FLEX-Bench: Recommendation Explanation Benchmark with Feature-Level User Preferences

Anonymous ACL submission

Abstract

Evaluating recommendation explanations is difficult because explanation quality depends on user preferences and is inherently personalized. Traditional text similarity metrics such as BLEU and ROUGE fail to capture the contextual nuances of recommendation explanations, and recent LLM-as-a-judge approaches still lack effective personalization strategies, relying mainly on indirect preference signals such as demographics or item-level information. To address this gap, we propose FLEX, an evaluation framework that extracts feature-level user-specific preferences from historical reviews and augments them with complementary item-specific features flagged by other users, producing a rich profile for personalized automatic evaluation. Our experiments show that FLEX consistently outperforms existing evaluation methods, achieving higher agreement with human labels. We further introduce a benchmark for recommendation explanation methods under our proposed evaluation framework. We benchmark ten heterogeneous systems across three datasets including MovieLens, Amazon Video Games, and Yelp, surfacing performance variance under different domains and evaluation dimensions.

1 Introduction

Recommender systems are widely used to help users navigate large catalogs of items, services, and media. As these systems increasingly influence everyday choices, explanation becomes an important complement to recommendation accuracy. Explanations can help users understand why an item is recommended, support informed decision making, and improve the perceived transparency of the system (Herlocker et al., 2000; Bilgic and Mooney, 2005). In practical settings, however, a recommendation explanation is not useful merely because it is fluent. It must also reflect the user's interests, present relevant context, and communicate the recommendation rationale in a form that users find informative and persuasive.

Evaluating these explanations is both critical and difficult. Because explanations serve to align recommendations with user needs, their assessment must capture several qualitative dimensions, such as whether they are persuasive, transparent, and personally relevant. Yet explanation quality is inherently personalized; what convinces one user may fail for another user with different preferences, prior experiences, and expectations (Tintarev and Masthoff, 2012; Lu et al., 2023). Unlike many text-generation tasks, recommendation explanations rarely have a fixed objective target, and judgments often reflect subjective user preference rather than a single notion of correctness.

Evaluation Paradigms. Because explanation quality is subjective and user-dependent, evaluation has traditionally depended on manual annotation. Human studies, where users or experts rate explanations along key quality dimensions have long been the gold standard (Tintarev and Masthoff, 2012; Lu et al., 2023). However, manual annotation is expensive, slow, and difficult to scale across multiple systems and datasets.

The rise of large language model (LLM) judges opens a new opportunity: LLMs can evaluate open-ended text according to explicit criteria, enabling automatic, scalable, and consistent assessments that align with human preferences. This also makes it feasible to benchmark heterogeneous explanation algorithms under a unified evaluation setting, where a fixed LLM evaluator serves as a proxy for human judgment. In this work, we propose an automatic pointwise evaluation framework, where an LLM judge assigns an absolute score to each explanation, enhancing scalability and reproducibility of explanation evaluation.

Personalized Evaluation of Recommendation Explanations. Effective personalization in expla-

Table 1: Illustrative example of a user's ground-truth review paired with an explanation utilizing preference features and a generic explanation (no user preferences). The feature-based explanation explicitly references the user's stated interests, whereas the non-personalized explanation only reports aggregate popularity.

User Review (Ground Truth)	Explanation w/ Feature	Explanation w/o Feature
Spielberg's films are a **must-see**. Just look at the cast; even before watching, I'd already rate it 4 stars. American films are also experts in **war movies**.	It's a **war movie**, a **very classic**, and highly rated.	Among users with similar interests to you, 93.0% of those who have seen this movie liked it.

nation evaluation requires user representations that capture what a user values. Existing approaches predominantly rely on indirect signals such as demographic attributes or coarse behavioral traces like user purchase histories. While these signals are easy to obtain, they offer only a shallow view of user preferences and often fail to capture the nuanced reasons a user likes or dislikes a specific item. Evaluators conditioned on such representations may therefore rely on stereotypical patterns rather than explicit user-specific signals, limiting their ability to judge whether an explanation truly aligns with the target user.

Table 1 illustrates the personalization gap: the generic template-style explanation reports aggregate statistics that do not directly reference the user's specific interest in war films, while the more expressive feature-informed explanation directly echoes the user's own language and preferences. An evaluator which does not account for explicit user-expressed preferences would be unable to draw a direct link connecting the strong overlap between the feature-informed explanation and the user's opinion.

We address this gap by constructing feature-level preference profiles directly extracted from a user's historical reviews and augmenting them with target-item features flagged by other users. The resulting augmented profile captures both user-specific tastes and item-specific context, serving as the conditioning signal for our pointwise LLM evaluator, FLEX.

Contributions. This paper makes three main contributions.

