HyperXRec: Unifying Preference Clusters and LLM Experts for Robust Explainable Recommendations

Xue Han[1], **Zhiwen Luo**[2,1], **Zhixiang Li**[1], **Nizar Bouguila**[2], **Weifeng Su**[1] and **Wentao Fan**[1*]
[1]Beijing Normal-Hong Kong Baptist University, Zhuhai, China
[2]Concordia University, Montreal, Canada
u430201701@mail.bnbu.edu.cn, zhiwen.luo@mail.concordia.ca, u430201707@mail.bnbu.edu.cn, nizar.bouguila@concordia.ca, wfsu@bnbu.edu.cn, wentaofan@bnbu.edu.cn

Abstract

Explainable recommendation is crucial for building user trust, yet producing natural-language rationales that faithfully reflect the underlying decision process remains challenging. Most LLM-based explainable recommenders incorporate collaborative signals through shallow prompting or lightweight adapters, which often yields generic explanations that are only loosely grounded in preference evidence, particularly when interactions are sparse. To address this gap, we propose HyperXRec, a novel framework that unifies preference modeling and explanation generation by integrating hyperspherical latent clustering with a cluster-guided mixture-of-experts (MoE) inside an LLM. Specifically, HyperXRec employs a dual-VAE with a von Mises-Fisher mixture prior to learn robust hyperspherical user/item embeddings, resulting in stable and semantically coherent clusters even under sparse interaction settings. These clusters are subsequently leveraged to guide fine-grained expert routing in the LLM, encouraging explanations that are both personalized and explicitly aligned with the learned preference structure. Experiments on multiple benchmarks demonstrate that HyperXRec consistently outperforms strong baselines in explanation fidelity, personalization, and robustness in cold-start scenarios.

1 Introduction

A long-standing goal in recommender systems is to move beyond black-box prediction and provide recommendations that are interpretable. Users are more likely to trust and adopt recommendations when they are accompanied by clear, personalized justifications. Accordingly, explainable recommendation has attracted growing attention as a means to improve transparency and user satisfaction [Zhang and Chen, 2020].

Large Language Models (LLMs) have emerged as a compelling foundation for generating fluent and context-aware explanations. Early LLM-based approaches produced plausible rationales by conditioning text generation on coarse user or item information [Li *et al.*, 2021]. Subsequent work incorporated richer signals (such as interaction histories and profile attributes) to strengthen personalization [Ma *et al.*, 2024]. More recently, several methods have attempted to inject collaborative filtering (CF) signals directly into LLMs through lightweight adapters, often implemented via Mixture-of-Experts (MoE) [Tang *et al.*, 2024]. This line of research aims to unify preference learning and language generation so that explanations are not only linguistically natural but also grounded in actual recommendation logic. Despite these advances, existing methods still struggle to faithfully align recommendation reasoning with the generated explanations [Brita *et al.*, 2025].

This limitation largely stems from a persistent misalignment between collaborative signals and natural language generation. In many existing frameworks, CF information is incorporated only superficially (e.g., as soft prompts or fixed embeddings), rather than being integrated into the LLM's internal reasoning and control pathways [Ma *et al.*, 2024]. Consequently, an LLM can produce fluent explanations that mention relevant item attributes, yet these explanations may remain loosely connected to the true factors driving the recommendation. Attempts to enhance representation power by

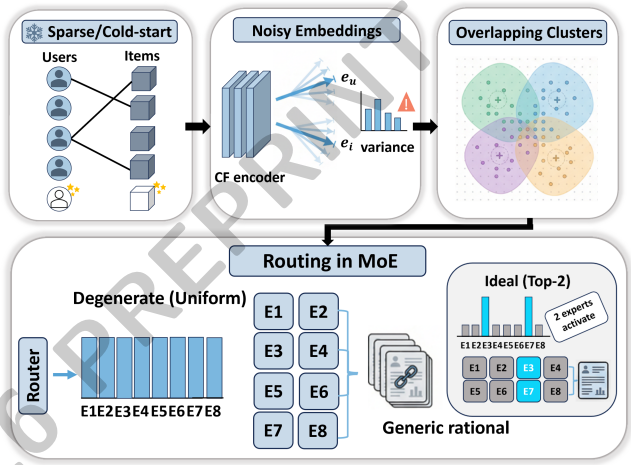


Figure 1: Sparse or cold-start conditions induce high-variance embeddings and overlapping clusters, leading to degenerate expert routing in MoE architectures and generic explanation generation.

