Beyond a Single Story: Meta-Reviewing Sparse and Incomplete User-generated Contents for Recommendation

Hongren Wang[1,♥], **Tianjun Wei**[2*], **Yingpeng Du, Jie Zhang, Yin-Leng Theng**
Nanyang Technological University, Singapore
[1]hongren001@e.ntu.edu.sg [2]tjwei2-c@my.cityu.edu.hk

Abstract

Data sparsity remains a long-standing challenge in recommender systems, and it becomes more severe for methods relying on user-generated content (UGC) such as textual reviews, which capture fine-grained preferences but require more user efforts to produce. As a result, UGC exhibits (1) **missing reviews**, where interactions lack any review, and (2) **incomplete reviews**, where available reviews cover only a subset of relevant attributes. Existing approaches often overlook these UGC-specific issues, leading to degraded accuracy. Motivated by **meta-review** in academic peer review, we propose MOSAIC (**M**eta-review **O**n **S**parse **A**nd **I**ncomplete user-generated **C**ontent), which constructs a meta-review for each target user by aggregating attribute-sentiment evidence from neighbor users' reviews. A multi-gate mixture-of-experts (MMoE) architecture jointly optimizes rating prediction and meta-review attribute-sentiment prediction, while an attention module personalizes the aggregated meta-review signals to each target user, yielding both refined rating predictions and attribute-level explanations. Experiments on four real-world datasets demonstrate that MOSAIC consistently outperforms state-of-the-art baselines in both recommendation accuracy and explanation quality, mitigating UGC sparsity and incompleteness while delivering consistent gains for users with **limited interaction history**. †

1 Introduction

The rapidly expanding volume of digital information and user data has made recommender systems (RSs) an indispensable tool for personalizing user experience and driving business growth across various Internet and web platforms (Li et al., 2024; Gheewala et al., 2025). Yet a foundational challenge has persisted at the core of these systems: data sparsity (Gunathilaka et al., 2025; Gheewala et al., 2025). Most prior work alleviates the **sparsity of behavioral feedback**, such as clicks and ratings (Cheng et al., 2023b; Lin et al., 2024; Liu et al., 2023), but such signals capture only coarse-grained preferences and miss user reasoning and context. User-generated content (UGC), such as reviews, is semantically rich and complements numerical ratings (Isinkaye et al., 2015). However, since writing reviews requires considerable user effort, UGC itself is even sparser than behavioral feedback, making the **sparsity of UGC a more severe and critical problem**.

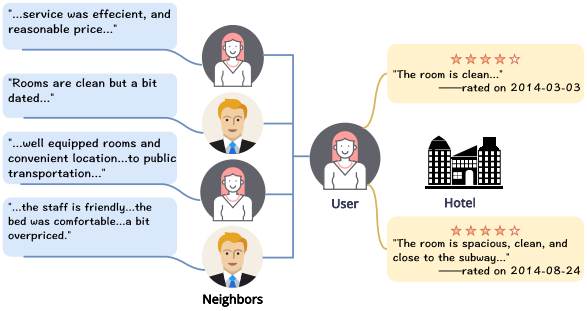

Figure 1: A *TripAdvisor* example illustrating review incompleteness: the same user mentions different attributes in different reviews of the same hotel. Aggregating neighbor reviews as a meta-review compensates for the missing attributes.

To combat this, advanced Natural Language Processing (NLP) techniques have been applied to extract fine-grained information from reviews (Wei et al., 2023; Liu et al., 2025; Liao et al., 2025). Early aspect-based methods (Dong et al., 2020; Shuai et al., 2022) identified opinions linked to product attributes but often struggled with noisy and unstructured text. Recent work (Zhang et al.,

*Corresponding author.
♥Collaborative Initiative, Interdisciplinary Graduate Programme, Nanyang Technological University, Singapore; College of Computing and Data Science, Nanyang Technological University, Singapore.
†Code is available on GitHub: https://github.com/wanghrrrr1207/MOSAIC-Recommendation.

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing, pages 9998–10013
October 24-29, 2026 ©2026 Association for Computational Linguistics

2024) leverages Large Language Models (LLMs) to extract a more precise extraction of item-related characteristics. Going further, LLM-based recommenders such as EXP3RT (Kim et al., 2025) treat LLMs as *review interpreters*, distilling reasoning from a teacher LLM to extract preferences from raw reviews for rating prediction. However, these methods fundamentally process information *available within each user's own reviews*, leaving user-item pairs without review content weakly supervised and the alleviation of UGC sparsity bounded by what individual users have written.

Beyond these limitations, a more subtle and granular form of data sparsity persists: the **inherent incompleteness of individual reviews**. As shown in Figure 1, users rarely provide exhaustive reviews that fully encapsulate their complete set of preferences, leaving attribute-level gaps in their preference profiles. For instance, a user might praise a tablet's "*high-resolution screen*" but omit the *"battery life"* that also matters to them. **Relying on a single user's reviews is often insufficient to fully capture their diverse preferences**. Existing NLP or LLM-based extraction methods cannot compensate for this absence, since they can only identify attributes explicitly mentioned in the text. While LLM-based recommenders such as LLMRec (Wei et al., 2024a) can generate auxiliary content to enrich sparse side information, such content remains susceptible to LLM hallucination and lacks verifiable grounding in real user feedback. This subtle yet critical form of sparsity significantly limits the ability of RSs to construct holistic user representations, hence motivating the development of strategies that enrich individual reviews with complementary information grounded in real user evidence.

In this work, we propose MOSAIC (**M**eta-review **O**n **S**parse **A**nd **I**ncomplete user-generated **C**ontent), which introduces the idea of *meta-review* from the academic peer review process (Bhatia et al., 2020) to address this fine-grained sparsity. MOSAIC hypothesizes that aggregating reviews from neighbor users with similar preferences yields a more comprehensive view than any single review. Unlike prior work that aggregates neighborhood information at the review or instance level(Shuai et al., 2022), our meta-review construction operates at the **attribute–sentiment level**, directly complementing absent or insufficient fine-grained preference signals.

MOSAIC proceeds in three components organized as a **cascaded framework**. First, we extract attribute-sentiment pairs from individual user reviews and aggregate them across neighbor users to **formulate meta-reviews**. Then, a multi-gate MoE (MMoE) structure is designed to jointly perform **rating prediction** and **attribute-sentiment label prediction**, where the latter provides explicit fine-grained preference signals beyond serving as an auxiliary explainability task. Finally, a **personalized attention module** conditions on the target user to selectively highlight relevant attributes from the meta-review, yielding refined rating predictions and attribute-level explanations.

In summary, our contributions are outlined as follows:

- Motivated by the meta-review in the academic peer review process, we introduce a new perspective, **attribute-level meta-reviews** aggregated from neighbor users, to address UGC **sparsity and incompleteness**.
- We propose MOSAIC, a novel multi-task review-based recommendation framework that incorporates a **personalized attention module**, selectively highlighting the attributes most relevant to the target user for refined rating prediction and attribute-level explanation.
- Extensive experiments on four public benchmark datasets demonstrate the superiority of MOSAIC over state-of-the-art baselines. Further ablation studies verify the effectiveness of each component, and additional analyses confirm consistent gains for users with limited interaction history.

2 Related Work

2.1 Explainable Recommender Systems

Explainable recommender systems provide textual or attribute-level justifications. Existing methods either extract existing phrases or reviews (Wang et al., 2022; Pugoy and Kao, 2021), or generate natural-language explanations with RNN-, Transformer-, PLM-, or LLM-based models (Li et al., 2017, 2021, 2023a; Ariza-Casabona et al., 2023; Li et al., 2023b; Zhou et al., 2024; Zhang et al., 2024). Recent hybrid methods further ground generation on retrieved reviews or attributes to reduce hallucination (Cheng et al., 2023a; Zhan et al., 2023). However, these methods still rely on available review content and thus struggle when reviews are missing or incomplete.

2.2 Review-based Recommendation

Review-based recommendation uses reviews to improve rating prediction beyond ID-based models (Koren et al., 2009). Prior work encodes review semantics with CNNs (Kim et al., 2016; Catherine and Cohen, 2017), attention mechanisms (Chen et al., 2018; Seo et al., 2017; Liu et al., 2019; Wu et al., 2019), graph or contrastive learning (Wang et al., 2023a; Liu et al., 2025; Shuai et al., 2022; Yang et al., 2023; Wang et al., 2023b; Wei et al., 2024b), and LLM-based aspect extraction (Li et al., 2023c). Nevertheless, most methods depend on individually coarse review-level signals, leaving UGC sparsity and attribute-level incompleteness insufficiently addressed. We provide a more comprehensive discussion of related studies in Appendix C.