- We propose FLEX, a recommendation explanation evaluation framework which uses feature-level preferences extracted from historical user reviews to construct user profiles for LLM-based evaluation.
- We conduct extensive experiments evaluating FLEX against competing evaluators and alternative profiling strategies. FLEX achieves the highest correlation with human judgments across all four explanation dimensions, outperforming baseline personalization settings and existing evaluators.
- We introduce a benchmark for recommendation explanation methods across three datasets: MovieLens, Amazon Video Games, and Yelp. We evaluate ten heterogeneous explanation methods with an easily extensible pipeline based on our automatic evaluator FLEX.

2 Related Work

2.1 Recommendation Explanation

Work on explainable recommendation spans early context-based rationales and more recent neural and LLM-based generators. This evolution was accompanied with an equivalent shift from simple to more complex explanation text, impacting the efficiency and efficacy of downstream evaluation tasks.

Early collaborative filtering explanations exposed neighborhood or rating context to improve transparency and user acceptance (Herlocker et al., 2000). Later work broadened the design space, showing that different explanation forms reveal different kinds of context and serve multiple user-facing goals (Tintarev and Masthoff, 2010, 2015). This early body of work established the theoretical importance od explanations and the quality dimensions they should target.

Neural approaches first enabled fluent review-like generation from recommendation signals (Dong et al., 2017; Li et al., 2017, 2020). Later work strengthened personalization through user and item conditioning, retrieval augmentation, and prompt- or LLM-based generation, improving alignment with user preferences (Li et al., 2021b;

Hada et al., 2021; Xie et al., 2022; Cheng et al., 2023; Raczynski et al., 2023; Li et al., 2023; Lei et al., 2024; Ma et al., 2024; Li et al., 2025b; Yang et al., 2025). While these methods produce increasingly sophisticated explanations, the resulting heterogeneity and varying user-alignment make systematic, user-grounded evaluation increasingly urgent, a challenge that motivates our work.

2.2 Evaluation of Recommendation Explanations

Evaluation has long been recognized as a central challenge for explainable recommendation (Tintarev and Masthoff, 2010, 2012; Chen et al., 2022). Foundational work argues that explanations should be evaluated against multiple goals rather than a single metric, and later work shows that these goals can be partially conflicting (Tintarev and Masthoff, 2010; Balog and Radlinski, 2020). This motivates multi-dimensional evaluation of user-facing quality.

Existing protocols include user studies, which are difficult to scale, and automatic metrics (Tintarev and Masthoff, 2012; Chen et al., 2022). Automatic evaluation methods often rely on general text similarity metrics such as BLEU and ROUGE. Previous work also proposes task-specific similarity metrics, such as FMR (Li et al., 2020), and ranking-oriented resources such as EXTRA (Li et al., 2021a). However, these approaches usually depend on evaluating against a reference text, which is unavailable in realistic scenarios, or relative explanation comparisons rather than direct modeling of user judgments.

The rise of LLMs enables a new paradigm of automatic evaluation which can be practical and personalized. Personalization is central to explanation evaluation because different users may respond differently to the same explanation (Tintarev and Masthoff, 2012; Lu et al., 2023). For example, (Lu et al., 2023) show that user satisfaction is closely tied to whether an explanation matches the information the user actually needs. Yet prior LLM-based automatic evaluators of recommendation explanations typically represent user context only indirectly, mainly through demographics and aggregate statistics or purchase histories (Zhang et al., 2024; Li et al., 2025a). Our work differs by representing user preference explicitly through feature-level preferences derived from historical reviews.

2.3 LLM-as-a-Judge and Benchmarks

The use of LLMs as judges has been studied systematically in general text evaluation, where strong language models were shown to correlate well with human preferences in open-ended settings (Zheng et al., 2023). Subsequent work found that structured criteria and rubric-based prompting can further improve human alignment (Liu et al., 2023). These developments motivate the use of LLM judges for recommendation explanations, where the evaluation target is subjective and multi-dimensional.

Recent work has begun to adapt this paradigm to recommendation explanations. Zhang et al. (Zhang et al., 2024) show that strong LLM judges can correlate well with user ratings and often outperform reference-based metrics in this task. ALERT (Li et al., 2025a) extends this line by introducing a broader recommendation explanation benchmark and pairwise LLM-based evaluators. At the same time, prior work shows that LLM judges remain imperfect: they can exhibit positional and verbosity biases (Wang et al., 2024), favor their own generations (Panickssery et al., 2024), and become unreliable on difficult comparisons (Tan et al., 2024). Reference-based baselines also remain important points of comparison, but commonly used metrics such as BLEU can have limited generalizability (Reiter, 2018), while other metrics such as BERTScore (Zhang et al., 2019) provide a stronger semantic alternative rather than a user-grounded one. Relative to prior work, our study focuses on automatic pointwise evaluation and uses obtainable feature-level user preference profiles rather than indirect user preference context.

3 FLEX: Feature-Level User Preference-based Explanation Evaluator

3.1 Problem Definition

We consider the task of automatic evaluation of recommendation explanations in a pointwise setting. Let $\mathcal{U}$ denote the set of users and $\mathcal{I}$ the set of items. For a user $u \in \mathcal{U}$ and an item $i \in \mathcal{I}$, a recommendation system produces an explanation $e_{u,i}$ in natural language. The goal of the evaluator is to predict the score that user u would assign to $e_{u,i}$ on one of several explanation-quality dimensions.