*Corresponding author.

incorporating additional components, such as Graph Neural Networks (GNNs) to model higher-order user-item relationships, have not yielded substantial gains in explanation quality [Li *et al.*, 2025d], and recent analyses suggest diminishing returns beyond the use of basic CF signals [Brita *et al.*, 2025]. This misalignment becomes even more pronounced in cold-start or sparse-data settings, where user interaction signals are inherently noisy or scarce [Li *et al.*, 2025a]. In such cases, it becomes difficult to learn consistent, preference-aligned representations to guide explanation generation. Moreover, prior efforts to impose structure on the latent preference space, such as clustering users via variational autoencoders (VAEs) [Cai and Cai, 2022] or Gaussian Mixture Models (GMMs) [Tang *et al.*, 2024], often struggle under sparse and high-dimensional conditions. These models frequently yield unstable or semantically incoherent clusters, providing unreliable guidance for downstream expert routing. As illustrated in Figure 1, under sparse or cold-start conditions, collaborative representations learned by standard CF encoders exhibit high variance, resulting in noisy embeddings and overlapping clusters that degrade routing in MoE-based architectures. Without reliable preference representations, routing tends to collapse to coarse, near-uniform expert activation rather than selective Top-m routing, leading to generic explanations that fail to capture fine-grained preferences. This reveals a fundamental gap between recommendation reasoning and explanation generation, especially under sparse interaction data.

To address these challenges, we propose HyperXRec, a unified framework for explainable recommendation that uses hyperspherical preference clusters as explicit routing signals to tightly couple collaborative preference modeling with LLM-based explanation generation. Specifically, HyperXRec employs a dual-VAE architecture equipped with a von Mises-Fisher Mixture Model (vMFMM) prior [Yang *et al.*, 2021; Fan *et al.*, 2026] to learn user and item embeddings on the unit hypersphere. To obtain robust and semantically meaningful latent structure under sparse interactions, we further introduce a perturbation-based cosine-consistency regularization that stabilizes the learned preference clusters. Building on this structured representation, HyperXRec integrates a fine-grained MoE module into the LLM, where expert activation is conditioned on inferred cluster assignments rather than generic or uniform routing. As a result, users with similar latent preference profiles activate similar expert pathways, encouraging explanations that are both personalized and faithfully aligned with the underlying collaborative reasoning. Our main contributions are as follows.

- We introduce a hyperspherical clustering for user-item interactions based on a dual-VAE with a vMFMM prior and cosine-consistency regularization, enabling stable preference clustering under sparse regimes.
- We propose a cluster-guided MoE integration for LLM-based explainers, where expert routing is conditioned on inferred cluster assignments to align explanation generation with collaborative preference structure.
- Extensive experiments on three real-world datasets demonstrate that HyperXRec improves explanation quality and cold-start robustness compared with strong LLM-based explainable recommendation baselines.

2 Related Work

2.1 Explainable Recommendation

Early recommender systems focused on content-based filtering, where explicit item attributes such as categories or brands were used to provide intuitive explanations to users [Pazzani and Billsus, 2007; Tintarev and Masthoff, 2007]. With the rise of CF, user-based and item-based methods leveraged similarity and co-occurrence patterns to generate recommendations with simple yet interpretable rationales [Koren *et al.*, 2009]. To enhance semantic interpretability, subsequent studies incorporated user-generated content such as reviews, aligning latent factors with interpretable textual aspects [McAuley and Leskovec, 2013]. Beyond feature-level explanations, graph-based methods and attention mechanisms were explored to extract interpretable signals from user-item interactions and associated textual evidence [Li *et al.*, 2025b]. While these approaches improve transparency to some extent, they often rely heavily on auxiliary data quality and primarily capture correlations rather than the underlying reasoning process, limiting robustness under sparse or complex settings.