3 Motivation

3.1 LLM-Based Attribute-Sentiment Extraction

LLMs have demonstrated strong capabilities in semantic understanding and structured information extraction. Leveraging these capabilities, we employ an LLM to extract concise attribute-sentiment pairs from unstructured review texts, which serve as both interpretable building blocks for explanations and semantic features for review-based recommendation.

Let $s_{u,i}$ denote a review written by user u on item i. Given a predefined domain-specific attribute set $\mathcal{A} = \{a_1, a_2, \ldots, a_A\}$ with A attributes, we define the extraction function performed by the LLM as:

$$f_{\text{LLM}}(s_{u,i}, \mathcal{A}) \rightarrow \{(a_j, o_j) \mid a_j \in \mathcal{A}, o_j \in \mathcal{O} \cup \Phi\},$$

where o_j denotes the extracted sentiment for attribute a_j and $\mathcal{O}$ is the vocabulary of possible sentimental phrases (limited to 3 words). If an attribute a_j is not explicitly mentioned or lacks sentiment polarity in the review, the sentiment set is empty (represented by "N/A" in the natural language description).

To construct the attribute set $\mathcal{A}$ for each dataset, we reference domain-specific attribute taxonomies, guided by prior work such as FineRec (Zhang et al., 2024). In particular, the attribute list is constructed by selecting the most frequently mentioned features across all reviews, ensuring coverage and relevance. Then, the LLM is instructed using a carefully designed prompt to encourage structured and sentiment-aware extraction. In our paper, *gpt-4.1-nano* is used for attribute-sentiment pairs extraction, the full prompt template is provided in Appendix A.

Then, we adopt the sentiment analysis to leverage the extracted attribute-sentiment pairs. Specifically, we combine each attribute and its corresponding extracted sentiment as a new sentence: "*The {attribute} is {sentiment}*", and use a fine-tuned pre-trained BERT (Sanh et al., 2019) to do a sentiment classification with C classes. In this paper, we set $C = 4$, which categorizes the sentiment into one of four classes: *positive*, *slightly positive*, *slightly negative*, and *negative*. Upon obtaining the sentiment scores, we can obtain the sentiment label $\mathbf{Y}_{u,i} \in \{0, 1\}^{A \times C}$ for each user u and item i.

Notably, viewed as a superior alternative to traditional extraction methods(Zhou et al., 2024), the LLM does not participate in model training or inference and does not introduce additional computational cost(see Appendix B.4).

3.2 Incompleteness of User Review

In contrast to sparsity, where a review is entirely missing, incompleteness arises when a user, despite writing a review, does not mention all the attributes they considered. Although both phenomena degrade recommendation quality, **incompleteness of the review** has been largely overlooked in prior work (Lei et al., 2025; Vartak et al., 2017). In this section, we empirically investigate this overlooked problem and motivate our solution.

To validate the existence of review incompleteness, we first extracted attribute-sentiment pairs from the *TripAdvisor* dataset using six predefined attributes: *Service, Cleanliness, Rooms, Location, Price, and Sleep Quality*. As shown in Figure 2, our analysis reveals that half of the attributes had a coverage rate below 50% across all reviews, confirming that single user reviews are often incomplete. This phenomenon is also observable at the individual level. For example, Figure 1 illustrates a case where a user gave the same hotel an identical rating on two occasions but mentioned a different set of attributes each time. This suggests that a user's written feedback may not fully capture their comprehensive preferences.

In academic peer review, a single reviewer's opinion may not capture a paper's full quality, motivating the meta-review process that aggregates multiple reviewers (Bhatia et al., 2020). Inspired by this, we aggregate reviews from neighbor users to form a more complete representation of prefer-

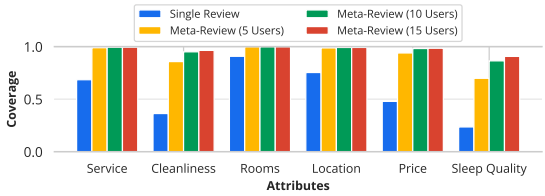

Figure 2: Coverage of attributes in single review and meta-reviews on the Tripadvisor dataset.

ences. Our exploratory analysis, depicted in Figure 2, demonstrates that aggregating reviews from as few as 10 neighbors can increase attribute coverage to over 80%, effectively counteracting incompleteness. This aggregation strategy also addresses the sparsity problem by generating preference signals for interactions where the target user left no review.

4 Methodology

4.1 Model Overview

MOSAIC frames recommendation as a multi-task learning problem that jointly performs **rating prediction** and **attribute-sentiment label prediction** over both user-level and group-level (meta-review) signals. To balance these two tasks and prevent negative transfer (Wei et al., 2023), we adopt a *Multi-gate Mixture-of-Experts* (MMoE) (Ma et al., 2018) architecture with task-specific towers. A **personalized attention module** then calibrates the predicted attribute-sentiments to each target user and refines the rating prediction. The full pipeline is illustrated in Figure 3.

4.2 Problem Formulation

In this paper, we formulate our framework as a multi-task learning problem that jointly performs **rating prediction** and **attribute-sentiment label prediction**.

Definition 1 (Rating Prediction)**.** *Suppose the user set $\mathcal{U}$ and item set $\mathcal{I}$, given a user $u \in \mathcal{U}$ and item $i \in \mathcal{I}$, our task is to predict a rating $\hat{r}_{u,i}$ to align with the groundtruth rating $r_{u,i}$.*

Definition 2 (Attribute-sentiment Label Prediction)**.** *Given a user $u \in \mathcal{U}$ and item $i \in \mathcal{I}$, we predict a sentiment label $\hat{\mathbf{Y}}_{u,i} \in \{0,1\}^{A\times C}$, which consists of A attributes and C sentiment types. The prediction is going to be aligned with the groundtruth sentiment label $\mathbf{Y}_{u,i}$ extracted from user review $s_{u,i}$, only when $s_{u,i}$ is available.*

4.3 Construction of User Meta-Review

Locating Neighbor Users. For a target user u and a target item i, we identify candidate neighbors as users who have reviewed the same item, optionally augmented with the target user's social friends. From this candidate set, we randomly sample $N = 15$ users as neighbor users $\mathcal{N}_{u,i}$. We intentionally do *not* filter neighbors by sentiment alignment with the target user since potential sentiment heterogeneity is mitigated by downstream majority voting. Candidate selection aims at broad attribute coverage, while **preference alignment is deferred to the personalized attention module**, which selectively emphasizes neighbor sentiments that match the target user's preferences. As illustrated in Figure 2, randomly sampling $N = 15$ users is already sufficient to achieve adequate diversity at the attribute level.

Generating User Meta-Review. Given the sentiment label $\mathbf{Y}_{u,i} \in [0,1]^{A\times C}$ for user u and item i, we can derive the meta-review attribute-sentiment vector $\mathbf{V}_{\mathcal{N}_{u,i}} \in \mathbb{R}^{A\times C}$ for the user neighbor set $\mathcal{N}_{u,i}$ through mean pooling, and obtain the meta-review attribute-sentiment labels $\mathbf{Y}_{\mathcal{N}_{u,i}} \in [0,1]^{A\times C}$ for the user neighbor set $\mathcal{N}_{u,i}$ through majority voting on each attribute respectively:

$$\mathbf{V}_{\mathcal{N}_{u,i}} = \frac{\sum_{u\in\mathcal{N}_{u,i}} \mathbf{Y}_{u,i}}{|\mathcal{N}_{u,i}|},\ \mathbf{Y}_{\mathcal{N}_{u,i}} = \text{MajorVote}_{a\in\mathcal{A}}(\mathbf{V}_{\mathcal{N}_{u,i}}), \tag{1}$$

where $\text{MajorVote}_{a\in\mathcal{A}}$ denotes selecting the sentiment labels with the maximum value in each attribute a, thereby helps reduce noise and mitigate potential sentiment heterogeneity across users. The derived meta-review sentiment representations, $\mathbf{V}_{\mathcal{N}_{u,i}}$ and $\mathbf{Y}_{\mathcal{N}_{u,i}}$, are jointly used as supervision signals for the sentiment analysis of the neighbor users, which helps balance collective signals and reduces the influence of extreme or inconsistent user opinions.