Let $\mathcal{D} = \{\mathrm{P}, \mathrm{T}, \mathrm{IA}, \mathrm{S}\}$ denote the set of evaluation dimensions. For each $d \in \mathcal{D}$, let $y_{u,i}^{(d)} \in \{1, \ldots, K\}$ be the human rating for the user-item

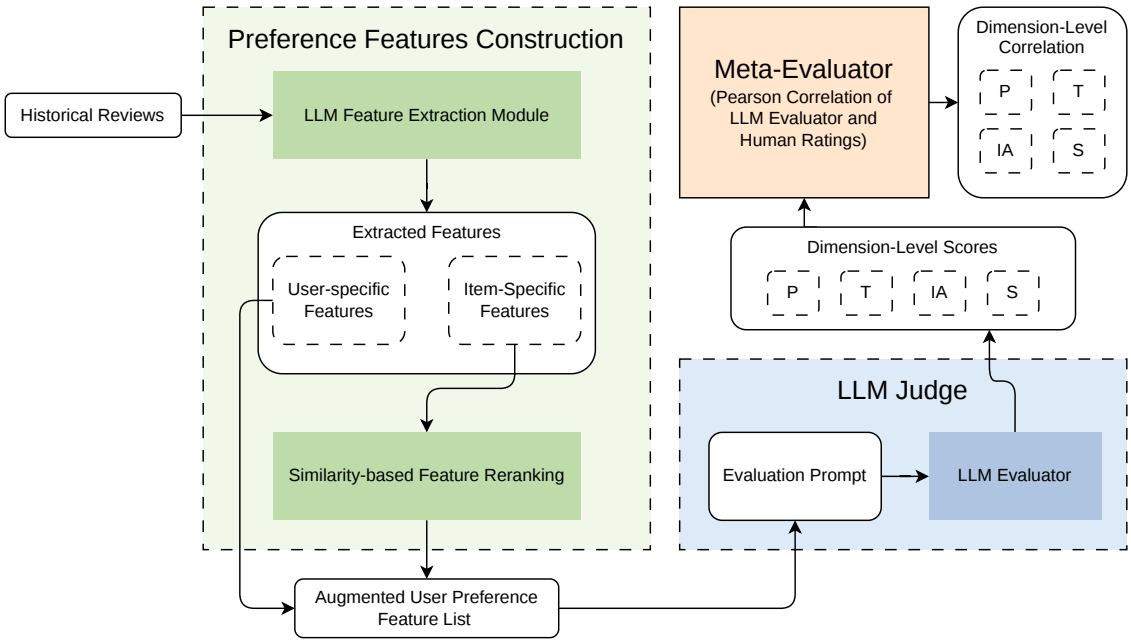

Figure 1: Overview of the augmented-features evaluation pipeline. Evaluation dimensions are abbreviated as P = persuasiveness, T = transparency, IA = interest accuracy, and S = satisfaction.

pair (u, i) on dimension d, where $K = 5$ in our experiments. The evaluator predicts a score $\hat{y}_{u,i}^{(d)}$ by conditioning on the explanation text and, optionally, user-side information:

$$\hat{y}_{u,i}^{(d)} = f_\theta\big(e_{u,i}, \phi(u, i)\big), \tag{1}$$

where $\phi(u, i)$ denotes the user profile supplied to the evaluator and θ denotes the parameters of the judge, including the underlying LLM and prompt design. When no user context is used, $\phi(u, i) = \emptyset$. Evaluation quality is measured by the agreement between $\hat{y}_{u,i}^{(d)}$ and $y_{u,i}^{(d)}$ over a set of user-item pairs, and we use Pearson correlation as the primary metric.

The quality of $\phi(u, i)$ is critical in explanation evaluation. If the profile only weakly reflects user preference, the judge must infer the user's taste from incomplete or indirect evidence. We therefore define $\phi(u, i)$ using an augmented feature-level preference representation. The base profile is derived from the user's historical reviews, excluding the target item to avoid leakage. We then augment this profile with up to 5 item-specific features extracted from other users' historical reviews for the same target item, excluding the target user. These additional features are selected to be semantically dissimilar to the target user's features, so that the evaluator receives both user-specific preference context and complementary target item-specific context.