With the rapid development of LLMs, explainable recommendation has entered a new stage. Existing LLM-based methods typically inject collaborative filtering signals via prompts or adapters as external conditions, improving explanation fluency and surface plausibility, especially in zero-shot or sparse-data settings [Li *et al.*, 2023; Ma *et al.*, 2024; Zhu *et al.*, 2024]. However, because such signals are not integrated into the core reasoning process of LLMs, the resulting explanations are often weakly aligned with the underlying recommendation logic [Brita *et al.*, 2025]. More recent work explores MoE architectures to more tightly couple preference modeling with generation, for example by routing user-item pairs to specialized experts [Tang *et al.*, 2024]. Nevertheless, existing approaches typically rely on Gaussian latent spaces and coarse or heuristic routing, limiting the alignment between expert activation and fine-grained user preference semantics [Fedus *et al.*, 2022].

2.2 Latent Preference Modeling and Representation Learning

Beyond explainable recommendation, a complementary line of research focuses on preference modeling and representation learning, aiming to derive more stable and structured representations of user interests. Many approaches employ VAEs [Cai and Cai, 2022; Han *et al.*, 2025] or GMMs [Tang *et al.*, 2024] to model user preferences in a distributional form, capturing group-level patterns and improving robustness under sparse interactions. Building on this line of research, some methods further introduce mixture-based or hierarchical structures under multi-intent assumptions to capture diverse interests and uncover semantically meaningful user groups [Li *et al.*, 2025c; Qiu *et al.*, 2025]. Despite their improved expressiveness and recommendation performance, these latent representations are rarely leveraged to directly guide natural language explanation [Li *et al.*, 2025e;

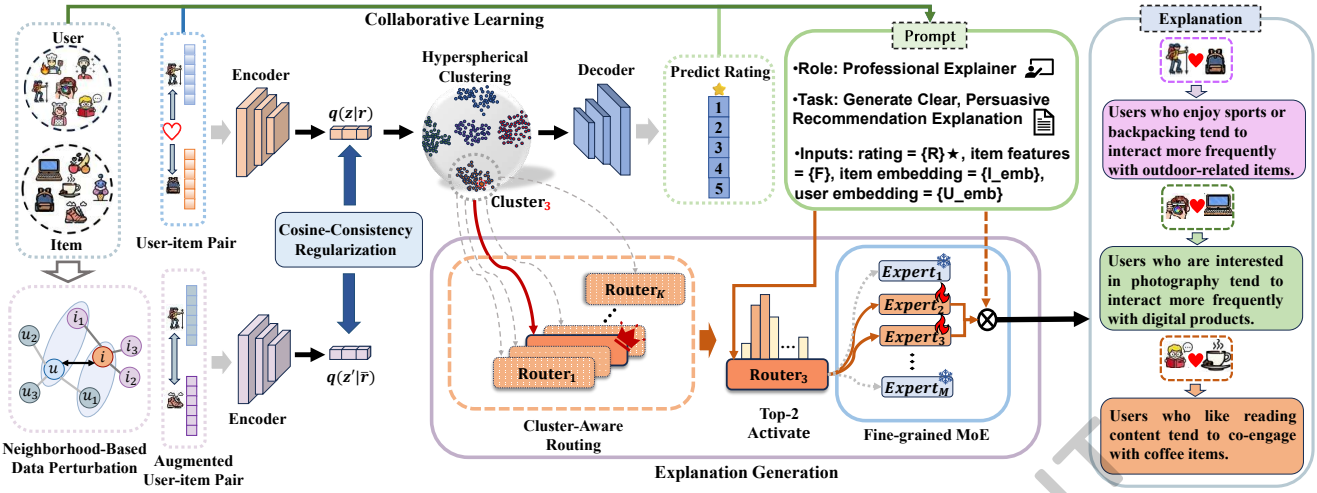


Figure 2: Overview of the proposed HyperXRec framework for cluster-guided explainable recommendation.