4.4 Model Architecture

Input Representation. The rating prediction task uses only the target user embedding $\mathbf{e}_u$ and item embedding $\mathbf{e}_i$, while the attribute-sentiment prediction task additionally incorporates an aggregated neighbor embedding:

$$\mathbf{e}_{\mathcal{N}_{u,i}} = \frac{1}{|\mathcal{N}_{u,i}|}\sum_{g_i\in\mathcal{N}_{u,i}} \mathbf{e}_{g_i}, \tag{2}$$

where $\mathcal{N}_{u,i}$ is the located neighbor users. Thus, For task $t \in \{r, o\}$, where r denotes rating prediction and o denotes meta-review sentiment prediction, the input vectors are:

$$\mathbf{e}^r = \mathbf{e}_u || \mathbf{e}_i \in \mathbb{R}^{2d}, \mathbf{e}^o = \mathbf{e}_{\mathcal{N}_{u,i}} || \mathbf{e}_i \in \mathbb{R}^{2d}, \tag{3}$$

where $||$ denotes the concatenation operation.

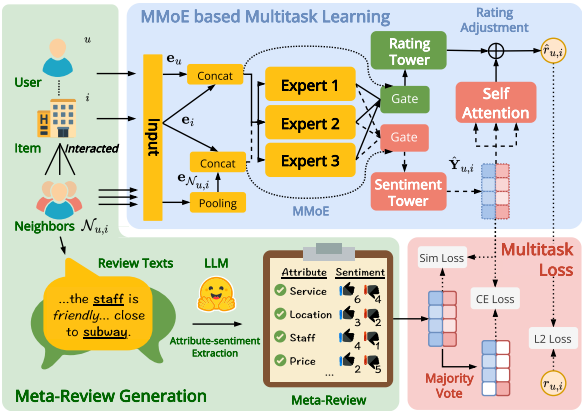

Figure 3: Overview of MOSAIC. (1) **Meta-review Generation** LLM extracts attribute-sentiment pairs from neighbor users' reviews. (2) **MMoE-Based Multitask Learning**: MMoE jointly models rating and attribute-sentiment prediction, followed by a personalized attention module that conditions on the target user to refine ratings. (3) **Multitask Loss**: regression, cross-entropy, and similarity loss jointly supervise training.

4.4.1 Multi-Task Integration of Meta-Review

To balance the two tasks while mitigating negative transfer between individual reviews and meta-review signals, we adopt the MMoE framework with shared experts and task-specific gating networks. Each gate dynamically weights the experts' contributions for its task, creating functionally distinct feature spaces for the disparate input representations of the two tasks. The components of the MMoE in MOSAIC are detailed as follows:

Expert Networks. For each expert $k = 1, \ldots, K$, we employ a Transformer encoder (Li et al., 2021) to capture contextualized representations. The expert transformation is given by:

$$\mathbf{x}_k^t = \phi(\mathbf{W}_k^I \mathbf{h}^t + \mathbf{b}_k^I), \;\; \mathbf{z}_k^t = \text{Transformer}_k(\mathbf{x}_k^t), \quad (4)$$

where $\mathbf{h}^t$ is the input vector after the feature representation layer, $\mathbf{W}_k^I, \mathbf{b}_k^I$ are input projection parameters for expert k, $\mathbf{x}_k^t$ is the projected input to expert k, and $\mathbf{z}_k^t$ is its contextualized output.

Task-specific Gating. The MMoE module adaptively assigns expert contributions to each task through a softmax gating mechanism. For task $t \in \{r, o\}$, the output is:

$$\mathbf{m}^t = \textstyle\sum_{k=1}^{K} \alpha_k^t \mathbf{z}_k^t, \quad (5)$$

where the gating weights α_k for each expert k are defined as:

$$\alpha_k = \frac{\exp\left((\mathbf{w}_k^t)^T \mathbf{h}^t + b_k^t\right)}{\sum_{j=1}^{K} \exp\left((\mathbf{w}_j^t)^T \mathbf{h}^t + b_j^t\right)}, \quad (6)$$

where $\mathbf{w}_k^t, b_k^t$ are gating parameters for expert k and task t.

Task-specific Towers. Given the aggregated representations $\mathbf{m}^r$ and $\mathbf{m}^o$ from the MMoE module, two task-specific towers produce the final outputs of each task. The rating tower applies an MLP on $\mathbf{m}^r$ to predict the initial rating $\hat{r}^r$, while the sentiment tower applies an MLP followed by a softmax on $\mathbf{m}^o$ to produce attribute-wise sentiment distributions $\hat{\mathbf{y}}_a \in \mathbb{R}^C$ for each attribute $a = 1, \ldots, A$.

4.4.2 Personalization of Meta-Review Signals

While meta-reviews aggregate sentiment signals across multiple neighbor users, the target user's specific preferences may emphasize only a subset of these attributes. We therefore introduce a user-specific personalization step that selectively highlights the attributes most relevant to the target user, ensuring that the aggregated neighbor signals are calibrated to individual preferences rather than treated uniformly.

Specifically, the personalization step is implemented as a self-attention module that takes the rating features $\mathbf{m}^r$ as the query and predicted attribute-sentiment signals as keys and values, producing user-conditioned weights that emphasize attributes aligned with the target user's preferences and suppress less relevant ones. We then collect all attribute predictions into $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_{a_1}, \ldots, \hat{\mathbf{y}}_{a_A}]$ and project them through a sentiment embedding space via a learnable matrix $\mathbf{W}^S \in \mathbb{R}^{C \times d}$, followed by a self-attention layer to obtain the pooled representation:

$$\mathbf{p} = \text{Self-attention}(\hat{\mathbf{Y}}\mathbf{W}^S, \mathbf{m}^r). \quad (7)$$

The adjustment rating score $\hat{r}^o$ is obtained by combining the sentiment representation $\mathbf{p}$ with the rating features $\mathbf{m}^r$ through non-linear perceptron layer $\Gamma(\cdot)$ as the fusion gate:

$$\hat{r}^o = MLP\left(\mathbf{m}^r \cdot \Gamma(\mathbf{m}^r || \mathbf{p}) + \mathbf{p} \cdot (1 - \Gamma(\mathbf{m}^r || \mathbf{p}))\right). \quad (8)$$

The attention module adaptively weighs user-item representations and meta-review sentiment, enabling the model to refine the final rating prediction with contextual signals.

4.5 Loss Functions

The final rating combines the initial prediction $\hat{r}_{u,i}^r$ and the meta-review guided adjustment $\hat{r}_{u,i}^o$, supervised by squared error against the ground truth $r_{u,i}$:

$$\mathcal{L}_{rating} = \frac{1}{|\mathcal{T}|} \textstyle\sum_{(u,i) \in \mathcal{T}} \left(\hat{r}_{u,i} - r_{u,i}\right)^2, \quad (9)$$

where $\mathcal{T}$ is the training set.

In the meta-review attribute-sentiment label prediction task, the user-level sentiment label $\mathbf{Y}_{u,i}$ supervises $\hat{\mathbf{Y}}_{\mathcal{N}_{u,i}}$ at the user-level to maintain label consistency, while the ground-truth group-level labels $\mathbf{Y}_{\mathcal{N}_{u,i}}$ from meta-reviews via majority vote (Section 4.3) provide broader supervision. As $\mathbf{Y}_{\mathcal{N}_{u,i}}$ is derived from the meta-review sentiment vector $\mathbf{V}_{\mathcal{N}_{u,i}}$, the original sentiment distribution preserves richer and more nuanced information. Therefore, we design a joint loss that includes both the classification loss of the sentiment label $\mathbf{Y}_{\mathcal{N}_{u,i}}$ and the similarity of the meta-review sentiment vector $\mathbf{V}_{\mathcal{N}_{u,i}}$, controlled by parameters η_{cls} and η_{sim}. This joint supervision strategy enables the model to learn from consensus signals while preserving sentiment variability among neighbors, reducing the risk of overfitting to hard labels and limiting the impact of outlier users. The joint loss for meta-review sentiment prediction $\mathcal{L}_{meta}$ is given by:

$$\mathcal{L}_{meta} = -\frac{1}{|\mathcal{T}|} \sum_{(u,i)\in\mathcal{T}} \eta_{cls}(\mathbf{Y}_{u,i} \log \hat{\mathbf{Y}}_{\mathcal{N}_{u,i}} + \mathbf{Y}_{\mathcal{N}_{u,i}} \log \hat{\mathbf{Y}}_{\mathcal{N}_{u,i}}) + \eta_{sim}\, \mathrm{Sim}(\hat{\mathbf{Y}}_{\mathcal{N}_{u,i}}\mathbf{W}^S, \mathbf{V}_{\mathcal{N}_{u,i}}\mathbf{W}^S). \tag{10}$$

Finally, we can obtain the final loss function combination: $\mathcal{L} = \mathcal{L}_{rating} + \mathcal{L}_{meta}$. For the similarity function $\mathrm{Sim}(\cdot)$, we adopt the cosine similarity. The settings of hyperparameters η_{cls} and η_{sim} will be discussed in Section 5.3.