3.2 FLeX Evaluator Framework

FLeX operates in three stages, as summarized in Figure 1: user-preference construction, pointwise LLM judging, and meta-evaluation. For each user, we first extract a structured feature-level preference profile from their historical reviews by prompting an LLM to produce concise aspect–sentiment pairs. This yields a compact representation of what the user likes, dislikes, or values most in their previous interactions. To capture item-specific context that may be absent from the user's own history, we augment this base profile with up to five semantically dissimilar features gathered from other users' historical reviews of the target item, producing a profile that combines both user-specific preferences and item-specific features flagged as critical by other users. The augmented profile, together with the target item and the candidate explanation, is then passed to the LLM judge. The judge follows a fixed rubric that defines four evaluation dimensions—persuasiveness (P), transparency (T), interest accuracy (IA), and satisfaction (S)—and returns an integer score from 1 to 5 for each. Finally, the meta-evaluation stage exists as an optional post-evaluation step if validating the evaluation results against a gold standard is needed. In this final stage,

Table 2: Pearson correlation (%) of explanation evaluator scores with different user profiling strategies between LLM-Judge and human labels. Evaluation dimensions are abbreviated as P = persuasiveness, T = transparency, IA = interest accuracy, and S = satisfaction. Best per column within each LLM is shown in **bold**; second-best is underlined.

LLM	User Profile	P	T	IA	S	Average
GPT-4o-mini	None	28.16	23.70	22.32	24.74	24.73
	Demographics	30.87	19.50	21.56	24.38	24.08
	Review History	32.04	27.43	26.95	28.71	28.78
	Features	33.49	29.22	29.41	28.21	30.08
	Augmented Features	**35.02**	**30.60**	**30.47**	**30.52**	**31.65**
Llama 3.1 8B	None	16.80	2.81	8.74	12.69	10.26
	Demographics	20.46	8.02	15.84	**19.67**	16.00
	Review History	10.21	3.03	7.60	12.63	8.37
	Features	**28.42**	**11.64**	15.76	17.44	**18.32**
	Augmented Features	25.77	7.11	**18.36**	17.79	17.26

we compute dimension-level Pearson correlation between the predicted scores and human ratings to quantify how well the evaluator tracks user judgments.

The four user-centered evaluation dimensions we follow are established in prior work on recommendation explanation goals and user perception (Tintarev and Masthoff, 2010; Lu et al., 2023; Zhang et al., 2024) and are defined as follows:

- *Persuasiveness*: "This explanation is convincing to me."
- *Transparency*: "Based on this explanation, I understand why this movie is recommended."
- *Interest accuracy*: "This explanation is consistent with my interests."
- *Satisfaction*: "I am satisfied with this explanation."

Together, these dimensions consider the comprehensive quality of an explanation and are sufficiently distinct to motivate multi-dimensional evaluation rather than a single scalar score.

4 Performance of FLEX

In this section, we conduct experiments to test the quality of our proposed evaluation framework FLEX. We use the MovieLens dataset version introduced by Lu et al. (Lu et al., 2023), which is a movie domain collection containing human-labeled explanations; it comprises 39 users, 140 items, and 2,535 explanations.

4.1 Performance Comparison of Different User Profiling Strategies

Prior LLM-based approaches have conditioned their evaluation on demographic attributes, aggregate statistics, or item-level information (Zhang et al., 2024; Li et al., 2025a), but these signals characterize user identity or actions without explicitly articulating tastes. We compare five profiling strategies to measure the impact of personalization techniques: *None* (no user context), *Demographics* (user identity and aggregate activity statistics), *Review History* (raw historical user reviews), *Features* (structured aspect–sentiment features extracted from the user's historical reviews), and *Augmented Features* (user features augmented with up to five semantically dissimilar target-item features from other users' historical reviews). The *Features* and *Augmented Features* settings are based on our proposed Preference Feature Construction module in Figure 1. In all personalized settings, the target user-item pair is excluded from the user-history portion of the profile to avoid information leakage.

Table 2 reports Pearson correlation with human ratings under different user profiling strategies for two judge backbones. Two patterns are notable. First, the augmented-features profile yields the most consistent gains for GPT-4o-mini. Relative to the non-augmented feature profile, it improves all four dimensions and raises the average correlation from 30.08 to 31.65. This suggests that supplementing user-history features with a small set of semantically dissimilar target item-specific features provides additional context that a strong judge can

Table 3: Meta-evaluation of candidate automatic evaluators on MovieLens. Pearson correlations (%) with human labels on the four explanation dimensions are reported. Best per column is shown in **bold**; second-best is underlined.

Evaluator	Persuasiveness	Transparency	Interest Accuracy	Satisfaction	Average
BLEU-1	5.66	3.95	5.34	6.17	5.28
ROUGE-1	4.04	2.43	4.21	4.18	3.72
FMR	4.74	4.99	8.44	10.68	7.21
BERTScore	21.03	19.66	23.40	24.90	22.25
LLMAsEval	23.92	11.58	15.76	21.15	18.10
ALERT	22.20	19.49	17.89	23.86	20.86
FLeX (Ours)	**35.02**	**30.60**	**30.47**	**30.52**	**31.65**

use effectively.

Second, the benefits of augmentation are more mixed for Llama 3.1 8B. The augmented-features profile improves interest accuracy and remains competitive on satisfaction, but it underperforms the non-augmented feature profile on persuasiveness, transparency, and average correlation. A plausible explanation is that the weaker evaluator is less able to exploit the richer augmented profile and is more sensitive to the extra contextual complexity. In contrast, raw review history again performs poorly, indicating that compact feature-based representations remain preferable to long unstructured histories.

