios, as illustrated in Figure 2.

3.1 Fine-Grained Explanation Refinement

Although existing ER methods excels in generating explanations similar to ground truth which are derived in reviews, it's remains doubt that whether the explanation are satisfied to users. As a result, we design a multi-agent collaborative framework to generate and refine explanations from user perceptions. Given a user set $\mathcal{U}$ and an item set $\mathcal{I}$, let $y_{u,i} = 1$ denote that user u has interacted with item i and provided a rating $r_{u,i} \in \{1, 2, 3, 4, 5\}$; otherwise, $y_{u,i} = 0$. A recommendation model f predicts $\hat{y} = f(u, i)$ and generates a recommendation list $\mathcal{I}_{rec} = \{i_1, i_2, \cdots, i_{n'}\}$, where $n' = p_{max_page} \times q_{items_per_page}$. We leverage explanation generation tool to generate initial explanations $e^0_{u,i}$ to justify why item i is recommended to user u.

$$e^0_{u,i} = g(u, i, \mathcal{P}_u, \mathcal{C}_i), \tag{1}$$

where $\mathcal{P}_u$ denotes the user profile and $\mathcal{C}_i$ denotes item details.

Traditional methods often generate explanations that excessively focus on matching reference texts, while overlooking users' actual preferences and needs. Rather than emphasizing textual similarity, our approach prioritizes the user perspective. We introduce the peer agent that evaluates and provides feedback on explanations from the perspective of the user. Given the initial explanations of recommended items, the peer agent analyzes the user profile and items information to give a 5-scale Likert score to explanations (Zhang et al., 2024b) from 4 aspects:

-Persuasiveness: This explanation is convincing to the user.

-Transparency: Based on this explanation, the user understands why this movie is recommended.

-Accuracy: This explanation is consistent with user's interests.

-Satisfaction: The user will be satisfied with this explanation.

We assign independent peer agent the role of human-like evaluator and require it to evaluate the generated explanations, which follows human research paradigms. In addition, this ensures that feedback concentrates on the explanation generation process without being affected by fatigue or other confounding factors, thus improving the stability and reliability of the results.

Some researchers have designed static refinement paths for predefined user goals. Nevertheless, such heuristic path design fails to account for the diverse needs of users, undermining the diversity of effective refinement. We design a planner agent, which allows LLM to plan refinement path in response to feedback. This makes the process highly versatile and able to handle complex, real-world scenarios. The planner agent takes the current explanation and feedback signal as input and generates a structured refinement plan:

$$\pi_{u,i} = Planner(\mathcal{F}_{u,i}, \mathcal{P}_u, \mathcal{C}_i), \tag{2}$$

During the refinement stage, the refiner is responsible for performing scheduled rewriting on recommendation explanations at the final stage of the explanation generation pipeline. It takes as inputs the user profile, the content summary of each candidate item, the initial explanation text, as well as the structured refine-plan and feedback summary derived from the evaluation and planning phase. It then refines the final explanation corresponding to each item, thereby converting the diagnosis-and-planning results into actionable textual optimiza-

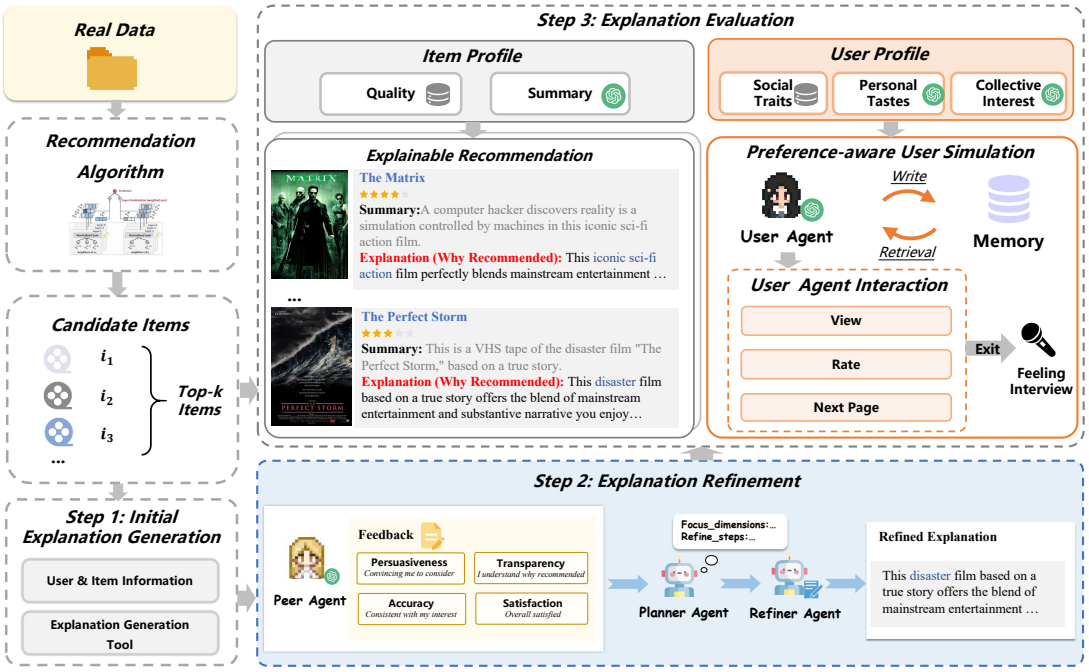


Figure 2: The overall framework of Agent4ExpRec. The simulator consists of explainable recommendation environment and user agents. Given the recommendation ranking derived from the recommendation algorithm, an initial explanation is generated by an explanation generation tool. Afterwards, through feedback and subsequent refinement, the final explanation is displayed to users together with the recommended content in a page-by-page manner.

tion outcomes.

$$e^*_{u,i} = Refiner(e^0_{u,i}, \pi_{u,i}, \mathcal{P}_u, \mathcal{C}_i) \quad (3)$$

The process is presented in Algorithm 1.

3.2 Explainable Recommendation Simulator

In this paper, we concentrate on post-hoc explanation generation. To reconstruct the explainable recommendation scenario, we elaborate on item profile generation, user agent design and recommendation page construction.

Item profile. This module comprises two components: quality and summary. Quality is obtained from historical user ratings, which is represented by $R_i = \frac{1}{\sum_{u\in\mathcal{U}} y_{ui}} \sum_{u\in\mathcal{U}} y_{ui} \cdot r_{ui}$, while summary is automatically generated by LLM.