Luo *et al.*, 2026]. Even in the few attempts that connect latent preferences with text generation, the coupling remains loose, limiting their effectiveness as stable and controllable signals for faithful explanation generation [Li *et al.*, 2020].

3 Proposed Method

3.1 Preliminaries and Framework Overview

We consider a dataset $\mathcal{D} = \{(u_n, i_n, r_{u_n,i_n}, \mathbf{f}_{i_n})\}_{n=1}^{N}$, where each interaction instance includes a user ID $u \in \mathcal{U}$, an item ID $i \in \mathcal{I}$, an observed rating $r_{u,i} \in \mathbb{R}_{>0}$ indicating u's preference for i, and a feature vector $\mathbf{f}_i \in \mathcal{F}$ describing item attributes. Given a user-item pair (u, i), a recommender estimates a preference score $y_{u,i}$. Beyond preference estimation, our goal is to generate a natural language explanation for each recommended item. Formally, the explanation is defined as:

$$\text{explanation}(u, i) = \text{generator}(u, i, r_{u,i}, \mathbf{f}_i, \mathcal{X}_u), \qquad (1)$$

where $\mathcal{X}_u$ denotes the collaborative context of user u, and $\text{generator}(\cdot)$ is instantiated as a large language model.

HyperXRec aims to generate faithful and personalized natural language explanations by explicitly aligning latent user preference structures with the reasoning process of a language model. The framework consists of two core components: (1) a hyperspherical deep clustering module that models user-item interactions on the unit hypersphere to learn compact and semantically coherent preference clusters, with a neighborhood-based consistency regularization to improve robustness under sparse interactions; and (2) a cluster-guided explanation generation module built on a MoE language model, where the learned preference clusters explicitly guide expert routing, enabling different experts to specialize in distinct preference semantics. The overall architecture of HyperXRec is illustrated in Figure 2.

3.2 Hyperspherical Representation Learning

To capture structured user-item preference patterns under sparse interactions, we adopt a dual-VAE framework to learn hyperspherical latent representations. Unlike Euclidean latent spaces, the hypersphere emphasizes angular similarity, which better reflects preference alignment and facilitates semantic clustering [Yang *et al.*, 2021; Luo *et al.*, 2025]. For each user-item pair (u, i) with embeddings $\mathbf{u}$ and $\mathbf{i}$, the probabilistic encoder defines a variational posterior $q(\mathbf{z} \mid \mathbf{u}, \mathbf{i})$ as a von Mises-Fisher (vMF) distribution [Luo *et al.*, 2025]. The vMF distribution is defined on the unit hypersphere $\mathbb{S}^{d-1}$ with density:

$$\mathcal{V}(\mathbf{x} \mid \boldsymbol{\mu}, \kappa) = C_d(\kappa) \exp\big(\kappa \boldsymbol{\mu}^\top \mathbf{x}\big), \qquad (2)$$

where $\boldsymbol{\mu}$ denotes the mean direction, κ controls the concentration around $\boldsymbol{\mu}$, and $C_d(\kappa)$ is the normalization constant ensuring the density integrates to 1 over the sphere $\mathbb{S}^{d-1}$. We then parameterize the variational posterior as:

$$q(\mathbf{z} \mid \mathbf{u}, \mathbf{i}) = \mathcal{V}(\mathbf{z} \mid \widehat{\boldsymbol{\mu}}_{u,i}, \widehat{\kappa}_{u,i}), \qquad (3)$$

where the mean direction $\widehat{\boldsymbol{\mu}}_{u,i}$ and concentration $\widehat{\kappa}_{u,i}$ are produced by a neural network $h(\cdot; \psi)$:

$$[\widehat{\boldsymbol{\mu}}_{u,i};\, \ln \widehat{\kappa}_{u,i}] = h(\mathbf{u}, \mathbf{i}; \psi). \qquad (4)$$

Given $\mathbf{z}$, the decoder reconstructs the observed rating $r_{u,i}$ using a categorical likelihood:

$$p(r_{u,i} \mid \mathbf{z}) = \text{Cat}(r_{u,i} \mid \text{softmax}(f(\mathbf{z}; \theta))), \qquad (5)$$

where $f(\cdot; \theta)$ outputs logits over the discrete rating space.