Demographic profiling yields mixed results. It improves some dimensions relative to the no-profile setting, but it does not provide reliable gains across all dimensions. This result suggests that identity-level cues can help with coarse personalization, yet they remain insufficient for fine-grained judgments about whether an explanation truly matches the user's interests. Overall, Table 2 indicates that explicit preference representation is a more effective personalization strategy than either demographics or raw review history, and that our feature augmentation step is especially beneficial when the underlying judge is strong enough to use the additional item-specific context.

4.2 Overall Performance of FLeX Evaluator

Before adopting FLeX as the benchmark evaluator for generation comparison, we compare its alignment with human ratings against a range of existing automatic evaluators to verify its suitability. We consider two broad classes.

Reference-based evaluators. BLEU-1 and ROUGE-1 compare a system explanation against a reference text using n-gram overlap. BERTScore replaces surface overlap with contextual embedding similarity. We also include FMR (Feature Matching Ratio), which measures whether the explanation covers reference-side features (Li et al., 2020).

LLM judges. LLMAsEval (Zhang et al., 2024) is a pointwise LLM judge conditioned on demographic user information; we run it on GPT-4o-mini as the backbone. ALERT (Li et al., 2025a) is an LLM-based evaluator built on a fine-tuned Llama3-8B backbone. Our proposed evaluator FLeX uses GPT-4o-mini as the backbone, and conditions the judge on our proposed augmented-features profile that combines historical user preference features with up to five target item-specific features from other users's historical reviews.

Results. Table 3 reports Pearson correlation with human ratings for all candidate evaluators. Three conclusions follow. First, traditional reference-based metrics are weak indicators of user judgment in this task. BLEU-1, ROUGE-1, and FMR all show low correlations across dimensions, consistent with the view that overlap on a lexical level is a poor proxy for user perception of explanation quality. Second, BERTScore substantially outperforms lexical overlap metrics but remains consistently below FLeX on all four dimensions, indicating that semantic reference similarity cannot fully substitute for explicit modeling of personalized user preference. Third, LLM-based judging benefits from personalization, but the type of personalization matters. Both LLMAsEval and ALERT outperform traditional reference-based baselines, yet neither matches FLeX, which achieves the highest correlation on all four dimensions. These results confirm that our proposed feature-level evaluator FLeX provides a more effective signal for personalized pointwise evaluation compared to other

Table 4: Benchmark results for recommendation explanation generation across MovieLens, Amazon Video Games, and Yelp. Scores are mean FLEX judgments on a 1–5 scale. Columns correspond to persuasiveness (P), transparency (T), interest accuracy (IA), and satisfaction (S).

		MovieLens				Amazon Video Games				Yelp			
Arch.	System	P	T	IA	S	P	T	IA	S	P	T	IA	S
Neural Models	Att2Seq	1.63	1.46	1.66	1.61	1.00	1.00	1.00	1.00	1.54	1.45	1.62	1.52
	NRT	1.20	1.10	1.23	1.18	1.47	1.25	1.46	1.37	1.57	1.52	1.61	1.55
	NETE	1.01	1.00	1.03	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Transformer	PETER	1.50	1.41	1.52	1.45	1.32	1.24	1.32	1.29	2.10	1.97	2.17	2.03
	CER	1.12	1.10	1.17	1.11	1.53	1.39	1.53	1.46	1.90	1.84	2.00	1.86
	ERRA	1.00	1.00	1.00	1.00	1.26	1.13	1.25	1.21	1.60	1.56	1.64	1.58
LLM	ReXPlug	2.12	2.25	1.88	1.82	2.06	2.06	1.67	1.70	2.50	2.43	2.13	2.24
	PEPLER	1.05	1.01	1.05	1.05	1.59	1.39	1.62	1.52	2.06	1.94	2.12	1.97
	RecExp.	2.45	3.04	2.14	2.12	3.76	4.48	3.85	3.68	3.87	4.52	3.94	3.86
	MAPLE	1.09	1.07	1.10	1.08	1.39	1.13	1.42	1.27	3.28	3.22	3.66	3.24

methods, making FLEX a suitable evaluator for our benchmark in Section 5.

5 FLeX-Bench Benchmark of Recommendation Explanation Methods

5.1 Benchmark Design

The benchmark evaluates the quality of individual recommendation explanations produced by heterogeneous generation systems under the same pointwise evaluation protocol defined in Section 3. We use the augmented-features profile and GPT-4o-mini backbone, assigning integer scores for persuasiveness (P), transparency (T), interest accuracy (IA), and satisfaction (S) on the 1–5 scale as described in Section 3.2. Our benchmark applies our proposed evaluator FLEX across all systems and datasets, producing dimension-level scores for each explanation under a consistent scoring protocol.