User Agent. We provided user agents to emulate individualized human behaviors, each agent is equipped with user profile. Specifically, the user profile incorporates the perspectives of social traits, unique tastes, and collective interests to guide the LLM to simulate real users. The mathematical calculations of social traits are shown as follows: Activity captures user interaction frequency, and is defined for user u as: $T^u_{act} = \sum_{i\in\mathcal{I}} y_{ui}$. Conformity captures how closely a user's ratings follow average item ratings, and is defined for user u as: $T^u_{conf} = \frac{1}{\sum_{i\in\mathcal{I}} y_{ui}} \sum_{i\in\mathcal{I}} y_{ui} \cdot |r_{ui} - R_i|^2$. Then we utilize LLM to summarize the user unique tastes based on user interaction data.

While users exhibit diverse preferences, they also share collective interest shaped by trends or social factors (Feng et al., 2026). Users with similar behaviors may interact with an item for the same reason. We use collective interest as the third component of user profile. To obtain collaborative relations among users, we utilize the user embeddings z_u obtained from the recommendation model for clustering, and find the intersection of the categories that users in the same cluster have interacted with, then leverage the LLM to generate a collective interest.

ER environment construction. We present a list of recommended items and corresponding explanations to user. The recommended list $\mathcal{I}_{rec}$ is obtained by recommendation algorithm. Then, we assign user agents with user profiles to action as humans based on the observed content: view, rate, and generate post viewing feelings for items. If the user agent expresses fatigue or dissatisfaction, it can exit the current page. Otherwise, the user agent will further simulate and move on to the next page.

Table 1: Statistics of the datasets.

Dataset	Amazon	TripAdvisor	Yelp
#Users	4,811	9,765	27,147
#Items	5,459	6,280	20,266
#Interactions	288,693	320,023	1,293,247

The algorithm description of the simulation process is presented in Algorithm 2. All the prompts can be found in the Appendix E.

4 Experimental Setup

4.1 Datasets

We use three real-world explainable recommendation datasets that cover diverse domains of user interaction: Amazon (movies & TV)[1], TripAdvisor (hotel)[2] and Yelp (restaurant)[3]. We directly use their corresponding recommendation explanations collected from GitHub repository[4] of (Li et al., 2021), where each instance includes a user ID, item ID, and an explanation sentence, along with additional information such as ratings, item title and item description. The explanations are sentences extracted from user-generated reviews. We filtered out users with less than 20 interactions. The statistics data of the processed dataset is shown in Table 1.

4.2 Evaluation Metrics

For text quality, we use the USR (Unique Sentence Ratio). USR (Li et al., 2020) is defined as the ratio of unique sentences to the total number of sentences in the generated explanations, which captures lexical-level diversity by quantifying sentence repetition within the outputs. For factuality, we use FMR (Feature Matching Ratio) and FCR (Feature Coverage Ratio). FMR measures whether a generated explanation contains the feature in the ground-truth. FCR calculates the number of distinct features contained in all the generated explanations, divided by the total number of features in the whole dataset. API Cost (Cost) reports the total dollar cost required to complete a explanation generation and evaluation task using the agents. This metric considers API usage and emphasizes the importance of designing cost-effective agents, as noted by (Kapoor et al., 2024).

[1]https://amazon-reviews-2023.github.io
[2]https://www.tripadvisor.com
[3]https://www.yelp.com/dataset
[4]https://github.com/lileipisces/PETER

4.3 Baselines

We compare our model's performance on explanation generation against the following baselines:

- **NRT** (Li et al., 2017b) jointly predicts user-item ratings and produces concise textual explanations, conditioned on user and item IDs using a GRU-based decoder.
- **Att2Seq** (Dong et al., 2017a) introduces an attention-enhanced attribute-to-sequence model that generates explanations by encoding user and item attributes.
- **PETER** (Li et al., 2021) proposes a personalized Transformer-based architecture that formulates the explanation generation task as a language modeling problem, effectively capturing long-range dependencies in user-item interactions.
- **XRec** (Ma et al., 2024) leverages the language capabilities of LLMs and integrating collaborative signals to generate explanations.
- **LLMX** (Luo et al., 2024) employs LLMs directly to generate explanations following special instruction.
- **RefineX** (Zhang et al., 2025) refines initial explanations through a predefined trajectory during the inference stage to enhance their quality across multiple aspects.

4.4 Implementation Details

We use LightGCN as the recommendation model and obtain collaborative embeddings for users, and set $k = 20$ for user clustering. We use GPT-5.4-mini (Singh et al., 2026) for all experiments with the same hyperparameters. We set the number of user agents as $n \in \{100, 200, 500\}$ to verify the degree of alignment with real users. For explanation evaluation, we fix $n = 200$ with a maximum page number of 2 and 4 items per page. The explanation generation tool is LLMX. We employ the all-MiniLM-L6-v2 model from Sentence-Transformers for memory retrieval. All experiments were conducted on a server equipped with 3 $\times$ NVIDIA GeForce RTX 2080 Ti GPU. The experimental result is the average value of the two runs. Additional implementation details are provided in Appendix B.

Method	Amazon				Tripadvisor				Yelp			
	USR (%)	FMR (%)	FCR (%)	Cost ($)	USR (%)	FMR (%)	FCR (%)	Cost ($)	USR (%)	FMR (%)	FCR (%)	Cost ($)
ID-based												
NRT	22.42	9.18	6.67	–	21.54	6.83	14.05	–	19.88	6.95	9.97	–
Att2Seq	15.34	5.11	6.56	–	5.96	6.11	8.56	–	13.20	7.01	9.54	–
PETER	23.09	9.26	8.59	–	7.76	7.02	12.57	–	20.04	8.09	10.56	–
LLM-based												
XRec	95.18	9.17	7.82	–	94.32	6.04	8.13	–	93.24	6.79	9.63	–
LLMX	95.23	6.84	6.56	**2.51**	94.57	5.67	7.84	**2.55**	93.57	5.54	6.22	**2.48**
RefineX	95.19	8.69	8.04	8.77	93.29	6.24	8.75	9.20	95.32	6.98	10.06	9.09
Ours	**99.54**	**18.61**	**19.26**	4.46	**95.31**	**12.09**	**16.82**	3.89	**96.73**	**8.28**	**12.59**	3.97

Table 2: Performance comparison on three datasets. Best results are in **bold** and second-best are underlined.