5 Experiments

5.1 Datasets and Baselines

Datasets. We conduct experiments on four publicly available datasets spanning diverse application domains, which are *Yelp* (restaurants), *TripAdvisor* (hotels), *Amazon Beauty*, and *Amazon Sports*. Each record contains a user ID, an item ID, a numeric rating, and a corresponding review text (can be empty), annotated with attribute-sentiment pairs as described in Section 3.1. We filter users with fewer than three rated items. Data statistics and predefined attributes are presented in Table 1. Full descriptions and preprocessing details are in Appendix B.1.

Baselines. We compare MOSAIC against several state-of-the-art baselines, covering classical matrix factorization (**PMF** (Mnih and Salakhutdinov, 2007) and **SVD++** (Koren, 2008)), review-based recommendation models (**RGCL** (Shuai et al., 2022), **APH** (Liu et al., 2025)), explainable recommendation models (**NETE** (Li et al., 2020), **PETER** (Li et al., 2021), **PEPLER** (Li et al., 2023a), **CER** (Raczyński et al., 2023), **SERMON** (Liao et al., 2025)), and a LLM-based recommender (**LLMRec** (Wei et al., 2024a)). For the LLM-based baseline, we replace its original LLM backbones with *gpt-4.1-nano* to ensure fair comparison with MOSAIC's extraction pipeline. Detailed descriptions are provided in Appendix B.2.

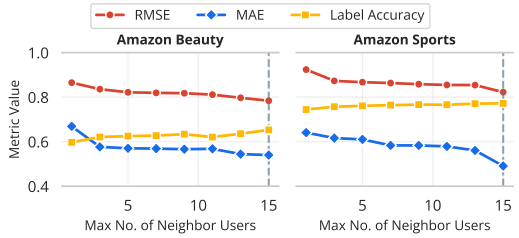

Figure 4: Performance with different numbers of neighbor users.

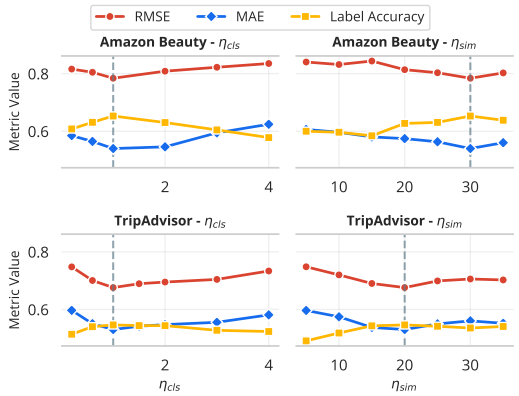

Figure 5: Performance with different η_{cls} and η_{sim}.

5.2 Overall Performance

We evaluate rating prediction with RMSE and MAE, and attribute-sentiment prediction with Accuracy, F1 Score, Precision, and Recall. Full experimental settings, including data split, training and evaluation details are in Appendix B.3.

Recommendation Performance. Table 2 reports the rating prediction performance on the four datasets. The results demonstrate that MOSAIC consistently outperforms all comparative methods under both metrics across all datasets. In particular, our method achieves superior performance over classical collaborative filtering models such as PMF and SVD++, review-based methods such as RGCL and APH, joint rating–explanation models (NETE, PETER, PEPLER, CER, SERMON), a LLM-based recommender(LLMRec). Notably, MOSAIC surpasses RGCL by aggregating neighborhood information at the **attribute-level** rather than the review level, directly addressing review incompleteness. Additionally, LLMRec's weaker

Table 1: Statistics of the datasets.

Dataset	# users	# items	# reviews	Attributes list
Yelp	19,348	9,020	103,022	Ambience, Cleanliness, Food, Location, Parking, Price, Service
TripAdvisor	3,255	6,261	104,570	Service, Cleanliness, Rooms, Location, Price, Sleep Quality
Amazon Beauty	12,784	5,190	32,506	Color, Effectiveness, Price, Ingredients, Scent, Size
Amazon Sports	32,374	24,122	140,603	Brand, Comfort, Functionality, Material, Price, Quality, Size

Table 2: Performance on rating prediction. In each row, the best result is in **bold** and the best baseline is underlined. * indicates statistical significance with $p < 0.01$.

Dataset	Metric	PMF	SVD++	RGCL	NETE	PETER	PEPLER	CER	SERMON	APH	LLMRec	MOSAIC
Yelp	RMSE	0.9866	0.9405	0.9086	1.0529	0.9454	0.8592	0.8652	1.0871	0.8504	1.1930	**0.8239***
	MAE	0.7119	0.7284	0.6951	0.7650	0.7064	0.5854	0.6634	0.8142	0.6576	0.8975	**0.5708**
Trip-Advisor	RMSE	0.8777	0.9010	0.7798	0.7852	0.8058	0.7799	0.8212	0.8936	0.7826	1.0238	**0.6759***
	MAE	0.6769	0.7060	0.5989	0.5919	0.6329	0.6160	0.6406	0.6913	0.6055	0.8231	**0.5303***
Amazon Beauty	RMSE	1.1073	0.8886	0.8565	0.9884	1.1231	0.9613	1.1236	1.2940	0.8574	1.3176	**0.7840***
	MAE	0.8618	0.7124	0.6717	0.6774	0.8884	0.6650	0.8616	0.9141	0.6452	0.7512	**0.5400***
Amazon Sports	RMSE	0.9659	1.4380	0.8866	0.9181	0.9360	0.9112	0.9375	1.0524	0.9383	1.4580	**0.8224***
	MAE	0.6912	1.0829	0.5901	0.6478	0.6745	0.6493	0.6823	0.7473	0.6840	1.0326	**0.4909***

performance indicates that LLM-driven graph augmentation alone is less effective for fine-grained rating regression. MOSAIC focuses on user-side evidence and achieves consistent advantages on all four datasets.

Meta-review Attribute-sentiment Prediction Performance. For this task, we include only explanation-capable methods, namely NETE, PETER, PEPLER, CER, and SERMON. As shown in Table 3, MOSAIC achieves the best overall performance. While some baselines achieve slightly higher precision, their lower F1 and recall reveal a tendency to predict positive labels only when highly certain. MOSAIC instead identifies a broader range of correct labels, yielding more balanced performance across metrics.

Beyond these performance gains, Appendix B.4 shows that MOSAIC maintains competitive training speed and low memory consumption compared to baselines.

5.3 Ablation and Sensitivity Analysis

We conduct ablation and sensitivity analysis to evaluate the contribution of each proposed component and the model's robustness to key hyperparameters. Table 5 shows that the full model consistently outperforms all variants, validating each proposed strategy. Specifically, we evaluate the performance of the following model variants:

Effect of Neighbor Users. Removing neighbor users causes the largest drop on both datasets (e.g., RMSE 0.6759 $\rightarrow$ 0.7974 on TripAdvisor). As shown in Figure 2, without meta-reviews aggregated from neighbors, the model loses access to the broader attribute coverage and the cross-user evidence required to compensate for review incompleteness, which directly translates to weaker rating prediction and sentiment label accuracy.

Effect of Number of Neighbor Users. To examine the model's *sensitivity* to the number of neighbors, Figure 4 illustrates that increasing the maximum number of neighbor users generally improves RMSE, MAE, and label accuracy. These results confirm that richer attribute coverage enhances both prediction tasks. This aligns with our Figure 2, where 10–15 neighbors are sufficient to achieve over 80% attribute coverage.

Effect of Attention. Replacing the self-attention module with a uniform MLP deteriorates performance across all metrics, such as RMSE 0.7840 $\rightarrow$ 0.9252 on Beauty. The MLP treats all attributes in the meta-review uniformly, whereas the attention module learns user-specific weights that highlight relevant attributes and suppress less relevant ones. Without this personalization step, the group-level meta-review signals cannot be calibrated to the target user, leading to biased rating adjustment.

Effect of Meta-Review Supervision. The two coefficients play complementary roles: η_{cls} aligns predictions with majority-voted hard labels, while η_{sim} preserves the fine-grained sentiment distribution that majority voting discards. Disabling either or both signals causes sentiment accuracy to collapse. Interestingly, MAE slightly improves when removing both compared to removing only one, reflecting the inherent competing tension between the two tasks. *Beyond ablation,* we vary $\eta_{cls} \in [1, 5]$ and $\eta_{sim} \in [5, 35]$ (with a step size of 5), shown in Figure 5. The results indicate that η_{sim} requires a substantially larger weight than η_{cls} to balance the soft-distribution alignment against the hard-label classification(e.g. $\eta_{cls} = 1$, $\eta_{sim} = 30$ on the *Beauty* dataset).