In this benchmark, we use three sampled datasets. The MovieLens dataset used in Section 4, with 310 unique user-item pairs. The Amazon Video Games dataset, a review-based corpus with 1,117 users, 1,004 items, and 1,117 unique user-item pairs. The Yelp dataset, with 2,000 users, 1,897 items, and 1,117 unique user-item pairs. For the benchmark experiments in Section 5, we split the MovieLens dataset into equal halves (50% train / 50% test), while the Yelp dataset provides 28,467 training samples and the Amazon Video Games dataset provides 6,145 training samples.

The benchmark covers ten explanation generation systems across three architecture families. The neural models are Att2Seq, NRT, and NETE (Dong et al., 2017; Li et al., 2017, 2020), early neural generators that condition on user and item recommendation signals. The Transformer-based systems are PETER, CER, and ERRA (Li et al., 2021b; Raczynski et al., 2023; Cheng et al., 2023), models that extend Transformer sequence modeling with personalization, coherence objectives, or retrieved context. The LLM-based systems are ReXPlug, PEPLER, RecExplainer, and MAPLE (Hada et al., 2021; Li et al., 2023; Lei et al., 2024; Yang et al., 2025), ranging from plug-and-play pretrained language model adaptation to alignment- and aspect-controlled generation strategies.

5.2 Results and Analysis

Table 4 reports mean FLEX scores on the 1–5 scale for all systems across MovieLens, Amazon Video Games, and Yelp, with systems grouped by architecture family. Figure 2 displays the dimension-averaged scores for each system by dataset.

RecExplainer is the strongest system across all three datasets, achieving averages of 2.44, 3.94, and 4.05 on MovieLens, Amazon Video Games, and Yelp respectively. Dimension-level scores reveal additional structure: RecExplainer leads most strongly on transparency, reaching 3.04 on MovieLens and 4.48 and 4.52 on Amazon Video Games and Yelp, while its interest accuracy and satisfaction scores are comparatively lower on MovieLens. MAPLE shows a contrasting pattern, scoring near

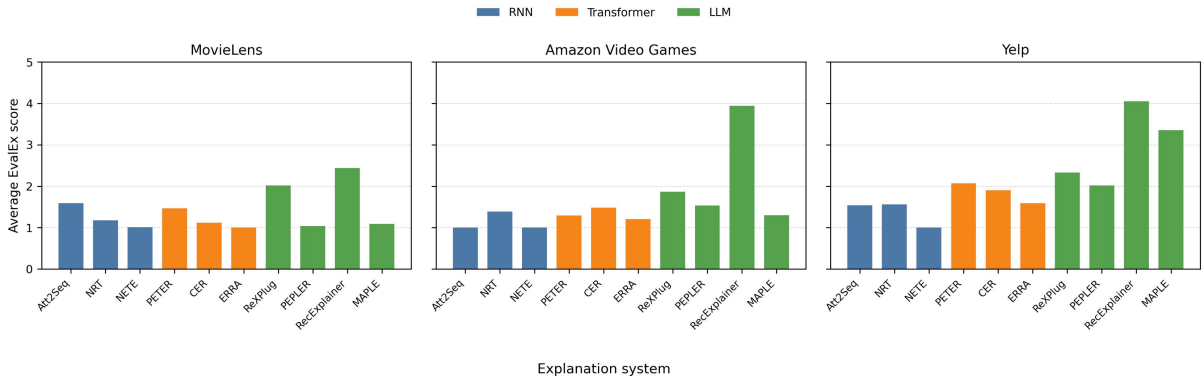

Figure 2: Dimension-averaged FLEX scores for each explanation generation system across MovieLens, Amazon Video Games, and Yelp.

the floor on MovieLens and Amazon Video Games but rising substantially on Yelp to an average of 3.35. More broadly, LLM-based generators appear to be more promising with a higher performance ceiling relative to neural and Transformer-based systems.

Domain also influences benchmark scores. Yelp yields the highest average scores across systems, while MovieLens and Amazon Video Games remain lower and similar to each other. Yelp provides the richest historical preference context, giving the judge a stronger basis for assessing user-explanation alignment. Systems such as RecExplainer that consistently perform strongly across all three datasets offer more convincing evidence of generalizability across different domains than systems whose scores vary across datasets.

6 Conclusion

This work addresses the challenge of evaluating recommendation explanations in a personalized, scalable, and user-grounded manner. We introduced FLEX, a personalized evaluator that judges recommendation explanations by conditioning an LLM on augmented feature-level user preference profiles. By extracting structured aspect–sentiment features from a user's historical reviews and enriching them with target-item context from other users, FLEX captures explicit user preferences which provide relevant signals during evaluation.

Our experiments show that feature-based profiles yield stronger alignment with human judgments than other personalization baselines, and that target-item feature augmentation provides consistent additional gains when the judge backbone is sufficiently strong. Compared to existing reference-based metrics and prior LLM judges, FLEX achieves the highest Pearson correlation with human ratings across all four evaluation dimensions in our experimental setting. We further presented a cross-domain benchmark covering ten explanation systems under our evaluation framework, providing an extensible unified protocol for comparing heterogeneous architectures.