1:m	Method	Accuracy	Precision	Recall	F1 Score
1:1	Ours	**0.66**	**0.66**	**0.67**	**0.65**
	Agent4Rec	0.64	0.64	0.66	0.64
1:2	Ours	0.64	**0.66**	0.57	0.59
	Agent4Rec	0.61	0.63	0.55	0.56
1:3	Ours	0.63	0.62	0.53	0.55
	Agent4Rec	0.61	0.59	0.55	0.51

Table 3: User Agent Taste Alignment Performance.

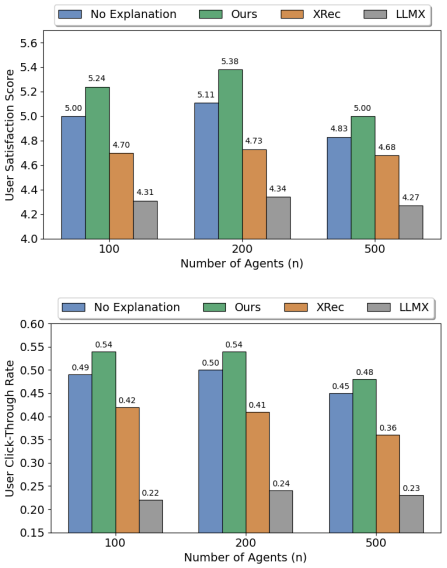


Figure 3: The impact of explanations on user behavior.

5 Results Analysis

5.1 Explanation Quality Assessment

As demonstrated in Table 2, our proposed explanation generation framework achieves highly competitive performance across all three scenarios. The experimental results indicate that LLM-based methods outperform traditional ID-based methods in terms of textual quality. Furthermore, our framework enhances the factual credibility of generated explanations, effectively mitigating hallucinations and producing accuracy explanations that align with users preferences. Our framework demonstrates relatively low API consumption while maintaining competitive performance. While RefineX rely on heuristic refinement and multi-round iterations, our framework employs a single-round revision process guided by precise feedback, thereby reducing computational overhead and shortening response latency.

5.2 User Alignment Evaluation

To investigate what extent can user agents truly simulate the user behaviors in recommender systems, we randomly recommended 20 items to the user in the validation mode on the Amazon dataset. The ratio between the items that the user has interacted with (i.e., $y_{u,i} = 1$) but were not utilized for profile initialization and the items that the user has not interacted with (i.e., $y_{u,i} = 0$) is set as $1 : m$. The user agent's responses are interpreted as binary preference signals, where higher scores indicate stronger alignment with the user's historical interaction distribution. Table 3 reports the results across various metrics. The user agent consistently identify items aligned with the preferences of users, and the increased collective preferences can further enhance the simulation accuracy of the user agent, further demonstrating the possibility of the generative agent simulating the preferences of the recommended users.

5.3 User Behavior Analysis

To investigate the impact of recommendation explanations on user behavior, we analyzed user agents operating in simulation mode on the Amazon dataset. The results presented in Figure 3 show that recommendations accompanied by our generated explanations achieve the highest levels of user satisfaction and click-through rate. These findings demonstrate that high-quality explanations can effectively enhance user satisfaction and increase users' willingness to interact with recommendation systems. In contrast, the explanations generated by XRec and LLMX lead to lower user en-

Base LLM	Strategy	USR (%)	FMR (%)	FCR (%)	Cost ($)
DeepSeek-V4-flash	Multi-Aspect	97.83	15.30	16.81	0.38
	Single-Aspect	95.79	14.59	15.43	0.53
GPT-5.4-mini	Multi-Aspect	99.54	18.61	19.26	4.46
	Single-Aspect	96.75	15.23	17.98	4.79

Table 4: Explanation Evaluation Results of Multi-Aspect and Single-Aspect Refinement.

Method	USR (%)	FMR (%)	FCR (%)	Cost ($)
Origin (DeepSeek)	97.83	15.30	16.81	0.38
w/o planner	96.36	12.59	13.14	0.27
w/o feedback & planner	96.12	10.61	11.32	0.20
Origin (GPT)	99.54	18.61	19.26	4.46
w/o planner	99.56	14.27	15.84	3.23
w/o feedback & planner	98.95	11.67	13.78	2.60

Table 5: Ablation studies

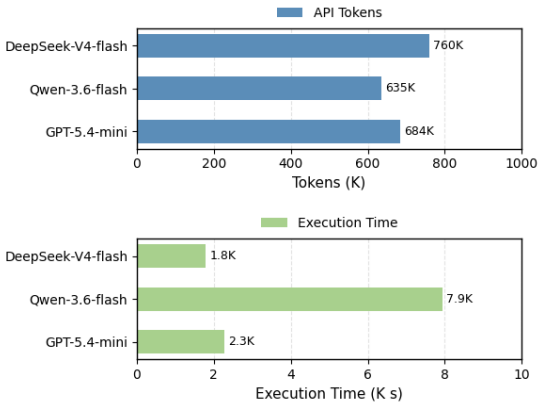


Figure 4: Efficiency of different LLMs.

gagement, highlighting the importance of generating user-centric explanations that better align with users' preferences and needs.

5.4 Analysis of the Refinement Strategies

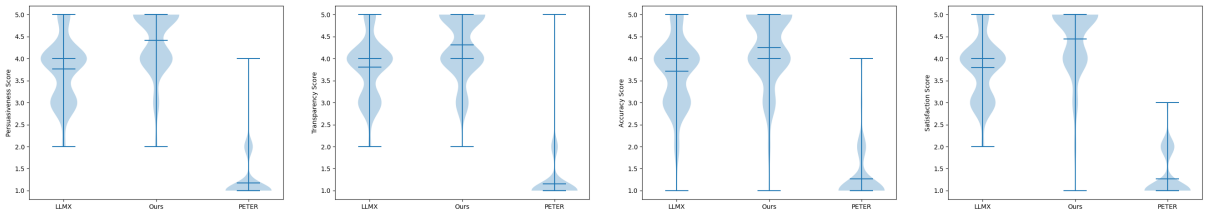
To investigate the effectiveness of different refinement approaches, we implement two strategies for explanation refinement on Amazon dataset. We systematically evaluate these strategies to examine the trade-off between efficiency and explanation quality. **Multi-Aspect** batches all items that require refinement into a single LLM prompt: each item is provided with its profile, the current explanation, the full refinement plan, and the feedback. Then the model returns revised explanations for all such items in one structured response. **Single-Aspect** applies a sequential, multi-specialist pipeline: for each item that enters refinement, four specialist prompts take the text produced by the previous stage as input and perform a targeted refinement. It can be seen from the result that Multi-Aspect can provide better explanations at a lower cost. Modifications among multiple agents might lead to excessive alterations in the explanation, causing it to deviate from the user's requirements.