Table 3: Performance on meta-review sentiment prediction. In each column, the best result is in **bold** and the best baseline is underlined. * indicates statistical significance with $p < 0.01$. Acc., F1, Prec., Rec. denote Accuracy, F1 Score, Precision, and Recall respectively.

Method	Yelp				TripAdvisor				Amazon Beauty				Amazon Sports			
	Acc.	F1	Prec.	Rec.	Acc.	F1	Prec.	Rec.	Acc.	F1	Prec.	Rec.	Acc.	F1	Prec.	Rec.
NETE	0.5909	0.1912	0.1625	0.2321	0.2773	0.1199	0.2487	0.2476	0.5959	0.1874	0.2257	0.2500	0.3913	0.1522	0.1320	0.2371
PETER	0.5977	0.2024	0.1701	0.2500	0.3444	0.1835	0.3685	0.2867	0.6122	0.2232	0.3617	0.2668	0.4225	0.1586	0.3956	0.2544
PEPLER	0.5970	0.2535	0.2665	0.2593	0.3714	0.2193	0.2799	0.2973	0.5965	0.1868	0.1491	0.2500	0.4203	0.1481	0.3141	0.2501
CER	0.6145	0.2475	0.2668	0.2580	0.3632	0.1979	0.3702	0.2947	0.5974	0.1896	**0.4042**	0.2513	0.4226	0.1594	0.4100	0.2549
SERMON	0.6797	0.2030	0.2908	0.2501	0.3696	0.2066	0.3657	0.2969	0.6465	0.2684	0.3273	0.2954	0.4243	0.1741	0.3534	0.2613
MOSAIC	**0.7064***	**0.3488***	**0.5217***	**0.3335***	**0.5463***	**0.2877***	**0.3769**	**0.3102**	**0.6529**	**0.3101***	0.3119	**0.3277***	**0.7720***	**0.5160***	**0.5446***	**0.5201***

Table 4: Rating prediction on Yelp and Amazon Sports across user groups partitioned by interaction count. **Improv.** denotes MOSAIC's relative gain over the best baseline. **Best** and runner-up results are highlighted.

Dataset	Group (# Samples)	Metric	MOSAIC	RGCL	PETER	PEPLER	APH	LLMRec	Improv.
Yelp	Sparse ($\leq$ 1) (13,604)	RMSE	**0.8904**	1.1621	1.1268	1.3054	1.0581	1.4567	**15.85%**
		MAE	**0.5902**	0.8682	0.8793	1.0589	0.8477	1.1338	**30.38%**
	Moderate (2–5) (15,070)	RMSE	**0.7924**	1.0252	0.9872	1.2083	0.9497	1.2383	**16.56%**
		MAE	**0.5902**	0.7426	0.7624	0.9762	0.7482	0.9179	**20.52%**
	Active ($>$ 5) (8,749)	RMSE	**0.7860**	0.9244	0.8854	1.1194	0.8478	0.9819	**7.29%**
		MAE	**0.5984**	0.6639	0.6865	0.9017	0.6648	0.7561	**9.87%**
Sports	Sparse ($\leq$ 1) (14,701)	RMSE	**0.7050**	0.9436	0.9890	0.9635	0.9119	1.6336	**22.69%**
		MAE	**0.4425**	0.6053	0.6156	0.7132	0.6740	1.2683	**26.90%**
	Moderate (2–5) (11,162)	RMSE	**0.6075**	0.8980	0.9601	0.8993	0.8743	1.4076	**30.52%**
		MAE	**0.3905**	0.5496	0.6022	0.6514	0.6291	0.9617	**28.95%**
	Active ($>$ 5) (4,791)	RMSE	**0.5266**	0.7693	0.8241	0.7831	0.7726	1.2621	**31.55%**
		MAE	**0.3577**	0.4666	0.5373	0.5771	0.5469	0.8281	**23.34%**

Table 5: Ablation study.

Model	Beauty			TripAdvisor		
	RMSE	MAE	Accuracy	RMSE	MAE	Accuracy
w/o Neighbor Users	0.9331	0.6778	0.5887	0.7974	0.6167	0.5168
w/o Attention	0.9252	0.6452	0.6372	0.7857	0.6109	0.5003
$\eta_{cls}, \eta_{sim} = 0$	0.9721	0.5663	0.3073	0.7249	0.5734	0.4188
$\eta_{sim} = 0$	0.8533	0.6007	0.5283	0.7172	0.5681	0.4969
$\eta_{cls} = 0$	0.8485	0.6051	0.6289	0.7154	0.5706	0.5361
Full Model	**0.7840**	**0.5400**	**0.6529**	**0.6759**	**0.5303**	**0.5463**

5.4 Performance on Sparse Users

We further evaluate MOSAIC on users with limited interaction history. With few review evidence, it leaves a challenging setting for preference modeling due to highly limited attribute coverage.

Setting. We partition the test interactions of the Yelp and Amazon Sports datasets into three groups based on the number of training interactions associated per test user: *Sparse* (≤ 1 training interaction), *Moderate* (2–5), and *Active* (> 5). The *Sparse* group corresponds to users with the minimum training interactions under our split. We compare MOSAIC against the strongest baselines from Table 2 (RGCL, APH, PEPLER, PETER) and LLMRec, whose explicit focus on data sparsity makes it particularly relevant here.

Results. Table 4 shows MOSAIC consistently outperforms all baselines across the three groups on both datasets. The advantage is particularly pronounced in the *Sparse* group, where MOSAIC reduces MAE by 30.38% on Yelp and 26.90% on Amazon Sports relative to the strongest baseline. This pattern strongly illustrates that meta-reviews aggregated from neighbor users provide the missing fine-grained signals when the target user's own evidence is minimal. The advantage persists in the *Moderate* and *Active* groups, suggesting that group-level evidence remains informative even when individual review history grows richer. These results validate our design motivation that aggregating attribute-level meta-reviews offers a consistent benefit across a wide spectrum of user activity levels. Notably, LLMRec shows weaker performance throughout, suggesting that LLM-driven graph augmentation alone may be insufficient for the attribute-level reasoning required in review-based rating prediction.

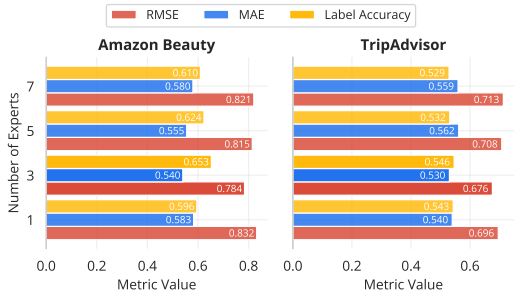

Figure 6: Performance with different number of experts K.

5.5 Number of Experts and Gating Behavior in MMoE

We analyze the impact of varying the number of experts in the MMoE module to account for the heterogeneity of task-specific inputs. As shown in

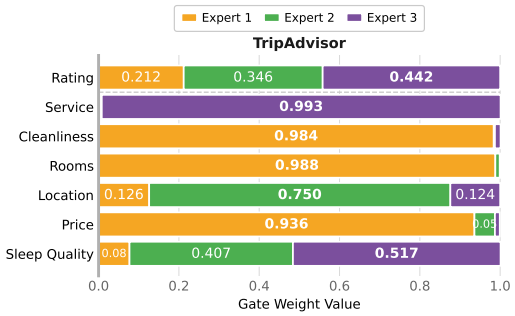

Figure 7: Mean gate weights across experts at K=3 on TripAdvisor(13,027 test samples).

Figure 6, performance consistently improves as the number of experts K increases from 1 to 3, while further increasing the number of experts leads to diminishing performance gains. The best trade-off for both rating prediction and attribute-sentiment prediction is observed when the number of experts is 3.

To understand this trend, we examine the learned gate distributions on TripAdvisor. As shown in Figure 7, at K=3 each attribute gate concentrates its weight on a dominant expert (e.g., Cleanliness 0.984 and Rooms 0.988 on Expert 1, Location 0.750 on Expert 2, Service 0.993 on Expert 3), while the rating gate distributes far more evenly (0.212 / 0.346 / 0.442). The two tasks thus learn markedly different routing patterns over the same shared experts, confirming that the task-specific gates perform differentiated, semantically meaningful routing rather than collapsing to a shared representation, which is precisely what a single shared bottom cannot produce by construction. Importantly, gating does not break down at larger K and every attribute is still routed to a dominant expert. However, the extra experts do not form new specialists but instead fragment attributes that a single expert handled at $K = 3$, so each expert receives a weaker, more diffuse training signal as the parameter count grows. This explains the slight degradation beyond three experts, with full $K = 5/7$ distributions and analysis are provided in Table 7 in the Appendix B.5. Overall, these results confirm that K=3 offers the best balance between expert specialization and model capacity.