7 Limitations

The scope of meta-evaluation in this work is partly constrained by the scarcity of publicly available datasets with user-grounded explanation ratings; additional such resources would enable more comprehensive study of personalized explanation evaluation across domains and user populations. As with any LLM-based evaluation framework, FLEX may also inherit biases associated with the choice of judge backbone, including tendencies to favor verbosity, fluency, and self-consistency. In real evaluation settings, these tendencies would be difficult to fully disentangle from genuine explanation quality without a controlled experiment studying the quality of the chosen judge backbone. Finally, while the benchmark currently covers ten heterogeneous systems, a broader range of explanation algorithms remains to be evaluated; the pipeline is designed to be easily extensible, and we encourage future work to broaden system coverage.

References

Krisztian Balog and Filip Radlinski. 2020. Measuring recommendation explanation quality: The conflicting goals of explanations. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pages 329–338.

Mustafa Bilgic and Raymond J Mooney. 2005. Explaining recommendations: Satisfaction vs. promotion. In *Beyond personalization workshop, IUI*, volume 5, page 153.

Xu Chen, Yongfeng Zhang, and Ji-Rong Wen. 2022. Measuring" why" in recommender systems: A comprehensive survey on the evaluation of explainable recommendation. *arXiv preprint arXiv:2202.06466*.

Hao Cheng, Shuo Wang, Wensheng Lu, Wei Zhang, Mingyang Zhou, Kezhong Lu, and Hao Liao. 2023. Explainable recommendation with personalized review retrieval and aspect learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 51–64.

Li Dong, Shaohan Huang, Furu Wei, Mirella Lapata, Ming Zhou, and Ke Xu. 2017. 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.

Deepesh V Hada, Vijaikumar M, and Shirish K Shevade. 2021. Rexplug: Explainable recommendation using plug-and-play language model. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 81–91.

Jonathan L Herlocker, Joseph A Konstan, and John Riedl. 2000. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, pages 241–250.

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*, pages 1530–1541.

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*, pages 755–764.

Lei Li, Yongfeng Zhang, and Li Chen. 2021a. Extra: Explanation ranking datasets for explainable recommendation. In *Proceedings of the 44th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 2463–2469.

Lei Li, Yongfeng Zhang, and Li Chen. 2021b. Personalized transformer for explainable recommendation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4947–4957.

Lei Li, Yongfeng Zhang, and Li Chen. 2023. Personalized prompt learning for explainable recommendation. *ACM Transactions on Information Systems*, 41(4):1–26.

Piji Li, Zihao Wang, Zhaochun Ren, Lidong Bing, and Wai Lam. 2017. 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.

Yichuan Li, Xinyang Zhang, Chenwei Zhang, Mao Li, Tianyi Liu, Pei Chen, Yifan Gao, Kyumin Lee, Kaize Ding, Zhengyang Wang, and 1 others. 2025a. Alert: An llm-powered benchmark for automatic evaluation of recommendation explanations. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2704–2719.

Yuhan Li, Xinni Zhang, Linhao Luo, Heng Chang, Yuxiang Ren, Irwin King, and Jia Li. 2025b. G-refer: Graph retrieval-augmented large language model for explainable recommendation. In *Proceedings of the ACM Web Conference 2025*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 2511–2522.

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 Transactions on Information Systems*, 41(2):1–31.

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.

Arjun Panickssery, Samuel Bowman, and Shi Feng. 2024. Llm evaluators recognize and favor their own generations. *Advances in Neural Information Processing Systems*, 37:68772–68802.

Jakub Raczynski, Mateusz Lango, and Jerzy Stefanowski. 2023. The problem of coherence in natural language explanations of recommendations. *arXiv preprint arXiv:2312.11356*.

Ehud Reiter. 2018. A structured review of the validity of bleu. *Computational Linguistics*, 44(3):393–401.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Y Tang, Alejandro Cuadron, Chenguang Wang, Raluca Ada Popa, and Ion Stoica. 2024. Judgebench: A benchmark for evaluating llm-based judges. *arXiv preprint arXiv:2410.12784*.

Nava Tintarev and Judith Masthoff. 2010. Designing and evaluating explanations for recommender systems. In *Recommender systems handbook*, pages 479–510. Springer.

Nava Tintarev and Judith Masthoff. 2012. Evaluating the effectiveness of explanations for recommender systems: Methodological issues and empirical studies on the impact of personalization. *User Modeling and User-Adapted Interaction*, 22(4):399–439.

Nava Tintarev and Judith Masthoff. 2015. Explaining recommendations: Design and evaluation. In *Recommender systems handbook*, pages 353–382. Springer.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and 1 others. 2024. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9440–9450.

Zhouhang Xie, Sameer Singh, Julian McAuley, and Bodhisattwa Prasad Majumder. 2022. Factual and informative review generation for explainable recommendation. *arXiv preprint arXiv:2209.12613*.