5.5 Ablation studies

i) **w/o feedback** removes the feedback from the peer agent. The refiner directly modify the explanation. ii) **w/o planner** directly makes modifications by refiner based on the scores and feedback content. The results on Amazon dataset is shown in 5. Removing the feedback module causes the refiner to modify explanations solely based on the user profile. Although this setting reduces token consumption, it significantly degrades performance across textual evaluation metrics, resulting in inferior overall performance. In contrast, when the planner is removed, the refiner can still revise explanations according to the feedback. However, without the planner to interpret and organize feedback information, the refiner's ability to perform effective modifications is weakened. These findings suggest that the two key components work together in a complementary and synergistic manner, contributing to the superior performance of our full model.

5.6 Efficiency Analysis

Considering that token consumption is directly proportional to cost, response time is also an important factor in practical applications. We used different LLMs to assess the efficiency on Tripadvisor dataset, and the results are shown in Figure 4. Deepseek-V4-flash (DeepSeek-AI, 2026) achieves the best results in terms of execution time. Specifically, the average response time for generating explanations for a single user is approximately 9 seconds, demonstrating its strong potential for practical deployment that require both low latency and cost-effectiveness.



(a) (b) (c) (d)

Figure 5: Distribution of Human Evaluation of explanations on Amazon Dataset.

User Profile	I prefer animated films that combine elements of fantasy and adventure as well as being suitable for the whole family to watch. I have a penchant for black comedies, psychological thrillers, and satirical humor. I enjoy narrative styles that are challenging, unconventional, and often have a dark humorous flavor.
Recommended Item	The Mysterious River
LLMX	This dramatic crime thriller should appeal if you enjoy darker, more serious stories with psychological tension. Its strong historical rating also suggests it has been well received by many viewers.
PETER	This movie is very exciting.
Agent4ExpRec (Ours)	If you like psychological thrillers with a dark, serious mood, this crime drama should be a strong match. Its emotional intensity and moral tension give it the kind of suspenseful atmosphere that fits your taste for darker storytelling.
Ground-Truth	[Extract from the reviews by (Zhang et al., 2014b)] This movie had better suspense and a better storyline than most dramatic thrillers out there now.

Table 6: Example of the generated explanation by our proposed Agent4ExpRec and the baselines.

5.7 Human Evaluation

To further assess whether the generated explanations effectively align with user preferences and enhance user satisfaction, we conducted a human evaluation involving fifty participants from both the recommender systems community and other research domains, thereby better reflecting the diversity of real-world users. Figure 5 presents a comparison between our method and the baseline approaches across four aspects. Our method consistently achieves higher scores than PETER and LLMX in all aspects, demonstrating its superior ability to align with the preferences of real users.

5.8 Case Study

We provide an example in Table 6 to illustrate explanations generated by different methods. Figure 6 provides a intuitive understanding of our explanation refinement framework.

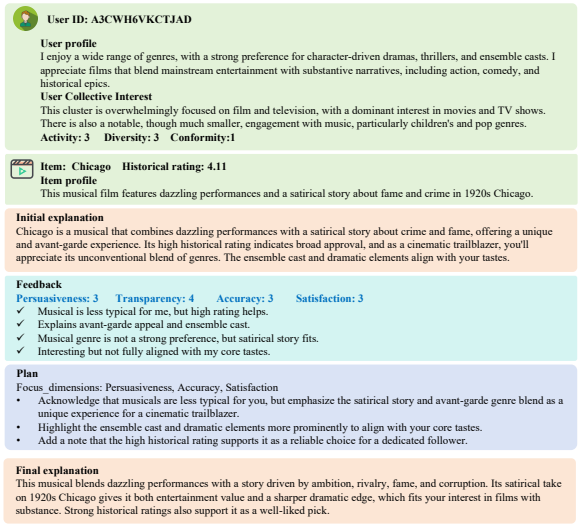


Figure 6: Case Study.

6 Conclusion

In this paper, we propose Agent4ExpRec, a novel framework for recommendation explanation generation and evaluation. We design a feedback-driven refinement mechanism to generate user-centered explanations. Furthermore, by leveraging LLM-powered agents to simulate users, we investigate user satisfaction and behavior under different explanations. Extensive experiments demonstrate that our framework achieves significant advantages in both efficiency and textual quality of explanations. We also analyzed the alignment between the agents and simulated users, offering a promising research direction for ER. In the future, we plan to extend Agent4ExpRec to broader recommendation domains and incorporate richer user interaction signals, such as dwell time and sequential browsing patterns, to further close the gap between simulated and real-world user behavior.

Limitations

Our research primarily focuses on user-oriented post-hoc explanations and therefore does not extensively investigate the underlying recommendation mechanisms, which are equally important for improving model transparency and providing actionable insights for system developers. In addition, the action space of the simulated users remains relatively limited and cannot yet fully capture the complexity and diversity of real-world user behaviors. As the framework relies on LLM-based user simulation, it may also inherit biases or behavioral inconsistencies introduced by LLMs, potentially leading to discrepancies between simulated and real user feedback. Future work will aim to incorporate a broader range of user interaction patterns and behavioral signals, enabling a more comprehensive analysis of how explainable recommendations influence user decision-making processes. We also plan to further enhance the realism, robustness, and reliability of the user simulation framework, thereby improving the credibility and practical applicability of simulation-based evaluation in explainable recommendation systems.

Ethics Statement

In this paper, we explore the effectiveness of leverage LLM-based agents on explainable recommendation. We are committed to using and modify datasets only for research purposes and not for commercial exploitation. The datasets we use are all publicly available and do not contain any personal privacy information. Additionally, our proposed approach does not generate harmful content or raise ethical issues.

References

Justin Chen, Archiki Prasad, Swarnadeep Saha, Elias Stengel-Eskin, and Mohit Bansal. 2025. MAgICoRe: Multi-agent, iterative, coarse-to-fine refinement for reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 32663–32686, Suzhou, China. Association for Computational Linguistics.

DeepSeek-AI. 2026. Deepseek-v4: Towards highly efficient million-token context intelligence.

Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J. Smola, Jing Jiang, and Chong Wang. 2014. 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*, KDD '14, page 193–202, New York, NY, USA. Association for Computing Machinery.

Li Dong, Shaohan Huang, Furu Wei, Maria Lapata, Ming Zhou, and Ke Xu. 2017a. Learning to generate product reviews from attributes. In *15th EACL 2017 Software Demonstrations*, pages 623–632. Association for Computational Linguistics.