5.6 Case Study

The model is designed to enhance the accuracy of rating prediction in RSs by leveraging reviews from neighbor users. Figure 8 presents a case to illustrate the effectiveness of incorporating these meta-reviews compared to relying solely on the target user's own review. When only using the user's own review, the limited information fails to comprehensively characterize the item, making it challenging for the model to accurately capture the user's overall preference on the target item with such a sparse contextual signal. Consequently, the model predicts a rating of 3.8, which deviates notably from the ground-truth rating of 5.0.

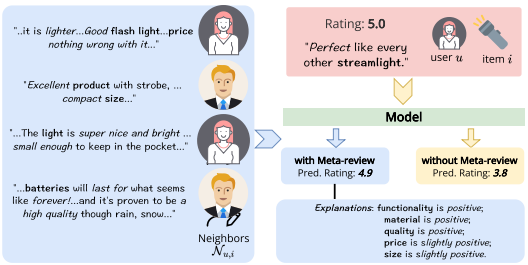

Figure 8: Case study to show the effectiveness of meta-review.

When leveraging the meta-review from neighbor users, this rich, aggregated information enables the generated meta-review to better model user preferences and complement the user's incomplete review, thereby explaining the perceived "perfect" evaluation. This approach effectively alleviates attribute incompleteness and review sparsity, leading to a much more accurate predicted rating of 4.9, closely matching the true score.

Furthermore, our model demonstrates interpretability through attribute-level sentiment prediction. The final explanation box aligns the overall positive review ("Perfect like every other streamlight") with specific positive sentiments on functionality, material, and quality, and mildly positive sentiments on price and size. This case highlights how meta-reviews successfully enhance both rating accuracy and explainability in real-world recommendation settings.

6 Conclusion

We present MOSAIC, which addresses UGC **sparsity and incompleteness** by aggregating attribute-sentiment evidence from neighbor users. MOSAIC combines an MMoE-based multitask architecture with a personalized attention module that conditions on the target user, jointly producing refined rating predictions and attribute-level explanations. Experiments on four real-world datasets show consistent gains over state-of-the-art baselines, and analyses confirm that meta-reviews expand attribute coverage beyond any single user's reviews and maintain consistent gains for users with limited interaction history.

Limitations

We acknowledge several limitations that suggest directions for future research. First, MOSAIC relies on a predefined domain-specific attribute taxonomy. Adapting to new domains requires reconstructing the attribute list, and emerging or niche attributes outside the predefined set cannot be captured. Future work could explore dynamic attribute discovery or open-vocabulary extraction. Second, neighbor selection in MOSAIC is based on shared item interactions and optional social links, which may be sparse for highly isolated users. Integrating graph-based or behavior-similarity neighbor finding could further extend MOSAIC's applicability. Finally, MOSAIC focuses on user-side neighbor-aggregated evidence. Integrating complementary item-side reasoning remains a promising direction for hybrid designs that further enhance performance in domains with rich item metadata.

MOSAIC employs gpt-4.1-nano to extract attribute–sentiment pairs from existing user reviews, which raises potential concerns about LLM hallucination. We mitigate this risk through several design choices: the LLM is used only as an offline extraction tool and does not participate in training, inference, or the generation of any new content shown to users; its outputs are constrained to a predefined attribute taxonomy with a bounded sentiment vocabulary (limited to three words per opinion); and the extracted sentiments are further verified by a fine-tuned BERT classifier rather than being trusted directly. Most importantly, the meta-review construction aggregates evidence across multiple neighbor users via majority voting at the attribute level, which suppresses hallucinated extractions from any individual review. While these safeguards substantially reduce the impact of hallucination, we acknowledge that residual extraction errors may still propagate into the resulting meta-reviews and should be monitored in real-world deployments.

Acknowledgments

This research is supported by the Collaborative Initiative, Interdisciplinary Graduate Programme (IGP), Nanyang Technological University, Singapore, and the Ministry of Education, Singapore, under its Academic Research Fund (AcRF) Tier 1 grant (RG16/25).

References

Alejandro Ariza-Casabona, Maria Salamó, Ludovico Boratto, and Gianni Fenu. 2023. Towards self-explaining sequence-aware recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 904–911.

Chaitanya Bhatia, Tribikram Pradhan, and Sukomal Pal. 2020. Metagen: An academic meta-review generation system. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 1653–1656, New York, NY, USA. Association for Computing Machinery.

Rose Catherine and William Cohen. 2017. Transnets: Learning to transform for recommendation. In *Proceedings of the eleventh ACM conference on recommender systems*, pages 288–296.

Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. 2018. Neural attentional rating regression with review-level explanations. In *Proceedings of the 2018 world wide web conference*, pages 1583–1592.

Hao Cheng, Shuo Wang, Wensheng Lu, Wei Zhang, Mingyang Zhou, Kezhong Lu, and Hao Liao. 2023a. 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.

Zhiyong Cheng, Sai Han, Fan Liu, Lei Zhu, Zan Gao, and Yuxin Peng. 2023b. Multi-behavior recommendation with cascading graph convolution networks. In *Proceedings of the ACM Web Conference 2023*, pages 1181–1189.

Kyunghyun Cho, Bart Van Merriënboer, Çağlar Gulçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1724–1734.

Xin Dong, Jingchao Ni, Wei Cheng, Zhengzhang Chen, Bo Zong, Dongjin Song, Yanchi Liu, Haifeng Chen, and Gerard De Melo. 2020. Asymmetrical hierarchical networks with attentive interactions for interpretable review-based recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7667–7674.

Shivangi Gheewala, Shuxiang Xu, and Soonja Yeom. 2025. In-depth survey: deep learning in recommender systems—exploring prediction and ranking models, datasets, feature analysis, and emerging trends. *Neural Computing and Applications*, pages 1–73.

Thennakoon Mudiyanselage Anupama Udayangani Gunathilaka, Prabhashrini Dhanushika Manage, Jinglan

Zhang, Yuefeng Li, and Wayne Kelly. 2025. Addressing sparse data challenges in recommendation systems: A systematic review of rating estimation using sparse rating data and profile enrichment techniques. *Intelligent Systems with Applications*, page 200474.

Folasade Olubusola Isinkaye, Yetunde O Folajimi, and Bolande Adefowoke Ojokoh. 2015. Recommendation systems: Principles, methods and evaluation. *Egyptian informatics journal*, 16(3):261–273.

Donghyun Kim, Chanyoung Park, Jinoh Oh, Sungyoung Lee, and Hwanjo Yu. 2016. Convolutional matrix factorization for document context-aware recommendation. In *Proceedings of the 10th ACM conference on recommender systems*, pages 233–240.

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.

Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, page 426–434, New York, NY, USA. Association for Computing Machinery.

Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.

Songyuan Lei, Xinglong Chang, Zhizhi Yu, Dongxiao He, Cuiying Huo, Jianrong Wang, and Di Jin. 2025. Feature-structure adaptive completion graph neural network for cold-start recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 12022–12030.

Lei Li, Li Chen, and Yongfeng Zhang. 2020. Towards controllable explanation generation for recommender systems via neural template. In *Companion proceedings of the web conference 2020*, pages 198–202.

Lei Li, Yongfeng Zhang, and Li Chen. 2021. 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. 2023a. Personalized prompt learning for explainable recommendation. *ACM Transactions on Information Systems*, 41(4):1–26.

Lei Li, Yongfeng Zhang, and Li Chen. 2023b. Prompt distillation for efficient llm-based recommendation. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, pages 1348–1357.

Pan Li, Yuyan Wang, Ed H Chi, and Minmin Chen. 2023c. Prompt tuning large language models on personalized aspect extraction for recommendations. *arXiv preprint arXiv:2306.01475*.

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*, SIGIR '17, page 345–354. ACM.

Yang Li, Kangbo Liu, Ranjan Satapathy, Suhang Wang, and Erik Cambria. 2024. Recent developments in recommender systems: A survey. *IEEE Computational Intelligence Magazine*, 19(2):78–95.

Hao Liao, Shuo Wang, Hao Cheng, Wei Zhang, Jiwei Zhang, Mingyang Zhou, Kezhong Lu, Rui Mao, and Xing Xie. 2025. Aspect-enhanced explainable recommendation with multi-modal contrastive learning. *ACM Transactions on Intelligent Systems and Technology*, 16(1):1–24.