To impose clustering structure, we introduce a vMFMM prior with an explicit cluster variable $k \in \{1, \ldots, K\}$:

$$p(k) = \text{Cat}(k \mid \boldsymbol{\pi}), \qquad p(\mathbf{z} \mid k) = \mathcal{V}(\mathbf{z} \mid \boldsymbol{\mu}_k, \kappa_k), \qquad (6)$$

which yields the marginal prior:

$$p(\mathbf{z}) = \sum_{k=1}^{K} \pi_k \, \mathcal{V}(\mathbf{z} \mid \boldsymbol{\mu}_k, \kappa_k). \qquad (7)$$

Each mixture component corresponds to a semantically coherent preference cluster on the unit hypersphere. Accordingly, the joint distribution for an interaction is

$$p(r_{u,i}, \mathbf{z}, k) = p(r_{u,i} \mid \mathbf{z})\, p(\mathbf{z} \mid k)\, p(k). \qquad (8)$$

We optimize the evidence lower bound (ELBO) with a mean-field posterior $q(\mathbf{z}, k \mid \mathbf{u}, \mathbf{i}) = q(\mathbf{z} \mid \mathbf{u}, \mathbf{i})\, q(k \mid \mathbf{u}, \mathbf{i})$:

$$\mathcal{L}_{\mathrm{c}} = \mathbb{E}_{q(\mathbf{z},k|\mathbf{u},\mathbf{i})}\Big[\log p(r_{u,i} \mid \mathbf{z}) + \log p(\mathbf{z} \mid k) + \log p(k) - \log q(\mathbf{z} \mid \mathbf{u}, \mathbf{i}) - \log q(k \mid \mathbf{u}, \mathbf{i})\Big]. \tag{9}$$

The cluster posterior is computed as

$$q(k \mid \mathbf{u}, \mathbf{i}) \propto \pi_k \exp\big(\kappa_k\, \boldsymbol{\mu}_k^{\top}\, \boldsymbol{\xi}_{u,i}\big), \tag{10}$$

where $\pi_k \geq 0$, $\sum_{k=1}^{K} \pi_k = 1$, and $\boldsymbol{\xi}_{u,i} = \mathbb{E}_{q(\mathbf{z}|\mathbf{u},\mathbf{i})}[\mathbf{z}]$ denotes the posterior mean direction. Since the vMF posterior does not admit a standard reparameterization, we adopt the rejection-sampling-based reparameterization of Davidson et al. [Davidson *et al.*, 2018] to enable backpropagation.

We use the maximum a posteriori (MAP) cluster index $k^{\star} = \arg\max_k q(k \mid \mathbf{u}, \mathbf{i})$ as the interaction-level semantic label, which will serve as the routing signal for the cluster-guided MoE explainer (**Section 3.4**). This formulation follows prior work on hyperspherical VAEs and vMF-based deep clustering [Guo *et al.*, 2026b; Guo *et al.*, 2026a].

3.3 Neighborhood Consistency Regularization

Under sparse interactions, small perturbations in user-item inputs may lead to disproportionate changes in hyperspherical embeddings and cluster assignments, which can destabilize downstream expert routing. To enhance the robustness of hyperspherical preference representations, we introduce a neighborhood-based data perturbation strategy during training [El-Kishky *et al.*, 2023]. Instead of perturbing latent representations directly, we perform perturbations at the input level by modifying user-item pairs, which better preserves collaborative semantics.