Li Dong, Shaohan Huang, Furu Wei, Mirella Lapata, Ming Zhou, and Ke Xu. 2017b. Learning to generate product reviews from attributes. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 623–632, Valencia, Spain. Association for Computational Linguistics.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. 2023. Alpacafarm: A simulation framework for methods that learn from human feedback. In *Advances in Neural Information Processing Systems*, volume 36, pages 30039–30069. Curran Associates, Inc.

Xinshun Feng, Mingzhe Liu, Yi Qiao, Tongyu Zhu, Leilei Sun, and Shuai Wang. 2026. Behavior tokens speak louder: Disentangled explainable recommendation with behavior vocabulary. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(25):21092–21100.

Fatih Gedikli, Dietmar Jannach, and Mouzhi Ge. 2014. How should i explain? a comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies*, 72(4):367–382.

Xu Huang, Jianxun Lian, Yuxuan Lei, Jing Yao, Defu Lian, and Xing Xie. 2025. Recommender ai agent: Integrating large language models for interactive recommendations. *ACM Trans. Inf. Syst.*, 43(4).

Sayash Kapoor, Benedikt Stroebl, Zachary S. Siegel, Nitya Nadgir, and Arvind Narayanan. 2024. Ai agents that matter. *Preprint*, arXiv:2407.01502.

Jieyong Kim, Hyunseo Kim, Hyunjin Cho, SeongKu Kang, Buru Chang, Jinyoung Yeo, and Dongha Lee. 2025. Review-driven personalized preference reasoning with large language models for recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, page 1697–1706, New York, NY, USA. Association for Computing Machinery.

Yuxuan Lei, Jianxun Lian, Jing Yao, Xu Huang, Defu Lian, and Xing Xie. 2024. Recexplainer: Aligning large language models for explaining recommendation models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, page 1530–1541, New York, NY, USA. Association for Computing Machinery.

Lei Li, Yongfeng Zhang, and Li Chen. 2020. Generate neural template explanations for recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, page 755–764, New York, NY, USA. Association for Computing Machinery.

Lei Li, Yongfeng Zhang, and Li Chen. 2021. Personalized transformer for explainable recommendation. In *ACL*.

Lei Li, Yongfeng Zhang, and Li Chen. 2023a. Personalized prompt learning for explainable recommendation. *ACM Transactions on Information Systems (TOIS)*.

Lei Li, Yongfeng Zhang, and Li Chen. 2023b. Personalized prompt learning for explainable recommendation. *ACM Trans. Inf. Syst.*, 41(4).

Piji Li, Zihao Wang, Zhaochun Ren, Lidong Bing, and Wai Lam. 2017a. Neural rating regression with abstractive tips generation for recommendation. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, page 345–354, New York, NY, USA. Association for Computing Machinery.

Piji Li, Zihao Wang, Zhaochun Ren, Lidong Bing, and Wai Lam. 2017b. Neural rating regression with abstractive tips generation for recommendation. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 345–354.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Hongyu Lu, Weizhi Ma, Yifan Wang, Min Zhang, Xiang Wang, Yiqun Liu, Tat-Seng Chua, and Shaoping Ma. 2023. User perception of recommendation explanation: Are your explanations what users need? *ACM Trans. Inf. Syst.*, 41(2).

Yucong Luo, Mingyue Cheng, Hao Zhang, Junyu Lu, and Enhong Chen. 2024. Unlocking the potential of large language models for explainable recommendations. In *Database Systems for Advanced Applications*, pages 286–303, Singapore. Springer Nature Singapore.

Qiyao Ma, Xubin Ren, and Chao Huang. 2024. XRec: Large language models for explainable recommendation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 391–402, Miami, Florida, USA. Association for Computational Linguistics.

Wenyu Mao, Jiancan Wu, Weijian Chen, Chongming Gao, Xiang Wang, and Xiangnan He. 2025. Reinforced prompt personalization for recommendation with large language models. *ACM Trans. Inf. Syst.*, 43(3).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, UIST '23, New York, NY, USA. Association for Computing Machinery.

Jakub Raczyński, Mateusz Lango, and Jerzy Stefanowski. 2023. The problem of coherence in natural language explanations of recommendations. In *ECAI*.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, and 467 others. 2026. Openai gpt-5 system card. *Preprint*, arXiv:2601.03267.

Jiakai Tang, Jingsen Zhang, Zihang Tian, Xueyang Feng, Lei Wang, and Xu Chen. 2025. Explainable recommendation with simulated human feedback. *ACM Trans. Inf. Syst.*, 44(1).

Zhefan Wang, Yuanqing Yu, Wendi Zheng, Weizhi Ma, and Min Zhang. 2024. Macrec: A multi-agent collaboration framework for recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 2024, page 2760–2764. ACM.

Mengyuan Yang, Mengying Zhu, Yan Wang, Linxun Chen, Yilei Zhao, Xiuyuan Wang, Bing Han, Xiaolin Zheng, and Jianwei Yin. 2024. Fine-tuning large language model based explainable recommendation with explainable quality reward. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 9250–9259.

An Zhang, Yuxin Chen, Leheng Sheng, Xiang Wang, and Tat-Seng Chua. 2024a. On generative agents in recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 1807–1817, New York, NY, USA. Association for Computing Machinery.

Jingsen Zhang, Zihang Tian, Xueyang Feng, Xu Chen, and Chong Chen. 2025. Enhancing recommendation explanations through user-centric refinement. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 8177–8191, Suzhou, China. Association for Computational Linguistics.

Xiaoyu Zhang, Yishan Li, Jiayin Wang, Bowen Sun, Weizhi Ma, Peijie Sun, and Min Zhang. 2024b. Large language models as evaluators for recommendation explanations. In *Proceedings of the 18th ACM Conference on Recommender Systems*, RecSys '24, page 33–42, New York, NY, USA. Association for Computing Machinery.

Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. 2014a. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, page 83–92, New York, NY, USA. Association for Computing Machinery.

Yongfeng Zhang, Haochen Zhang, Min Zhang, Yiqun Liu, and Shaoping Ma. 2014b. Do users rate or review? boost phrase-level sentiment labeling with review-level sentiment classification. In *SIGIR*.

A Algorithm

Algorithm 1 details the user-centric explanation refinement pipeline. Algorithm 2 presents the overall procedure of the Agent4Rec simulation framework.