Guanyu Lin, Chen Gao, Yu Zheng, Yinfeng Li, Jianxin Chang, Yanan Niu, Yang Song, Kun Gai, Zhiheng Li, Depeng Jin, and 1 others. 2024. Inverse learning with extremely sparse feedback for recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 396–404.

Chong Liu, Xiaoyang Liu, Lixin Zhang, Feng Xia, and Leyu Lin. 2023. Multi-granularity click confidence learning via self-distillation in recommendation. *arXiv preprint arXiv:2309.16322*.

Donghua Liu, Jing Li, Bo Du, Jun Chang, and Rong Gao. 2019. Daml: Dual attention mutual learning between ratings and reviews for item recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 344–352.

Junrui Liu, Tong Li, Di Wu, Zifang Tang, Yuan Fang, and Zhen Yang. 2025. An aspect performance-aware hypergraph neural network for review-based recommendation. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, WSDM '25, page 503–511. ACM.

Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1930–1939.

Andriy Mnih and Russ R Salakhutdinov. 2007. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc.

Reinald Adrian Pugoy and Hung-Yu Kao. 2021. Unsupervised extractive summarization-based representations for accurate and explainable collaborative filtering. 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 2981–2990.

Jakub Raczyński, Mateusz Lango, and Jerzy Stefanowski. 2023. The problem of coherence in natural language explanations of recommendations. In *ECAI 2023: 26th European Conference on Artificial Intelligence, September 30–October 4, 2023, Kraków, Poland–Including 12th Conference on Prestigious Applications of Intelligent Systems (PAIS 2023)*, pages 1922–1929. SAGE Publications 1 Oliver's Yard, 55 City Road, London, EC1Y 1SP.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Sungyong Seo, Jing Huang, Hao Yang, and Yan Liu. 2017. Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In *Proceedings of the eleventh ACM conference on recommender systems*, pages 297–305.

Jie Shuai, Kun Zhang, Le Wu, Peijie Sun, Richang Hong, Meng Wang, and Yong Li. 2022. A review-aware graph contrastive learning framework for recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pages 1283–1293.

Manasi Vartak, Arvind Thiagarajan, Conrado Miranda, Jeshua Bratman, and Hugo Larochelle. 2017. A meta-learning perspective on cold-start recommendations for items. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ke Wang, Yanmin Zhu, Haobing Liu, Tianzi Zang, and Chunyang Wang. 2023a. Learning aspect-aware high-order representations from ratings and reviews for recommendation. *ACM Transactions on Knowledge Discovery from Data*, 17(1):1–22.

Ke Wang, Yanmin Zhu, Tianzi Zang, Chunyang Wang, Kuan Liu, and Peibo Ma. 2023b. Multi-aspect graph contrastive learning for review-enhanced recommendation. *ACM Transactions on Information Systems*, 42(2):1–29.

Peng Wang, Renqin Cai, and Hongning Wang. 2022. Graph-based extractive explainer for recommendations. In *Proceedings of the ACM Web Conference 2022*, pages 2163–2171.

Tianjun Wei, Tommy W.S. Chow, Jianghong Ma, and Mingbo Zhao. 2023. Expgcn: Review-aware graph convolution network for explainable recommendation. *Neural Networks*, 157:202–215.

Wei Wei, Xubin Ren, Jiabin Tang, Qinyong Wang, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. 2024a. Llmrec: Large language models with graph augmentation for recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, WSDM '24, page 806–815, New York, NY, USA. Association for Computing Machinery.

Yibiao Wei, Yang Xu, Lei Zhu, Jingwei Ma, and Chengmei Peng. 2024b. Multi-level cross-modal contrastive learning for review-aware recommendation. *Expert Systems with Applications*, 247:123341.

Libing Wu, Cong Quan, Chenliang Li, Qian Wang, Bolong Zheng, and Xiangyang Luo. 2019. A context-aware user-item representation learning for item recommendation. *ACM Transactions on Information Systems (TOIS)*, 37(2):1–29.

Wei Yang, Tengfei Huo, Zhiqiang Liu, and Chi Lu. 2023. Review-based multi-intention contrastive learning for recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2339–2343.

Huijing Zhan, Ling Li, Shaohua Li, Weide Liu, Manas Gupta, and Alex C. Kot. 2023. Towards explainable recommendation via bert-guided explanation generator. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Xiaokun Zhang, Bo Xu, Youlin Wu, Yuan Zhong, Hongfei Lin, and Fenglong Ma. 2024. Finerec: Exploring fine-grained sequential recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 2024, page 1599–1608. ACM.

Huachi Zhou, Shuang Zhou, Hao Chen, Ninghao Liu, Fan Yang, and Xiao Huang. 2024. Enhancing explainable rating prediction through annotated macro concepts. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11736–11748.

A LLM Prompt Template

Prompt for Attribute-sentiment Pair Extraction

Analyze the review text.

Requirements:

1. Only consider these attributes: `{ATTRIBUTES_LIST}`
2. For each attribute, extract ONLY ONE opinion and limit it to a maximum of 3 words.
3. If the attribute is not mentioned or sentiment is ambiguous, use `'N/A'`.
4. The opinion must reflect CLEARLY POSITIVE or NEGATIVE sentiments.

`{REVIEW_TEXT}`

Output the extracted attribute-sentiment pairs in JSON format.

B Experiment Details

B.1 Datasets

All four datasets are publicly available for academic research under their respective terms of use, and the pretrained models (BERT and gpt-4.1-nano) are used in accordance with their license terms.

- **Yelp.** [2] This dataset comprises restaurant reviews collected from Yelp. It contains rich textual reviews and user-item interactions, making it suitable for modeling fine-grained review-based recommendations.
- **TripAdvisor.** [3] This dataset includes user reviews on hotels from TripAdvisor. It provides detailed feedback and multi-aspect opinions, which are valuable for explainable recommendations.
- **Amazon Beauty and Amazon Sports.** [4] These two dataset consists of reviews from Amazon. Compared to the above two datasets, they represent a product-centric domain with highly subjective user experiences.

After preprocessing, each of the *Yelp, TripAdvisor, and Amazon Sports* datasets retains approximately 100,000 high-quality user-item interactions with corresponding attribute-sentiment pairs.

B.2 Baseline

- **PMF** (Mnih and Salakhutdinov, 2007) and **SVD++** (Koren, 2008): Classical matrix factorization models that estimate ratings by latent factors of users and items.
- **RGCL** (Shuai et al., 2022): A relation-aware graph contrastive learning framework that constructs opinion-aware user-item graphs to capture semantic dependencies between opinions and interactions.
- **APH** (Liu et al., 2025): It implements review-based recommendations by jointly considering user preferences and item performance on specific aspects with a hypergraph.
- **NETE** (Li et al., 2020): It models interactions between user/item embeddings and opinion words to generate fine-grained explanations.
- **PETER** (Li et al., 2021): A pre-trained transformer-based model for explainable recommendation via a sequence prediction task.
- **PEPLER** (Li et al., 2023a): A prompt-enhanced pre-training model for personalized explanation generation.
- **CER** (Raczyński et al., 2023): A transformer-based explainable recommendation framework focusing on coherent alignment between explanations and ratings.
- **SERMON** (Liao et al., 2025): A multi-modal recommendation framework based on contrastive learning and aspect-level explanations that enhances the interpretability.
- **LLMRec** (Wei et al., 2024a): A recent LLM-based recommender that targets data sparsity by leveraging LLMs to augment the user-item interaction graph with generated edges and node attributes. For a fair comparison with MOSAIC, we replace the original LLM backbone (gpt-3.5-turbo) with *gpt-4.1-nano*, the same model used in our attribute-sentiment extraction pipeline.

B.3 Experimental Settings

Training Setting. To make a fair comparison, each dataset is divided into training, validation, and test sets with a ratio of 8:1:1, where we adopt a per-user split and ensure at least one interaction is assigned to each set for every user. For the training stage, all models are optimized using the Adam optimizer, with the learning rate fixed at 1×10^{-5}. The embedding size for all models is set to 512. The training batch size is fixed at 512. For our

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

Table 6: Computational cost on the Sports dataset.

Method	GPU Memory	Train Time /Epoch	Total Train Time
PETER	1.78 GB	9.3s	1.5 min (10 epochs)
PEPLER	9.59 GB	108.5s	12.7 min (7 epochs)
CER	1.85 GB	9.5s	1.9 min (12 epochs)
SERMON	6.08 GB	37.3s	13.7 min (20 epochs)
APH	15.53 GB	78.3s	8.9 min (7 epochs)
Ours	**1.57 GB**	**8.1s**	**1.3 min (10 epochs)**

model, the number of experts in the MMoE component is set to 3. All experiments are conducted on a single NVIDIA RTX A6000 48GB GPU and are repeated 5 times, with statistical significance analysis conducted to demonstrate the effectiveness of the proposed model.