Ching-Wen Yang, Zhi-Quan Feng, Ying-Jia Lin, Che Wei Chen, Kun-da Wu, Hao Xu, Jui-Feng Yao, and Hung-Yu Kao. 2025. Maple: Enhancing review generation with multi-aspect prompt learning in explainable recommendation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 31803–31821.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Xiaoyu Zhang, Yishan Li, Jiayin Wang, Bowen Sun, Weizhi Ma, Peijie Sun, and Min Zhang. 2024. Large language models as evaluators for recommendation explanations. In *Proceedings of the 18th ACM Conference on Recommender Systems*, pages 33–42.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

A Prompt Templates

This appendix reproduces the main prompts used in our pipeline. We include the feature extraction prompt used to construct user preference features from historical user reviews (also called self-explanations), and the evaluation prompt used by the LLM judge during pointwise scoring.

A.1 Feature Extraction Prompt

This prompt is used with GPT-4o-mini to extract structured user preference features from a user's historical reviews. The first part is the system prompt that defines the extraction task and output schema, and the second part is the user prompt template that inserts the review text.

System prompt.

```
You extract concise, canonical user preference
↪  features from a self-explanation.
Your goal is to reflect the user's actual views
↪  as a human would summarize them.

Return strictly valid JSON with shape:
{
  "features": [ { "category": str, "feature":
  ↪  str, "polarity": -1|0|1 }, ... ]
}.

Core principles:
- Prioritize features that are salient,
↪  emphasized, or central to the user's opinion.
↪  If a feature feels minor or incidental, omit
↪  it.
- Extract only opinion-bearing preferences or
↪  aversions, not neutral descriptions.
- You may extract implied preferences ONLY when
↪  they are clearly supported by evaluative
↪  language, contrasts, or comparisons in the
↪  text.
- Do NOT infer preferences that require external
↪  assumptions or common knowledge.

Feature guidelines:
- `feature` must be short, precise,
↪  self-contained, and meaningful on its own.
- Use wording that stays as close as possible to
↪  the user's original phrasing; avoid
↪  unnecessary abstraction.
- If a preference has scope or conditions, encode
↪  them directly in the feature text (e.g.,
↪  "slow pacing in dramas").
- `category` is optional context (e.g., "plot",
↪  "color", "genre"); the feature must remain
↪  understandable without it.
- `polarity`: +1 if the user expresses liking or
↪  preference, -1 if dislike or avoidance, 0 if
↪  neutral or unclear.
- Preserve the language of the input; do NOT
↪  translate.
- Make sure that any mentions of item-related
↪  preferences are captured.
- Avoid duplicates, opposites, or complementary
↪  features. Do not add new preferences.
- Extract only what another human would
↪  reasonably conclude about the user's tastes.

Examples:
Self-explanation: ", , "
Return:
{ "features": [
  {"category": "color", "feature": "",
  ↪  "polarity": 1 },
  {"category": "color", "feature": "",
  ↪  "polarity": -1 }
] }
```

```
Self-explanation: "I like cheap thrillers and I
↪  avoid long, slow dramas."
Return:
{ "features": [
  {"category": "price", "feature": "cheap
  ↪  thrillers", "polarity": 1 },
  {"category": "genre", "feature": "long, slow
  ↪  dramas", "polarity": -1 }
] }
```

User prompt template.

```
Self-explanation text (user's own words):
'{text}'
Extract the most prominent features the user
↪  expresses as specified above. Preserve the
↪  input language for returned features. Return
↪  only valid JSON and nothing else.
```

A.2 Evaluation Prompt

This prompt is used by the LLM judge to score a recommendation explanation on the four evaluation dimensions. It conditions the evaluator on the user's feature summary and requests structured JSON output.

```
You are an expert judge for movie recommendation
↪  explanations. Given the user profile, and
↪  rating criteria, rate the given explanations
↪  on the dimensions of accuracy,
↪  persuasiveness, satisfaction, and
↪  transparency.

Accuracy is defined as: ""This explanation is
↪  consistent with my interests.""
Persuasiveness is defined as: ""This explanation
↪  is convincing to me.""
Satisfaction is defined as: ""I am satisfied with
↪  this explanation.""
Transparecy is defined as: ""Based on this
↪  explanation, I understand why this movie is
↪  recommended.""

Assess those dimensions with integers between 1-5
↪  , where 1 indicates strongly disagree and 5
↪  indicates strongly agree.

Instructions: Include the user's high-level
↪  feature preferences when making your
↪  judgement.

User Feature Summary:
The user previously expressed interest in the
↪  following features with polarity 1
↪  (positive), -1 (negative), or 0 (neutral)
↪  opinion in relation to other items:
↪  {user_info}

Explanation to rate:
Movie: {item_name}
Explanation: {explanation}

Return JSON: {
  ""accuracy_score"": <int>,
  ""persuasiveness_score"": <int>,
  ""satisfaction_score"": <int>,
  ""transparency_score"": <int>,
  ""rationale"": <string>
}

DO NOT include any other text or formatting
↪  outside the JSON object. DO NOT inlude any
↪  header or footer tags such as ```JSON.
```