Algorithm 1 User-Centric Explanation Refinement

Input: User profile $\mathcal{P}_u$, Recommendation list $\mathcal{L}_t$, Item information $\{(\text{title}_i, \text{summary}_i)\}_{i\in\mathcal{L}_t}$
Output: Refined explanations $\{e_i^*\}_{i\in\mathcal{L}_t}$

for each item $i \in \mathcal{L}_t$ **do**
  $e_i^0 \leftarrow \text{LLMX}(\text{title}_i, \text{summary}_i, \mathcal{P}_u)$
  for each aspect $a \in A$ **do**
    $(s_i^a, \rho_i^a) \leftarrow \text{Peer}(a, e_i^0, \mathcal{P}_u)$
  end for
  $\pi_i \leftarrow \text{Planner}\big(e_i^0, \{(s_i^a, \rho_i^a)\}_{a\in A}, \mathcal{P}_u\big)$
  if $\pi_i \neq \varnothing$ **then**
    $e_i^* \leftarrow \text{Refiner}(e_i^0, \pi_i, \mathcal{P}_u, \text{summary}_i)$
  else
    $e_i^* \leftarrow e_i^0$
  end if
  Store $\{e_i^0, \{(s_i^a, \rho_i^a)\}_{a\in A}, \pi_i, e_i^*\}$ to memory
end for
return $\{e_i^*\}_{i\in\mathcal{L}_t}$

B Implementation Details

B.1 Execution Mode

Simulation. For each agent, the system sequentially presents the recommendation list along with its corresponding explanatory text. The user agent judges the relevance of items on each page to personal preferences, selects items to browse further, provides ratings and feedback, and decides whether to continue browsing or exit based on overall experience. This process proceeds until the maximum page limit is reached or the user terminates the session voluntarily.

Validation. To conduct quantitative assessment, we construct for each agent a candidate set consisting of positive samples from its validation or test set as well as unseen negative samples, and restrict the user agent to make a single selection from this constrained set. After the user interaction, during the post-interview stage, we asked the users to provide subjective evaluations of the explanations in the aforementioned four aspects. This mode focuses

Algorithm 2 Explainable Recommendation Simulation

Input: Dataset $\mathcal{D}$; recommender $\mathcal{R}$; user set $\mathcal{U}$; item set $\mathcal{I}$; page size K; max steps T; memory interval R
Output: Simulation metrics $\mathcal{M}$; user behavior logs $\{\mathcal{B}_u\}_{u\in\mathcal{U}}$

Train recommender $\mathcal{R}$ on $\mathcal{D}$ ▷ LightGCN
for $u \in \mathcal{U}$ **do**
  Construct user profile $\mathcal{P}_u$ from historical interactions and preferences
end for
for $u \in \mathcal{U}$ **do**
  $t \leftarrow 1$, *active* $\leftarrow$ **true**
  while $t \leq T$ and *active* **do**
    $\mathcal{L}_t \leftarrow \text{TopK}(\mathcal{R}(u, \mathcal{I}), K)$
    $\{e_i^*\}_{i\in\mathcal{L}_t} \leftarrow \textsc{Explanation}(u, \mathcal{L}_t, \mathcal{P}_u)$
    $\mathcal{C}_u \leftarrow \text{Query}(\mathcal{M}_u, \mathcal{L}_t)$
    $\boldsymbol{p} \leftarrow [\mathcal{P}_u, \mathcal{L}_t, \mathcal{C}_u, \{e_i^*\}]$
    for $i \in \mathcal{L}_t$ **do**
      $(r_i, f_i) \leftarrow User(\boldsymbol{p})$
    end for
    $o_t \leftarrow \text{Summarise}(\{(r_i, f_i)\}_{i\in\mathcal{L}_t})$
    $\mathcal{M}_u \leftarrow \mathcal{M}_u \cup \{\text{Embed}(o_t)\}$
    Append $(\mathcal{L}_t, \{r_i\}, \{f_i\})$ to $\mathcal{B}_u$
    decision $\leftarrow User(\mathcal{P}_u, \mathcal{C}_u, o_t)$
    if *decision* $=$ EXIT **then**
      active $\leftarrow$ **false**
    else
      $t \leftarrow t + 1$
    end if
  end while
end for
Store $\mathcal{M}$ and $\{\mathcal{B}_u\}_{u\in\mathcal{U}}$ to memory

on offline indicators for user alignment assessment and explanation subjective evaluation.

B.2 Temperature

The initial explanation generation is deployed by prompt templates from studies (Luo et al., 2024). In the refine pipeline, from the initial explanation generation, feedback, planning to refine/stepwise experts, all steps continue to use 0.25. The user agent's taste/satisfaction/viewing experience, etc. is weighted by 0.25. The extraction of the generated user profile, item profile, and collective interest uses 0.2.

B.3 Evaluation Metrics

USR measures the diversity of generated explanations at the sentence level, given by Eq.(4).

$$\text{USR} = |E|/N_g \tag{4}$$

where E denotes the set of generated unique explanations, N_g is the number of total explanations. FCR (Feature Coverage Ratio) is defined as the number of distinct features mentioned across all generated explanations divided by the total number of features in the dataset:

$$FCR = N_e/|\mathcal{F}| \tag{5}$$

where $\mathcal{F}$ is the collection of all features in the dataset, and N_e is the number of distinct features shown in the generated explanations. FMR (Feature Matching Ratio) measures whether the generated explanations contain the features present in the ground truth:

$$FMR = \frac{1}{|\mathcal{F}|} \sum_{u,i} \delta(g_{u,i} \in e_{u,i}) \tag{6}$$

where $e_{u,i}$ is the generated explanations for a user-item pair, $g_{u,i}$ is the given feature, and $\delta(x) = 1$ if x is true and $\delta(x) = 0$ otherwise.

B.4 Data Split for User Profile Construction

To prevent data leakage, user profiles are constructed exclusively from the training split. Interactions in the validation and test splits are excluded from profile initialization and are used solely as ground-truth labels for alignment evaluation. Specifically, the dataset is partitioned at the interaction level into mutually exclusive training, validation, and test splits with a ratio of 4:3:3. Both user profile construction and recommendation model training are performed using only the training split. The validation and test splits are treated strictly as held-out labels ($y_{u,i} = 1$) for evaluation. The ground-truth labels used in validation mode are also derived exclusively from the validation split, thereby ensuring a strict separation between profile construction and evaluation data.