Evaluation Setting. To assess rating prediction performance, we adopt the standard metrics of **Root Mean Squared Error (RMSE)** and **Mean Absolute Error (MAE)**. For explainability evaluation, our task focuses on attribute-sentiment prediction rather than generating textual explanations. Therefore, we adopt classical metrics from classification tasks— **Accuracy**, **F1 Score**, **Precision**, and **Recall**—to comprehensively evaluate the quality of predicted attribute-sentiment labels. For review generation baseline models, we derive the attribute-sentiment labels from textual explanations generated by each model, following the paradigm introduced in Section 3.1, and supervised by the user-level sentiment label $\mathbf{Y}_{u,i}$ for evaluation.

B.4 Computational Cost Analysis

In this section, we analyze the computational efficiency of the proposed framework. As clarified earlier, the LLM doesn't directly add to our model's training or inference costs. Consequently, the computational cost reported in Table 6 reports only the training-stage cost of the recommendation models and does not include the offline preprocessing.

As shown in Table 6, our model requires relatively low memory consumption and achieves fast training speed compared with representative baselines. The numbers in parentheses denote the number of training epochs determined by early stopping for each method. In particular, the per-epoch training time and total training time of our approach are comparable to lightweight models, and are substantially lower than those of generation-based or complex architectures. These results demonstrate that the proposed framework improves recommendation performance without introducing additional computational overhead, making it suitable for practical deployment.

B.5 Additional Analysis of Gate Distributions

We emphasize that gating does not break down at larger K. Table 7 shows that at $K = 5$ every attribute is still routed to a dominant expert and all five experts are used, yet RMSE degrades from 0.676 to 0.708 (+4.7%). What changes monotonically is the sharpness of specialization: the average effective number of experts per attribute (exp of the gate entropy; 1.0 = a dedicated expert) rises from 1.51 ($K = 3$) to 1.95 ($K = 5$) to 2.39 ($K = 7$). Additional experts do not create new specialists, they fragment existing ones instead. *Rooms*, for instance, is routed almost entirely to one expert at $K = 3$ but is split across two at $K = 5$. Each expert then receives a weaker, more diffuse training signal while the parameter count grows, which is consistent with the reduced sample efficiency we observe. This explains that once experts are fragmented, the model pays the capacity cost of multiple experts without any of them becoming a specialist, which on TripAdvisor is enough to fall slightly below the single-expert setting (RMSE 0.708 / 0.713 vs. 0.696). Fragmentation is most severe at $K = 7$, where the rating gate assigns near-zero weight to Expert 3 even though the attribute gates rely on it heavily, and only 4 of 7 experts are dominant for any attribute.

C Detailed Related Work

C.1 Explainable Recommender Systems

Explanations will increase trust among users towards RSs by providing the recommendation along with a textual explanation. Explanations from extraction-based methods (Wang et al., 2022; Pugoy and Kao, 2021) usually consist of incomplete phrases instead of natural language. With advances of natural language processing, generation-based models, most of which rely on recurrent neural networks, RNN, e.g. LSTM(Cho et al., 2014). NRT(Li et al., 2017) is a strong generation-based model that utilizes an RNN architecture to jointly generate more human-like textual explanations and predict ratings.

PETER(Li et al., 2021) is the first generation-based explainable model that incorporates Transformer architecture to perform explanation generation, and PEPLER (Li et al., 2023a), built upon PETER, employs pre-trained GPT-2 to enhance language understanding and reduce hallucinations. SEQUER (Ariza-Casabona et al., 2023), also extended from PETER, includes the user-review in-

Table 7: Mean gate weights at K=5 and K=7 on TripAdvisor (13,027 test samples). **Bold** marks the dominant expert per row. At K=7, the rating gate assigns near-zero weight (0.008) to one expert, indicating expert redundancy under over-parameterization.

(a) $K = 5$

Task / Gate	E1	E2	E3	E4	E5
Rating	0.181	0.169	0.214	0.215	**0.221**
Service	0.002	**0.988**	0.004	0.000	0.006
Cleanliness	0.009	0.009	0.014	**0.782**	0.186
Rooms	0.004	0.089	**0.648**	0.240	0.019
Location	**0.712**	0.215	0.012	0.014	0.047
Price	0.010	0.046	0.114	0.106	**0.725**
Sleep Quality	0.008	**0.922**	0.003	0.016	0.050

(b) $K = 7$

Task / Gate	E1	E2	E3	E4	E5	E6	E7
Rating	0.089	0.239	0.008	0.034	**0.255**	0.175	0.199
Service	**0.682**	0.017	0.041	0.108	0.090	0.059	0.003
Cleanliness	0.029	0.009	**0.773**	0.038	0.004	0.064	0.083
Rooms	0.038	0.029	0.028	0.008	0.007	**0.883**	0.006
Location	0.003	0.030	**0.667**	0.234	0.043	0.002	0.020
Price	0.050	0.006	**0.781**	0.022	0.017	0.047	0.077
Sleep Quality	0.004	0.006	0.006	**0.719**	0.224	0.006	0.033

teraction sequence as the context of the language model to generate better explanations. POD(Li et al., 2023b) is a Transformer-based model inspired by P5(Shuai et al., 2022). It adopts a pre-trained multitask LLM(Raffel et al., 2020) as its backbone and is fine-tuned to perform top-N recommendation, next-item recommendation, and explanation generation. With the development of LLMs, they have been increasingly employed as sentiment extraction or classification tools in the process of explanation generation(Zhou et al., 2024; Zhang et al., 2024).

To mitigate the hallucination of generation-based models, recent studies have adopted hybrid architectures to improve explanation tasks. ERRA (Cheng et al., 2023a) is composed of a retrieval mechanism to extract relevant attributes from the corpus, followed by a PETER-inspired architecture that generates explanations. ExBERT (Zhan et al., 2023) leverages Sentence-BERT to rank a user's historical reviews and selects top-K to construct pseudo user and item profiles, which are then encoded by a multi-head self-attention encoder to guide the BERT-based explanation generator. By grounding generation on retrieved reviews, it produces more personalized and aspect-aware explanations.

Regardless of whether a model is extraction-based or generation-based, existing approaches ultimately depend on the contextual features of reviews. As a result, they are inevitably subject to issues of hallucination and incompleteness, which makes it difficult to generate explanations that comprehensively cover all relevant attributes due to the inherent limitations of text generation models.

C.2 Review-based Recommendataion

Traditional RSs learn features using latent factor models that take user IDs and item IDs as input (Koren et al., 2009). However, these static feature learning methods suffer from data sparsity and scalability issues. Review-based recommendation aims to enhance rating prediction performance by leveraging the fine-grained semantic and opinion information embedded in user-generated reviews.

Early CNN-based methods leverage CNNs to extract textual features from reviews. ConvMF(Kim et al., 2016) integrates CNN-based textual features into probabilistic matrix factorization (PMF) to enhance the accuracy of rating prediction. TransNet(Catherine and Cohen, 2017) introduces an additional transform layer that maps user and item features to an approximation of target review representation, thereby mitigating the reliance on reviews at test time.

By integrating an attention mechanism, NARRE(Chen et al., 2018) identifies the most useful features from individual reviews, thereby improving both accuracy and interpretability. D-Attn(Seo et al., 2017) extends this idea by applying dual attention across and within reviews, while DAML(Liu et al., 2019) applies dual and mutual attention to model interactions between user reviews and rating features. Similarly, CARL(Wu et al., 2019) also jointly leverages textual and numerical signals to improve rating prediction performance.

GNNs can capture the high-order relationships and model complicated user-item interactions by leveraging both semantic and structural information. AHOR(Wang et al., 2023a) and APH(Liu et al., 2025) capture aspects from reviews to model users' fine-grained preferences and item representations through GNNs. RGCL(Shuai et al., 2022) and RMCL(Yang et al., 2023) utilize contrastive learning techniques to enhance user-item embeddings and interaction modeling through the integration of review data, while MAGCL(Wang et al., 2023b) improves preference granularity by disentangling reviews into multiple semantic aspects for contrastive learning. MCCL(Wei et al., 2024b) fur-

ther improves the capability of leveraging semantic information in a self-supervised manner.

PLLMPAR(Li et al., 2023c) extracts key aspects that reflect user preferences and item properties using a fine-tuned language model. This study paves the way for further exploration and development at the intersection of LLMs and personalized recommendation.