C Human Evaluation

We presented user profile and explanations to evaluators for further evaluation from four aspects. We also asked the evaluators the following questions: i). Taking into account your evaluation of the above explanatory text, what aspect of the explanation do you focus on the most? Are there any additional aspects that should also be considered? The summary of our responses is shown in Figure 7. ii). Please fill in your suggestions for improving the quality of the explanatory text. The summary of the answers is shown as follows:

(1) **Personalization & User Alignment**: Users emphasize that explanations should explicitly incorporate their historical preferences and provide clear mappings between movie attributes and user interests.

(2)**Length & Conciseness**: Many users suggest that overly long explanations reduce readability, recommending more concise outputs (e.g., 2-3 sentences).

(3) **Factual & Correctness**: Some responses highlight issues of incorrect or misleading information, which negatively affects credibility.

(4) **Structure & Readability**: Users prefer more structured explanations (e.g., bullet-style or clearly separated points) for easier comprehension. Visual elements such as lists and bold can be used to highlight key points and avoid excessive text length.

(5) **Style & Naturalness**: Feedback also suggests that explanations should avoid template-like expressions and adopt a more natural and conversational tone.

Overall, these findings indicate that users value explanations that are both personalized and concise, while maintaining factual accuracy and natural expression. These suggestions provide practical directions for future system refinement.

D Further Analysis

D.1 Peer agent as Evaluator

To observe the correlation between the peer agent as the evaluator and the real user, we calculated the

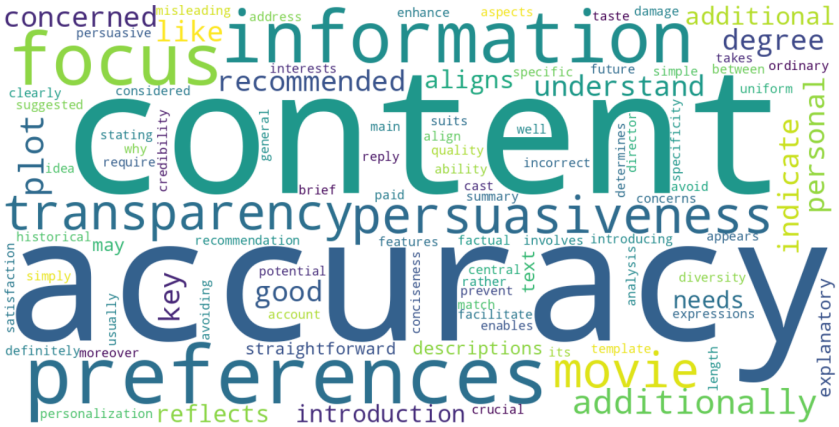
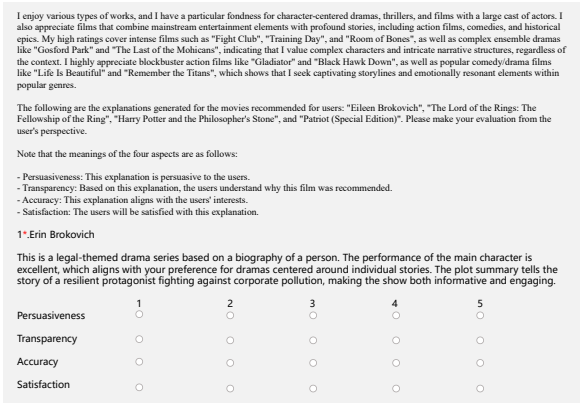


Figure 7: The word cloud of aspects that users pay attention to in the explanations on the Amazon dataset.

	Amazon			TripAdvisor			Yelp		
Method	**B-1**	**R-1**	**BERT**	**B-1**	**R-1**	**BERT**	**B-1**	**R-1**	**BERT**
PETER	**0.15**	0.20	0.24	**0.16**	0.21	0.24	**0.12**	0.18	0.24
LLMX	0.05	0.17	0.14	0.07	0.18	0.13	0.06	0.14	0.12
Ours	0.06	**0.22**	**0.30**	0.11	**0.25**	**0.31**	0.09	**0.21**	**0.30**

Table 7: Textual similarity between generated explanations and ground truth across Amazon, TripAdvisor, and Yelp datasets. B-1, R-1, and BERT denote BLEU-1, ROUGE-1, and BERT Score, respectively.


I enjoy various types of works, and I have a particular fondness for character-centered dramas, thrillers, and films with a large cast of actors. I also appreciate films that combine mainstream entertainment elements with profound stories, including action films, comedies, and historical epics. My high ratings cover intense films such as "Fight Club", "Training Day", and "Room of Bones", as well as complex ensemble dramas like "Gosford Park" and "The Last of the Mohicans", indicating that I value complex characters and intricate narrative structures, regardless of the context. I highly appreciate blockbuster action films like "Gladiator" and "Black Hawk Down", as well as popular comedy/drama films like "Life Is Beautiful" and "Remember the Titans", which shows that I seek captivating storylines and emotionally resonant elements within popular genres.

The following are the explanations generated for the movies recommended for users: "Eileen Brokovich", "The Lord of the Rings: The Fellowship of the Ring", "Harry Potter and the Philosopher's Stone", and "Patriot (Special Edition)". Please make your evaluation from the user's perspective.

Note that the meanings of the four aspects are as follows:

- Persuasiveness: This explanation is persuasive to the users.
- Transparency: Based on this explanation, the users understand why this film was recommended.
- Accuracy: This explanation aligns with the users' interests.
- Satisfaction: The users will be satisfied with this explanation.

1*.Erin Brokovich

This is a legal-themed drama series based on a biography of a person. The performance of the main character is excellent, which aligns with your preference for dramas centered around individual stories. The plot summary tells the story of a resilient protagonist fighting against corporate pollution, making the show both informative and engaging.

	1	2	3	4	5
Persuasiveness	○	○	○	○	○
Transparency	○	○	○	○	○
Accuracy	○	○	○	○	○
Satisfaction	○	○	○	○	○

Figure 8: Human evaluation interface.

Pearson correlation coefficient between the user agent's rating and the human rating for initial explanations. The results are shown in Table 8. We used the average score given by fifty human evaluators for 60 explanations as the human rating. Different LLMs perform differently in aligning with real user preferences, but most show a positive correlation. This indicates that it is reasonable for us to use agents to simulate real users' feedback on explanations.

D.2 Evaluation of Textual Similarity

We assess textual similarity between generated explanations and user reviews using common metrics,

Base Model	**Per.**	**Tran.**	**Acc.**	**Sat.**
DeepSeek-V4-flash	0.34	0.26	0.27	0.29
Qwen-3.6-flash	0.30	0.21	0.31	0.26
GPT-5.4-mini	0.45	0.40	0.56	0.54

Table 8: Pearson correlation between the ratings given by the peer agents and human users for explanations.

including BLEU, ROUGE, and BERT Score. As shown in Table 7, our method consistently outperforms all baselines in ROUGE-1 and BERT Score, demonstrating superior content coverage and semantic alignment with user preferences. It should be noted that textual similarity is not the primary optimization objective of our framework. Although PETER achieves higher BLEU-1 scores, this improvement is largely driven by token-level n-gram overlap with reference texts, which may primarily capture frequent expressions instead of deeper semantic alignment.

E Prompts

This section introduces the prompts used by the agents within our Agent4ExpRec. Through the collaboration of these specialized agents, the framework is able to generate explanations that are more personalized, coherent, and aligned with user preferences.

User Agent Recommendation Interaction

You excel at role-playing. Picture yourself as a user exploring a movie recommendation system. You have the following social traits:
Your activity trait is described as: {activity_dsc}
Your conformity trait is described as: {conformity_dsc}
Your diversity trait is described as: {diversity_dsc}
Beyond that, **your movie tastes are:** {taste}
And your rating tendency is: {high_rating}
Collective interest of your user group: {collective_interest_text}
If anything above conflicts with your stated tastes, traits, or rating tendency, follow your individual preferences — they take precedence.
Relevant context from your memory: {formatted_memories}

User Agent Movie-watching Decision

PAGE {current_page}
{recommended_items_str}
Some items may include a system field "Why recommended:" with a short justification;
treat it as the recommender's stated reason (you may agree or disagree).
Firstly, determine which movies align with your taste and which do not:
MOVIE: [name]; ALIGN: [yes or no]; REASON: [brief reason]
Secondly, decide how many to watch based on activity & diversity traits:
NUM: [number]; WATCH: [all movie names]; REASON: [brief reason];
Thirdly, rate the watched movies 1–5 considering conformity trait:
MOVIE: [name]; RATING: [1-5]; REASON: [why]; FEELING: [aftermath sentence];

User Agent Leave Decision

You excel at role-playing. Picture yourself as a user exploring a movie recommendation system.
Your activity trait: {activity_dsc}
Now you are in Page {current_page}. You may get tired with the increase of pages.
(above 2 pages = a little tired; above 4 pages = very tired)
Relevant context from your memory: {formatted_memories}
Generate an overall feeling based on memory and activity trait:
If positive → **POSITIVE: [reason]**
If negative → **NEGATIVE: [reason]**
Assess fatigue level. Decide whether to continue or exit:
To leave → **[EXIT]**; Reason: [brief reason]
To continue → **[NEXT]**; Reason: [brief reason]

Figure 9: Prompts for User Agent.

Peer Agent Prompt

System Prompt

You output strict JSON only. Be concise.

User Prompt

You judge how a USER with the profile below would react to each recommendation explanation.
User profile:
{user_profile}
Aspects (score each 1-5, 5=strongly agree with the aspect statement):
{dim_lines}
For each item, output scores and one short rationale per aspect.
Items:
{blocks}
Return ONLY valid JSON:

```
{"items":[
  {"item_id": <int>,
    "Persuasiveness": {"score": <1-5>, "rationale": "<short>"},
    "Transparency": {"score": <1-5>, "rationale": "<short>"},
    "Accuracy": {"score": <1-5>, "rationale": "<short>"},
    "Satisfaction": {"score": <1-5>, "rationale": "<short>"}
  }, ...
]}
```

Figure 10: Prompts for Peer Agent.

Planner Agent Prompt

System Prompt

You output strict JSON only.

User Prompt

You are a refinement planner. For each item, read the user profile, the current explanation, and the four-dimension feedback.
Prioritize aspects with score <= 3. Produce an ordered list of concrete refinement steps (what to change in the wording).

User profile:
{user_profile}

Per-item JSON lines:
{blob}

Return ONLY valid JSON:

```
{"plans":[
  {"item_id": <int>,
    "focus_dimensions": ["<names among
Persuasiveness,Transparency,Accuracy,Satisfaction>"],
    "steps": ["<step1>", "<step2>", ...]
  }, ...
]}
```

Figure 11: Prompts for Planner Agent.

Refiner Agent Prompt (Multi-Aspect)

System Prompt

You output strict JSON only.

User Prompt

Rewrite each item's recommendation explanation following its refine_plan.
Stay consistent with the movie summary; do not invent facts.
Keep 2-4 sentences, suitable to show to the end user next to the item.
User profile:
{user_profile}
Per-item JSON:

```
{
 "item_id": <int>,
 "title": <str>,
 "summary": <movie summary, max 800 chars>,
 "original_explanation": <initial explanation>,
 "refine_plan": {
  "focus_dimensions": [...],
  "steps": ["<step1>", ...]
 },
 "feedback_summary": { <peer scores> }
}
```

Refiner Agent Prompt (Single-Aspect)

System Prompt

Persuasiveness
You are the Persuasiveness specialist. Improve the explanation so it feels compelling and well-supported: clear reasons to watch, confident but honest wording, no empty flattery. Ground every claim in the movie summary; do not invent cast/plot facts not grounded in each movie summary. Output strict JSON only with a single key refined_text.
Transparency
You are the Transparency specialist. Improve the explanation so the user sees *why* this movie was recommended: explicit links between their profile/tastes and concrete aspects drawn from the movie summary. Avoid vague boilerplate. Output strict JSON only with key refined_text.
Accuracy
You are the Accuracy specialist. Improve the explanation so it matches the movie summary and the user profile: fix overclaims, remove invented facts, soften or qualify weak links. Output strict JSON only with key refined_text.
Satisfaction
You are the Satisfaction specialist. Polish for clarity, tone, and helpfulness: concise 2-4 sentences, respectful and easy to scan, consistent with prior edits. Output strict JSON only with key refined_text.

User Prompt

Refine this recommendation explanation in your specialty pass.
Other specialists will edit later; do not undo their future work—focus on your dimension only.
User profile: {user_profile}
Item details: Title / Summary / Rating
Current explanation (after prior specialist passes if any): {current_expl}
Your dimension: {dim} (Criterion: {ASPECT_DEFINITIONS[dim]})
Prior evaluator score for this dimension: {dim_score}/5
Evaluator note: {dim_rationale}
Rewrite to 2-4 sentences, consistent with the movie summary.
Refine the sentences, consistent with the {summary_label}.

Figure 12: Prompts for Refiner Agent.