Specifically, for a given user-item pair (u, i), we first extract their embedding representations from a pretrained encoder and construct a k-nearest-neighbor (kNN) structure based on cosine similarity. Based on this neighborhood structure, we generate perturbed pairs by either replacing the item with a neighboring item i_j interacted by the same user, yielding (u, i_j), or replacing the user with a neighboring user u_k who interacted with the same item, yielding (u_k, i). The perturbed pairs are then re-encoded with the encoder in Section 3.2 to obtain $\mathbf{z}'$.

To encourage local stability of hyperspherical representations, we enforce a cosine-based consistency regularization [Sinha and Dieng, 2021] between the original latent representation $\mathbf{z}$ and its perturbed counterpart $\mathbf{z}'$, defined as

$$\mathcal{L}_{\mathrm{dp}} = 1 - \cos(\mathbf{z}, \mathbf{z}'), \tag{11}$$

where $\mathbf{z}$ and $\mathbf{z}'$ denote the mean directions of the vMF posterior. This regularization encourages neighboring interactions to preserve angular proximity in the latent space. The final objective in the clustering phase is defined as:

$$\mathcal{L}_{\mathrm{cluster}} = \mathcal{L}_{\mathrm{c}} + \lambda\, \mathcal{L}_{\mathrm{dp}}, \tag{12}$$

where λ balances reconstruction accuracy and consistency regularization.

3.4 Cluster-guided Explanation Generation Module

Cluster-aware Expert Routing We propose a cluster-guided MoE architecture, where expert routing is explicitly conditioned on semantic preference clusters learned in the hyperspherical latent space. Given a user-item interaction (u, i), we obtain its cluster responsibility k from the hyperspherical clustering module according to Eq. (10). We use the MAP cluster index $k^{\star} = \arg\max_k q(k \mid \mathbf{u}, \mathbf{i})$ as the routing signal during explanation generation. This cluster index serves as a high-level semantic prior that governs expert routing. The cluster assignment is interaction-level and shared across all tokens, while expert selection is performed at the token level within the cluster-specific router [Zhou *et al.*, 2022].

During fine-tuning, we replace the feed-forward network (FFN) in each Transformer block of the LLM with a cluster-aware MoE layer [Lepikhin *et al.*, 2021]. For a token representation $\boldsymbol{h} \in \mathbb{R}^H$, the router associated with cluster $k^{\star}$ computes routing scores as:

$$\boldsymbol{r}_{k^{\star}} = \mathrm{softmax}(\mathbf{W}^{(k^{\star})}\boldsymbol{h}), \tag{13}$$

where $\mathbf{W}^{(k^{\star})} \in \mathbb{R}^{E \times H}$ denotes the routing parameters for cluster $k^{\star}$. Only the router corresponding to the assigned cluster is activated, and the top-m experts with the highest routing probabilities are selected for computation. This design enforces cluster-level semantic consistency in expert activation and avoids noisy cross-cluster routing.

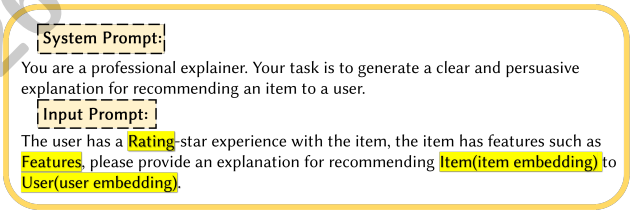


Figure 3: Prompt template for explanation generation.

Explanation Generation and Training Each expert is implemented as a lightweight gated FFN with element-wise multiplicative interactions [So *et al.*, 2021]:

$$e(\boldsymbol{h}) = \boldsymbol{W}^{(2)}\, \sigma\Big(\boldsymbol{W}^{(1)}\boldsymbol{h} \odot \boldsymbol{W}^{(3)}\boldsymbol{h}\Big), \tag{14}$$

where $\sigma(\cdot)$ denotes the SiLU activation function [Dai *et al.*, 2024]. The outputs of the selected experts are aggregated according to their routing weights:

$$\boldsymbol{h}' = \sum_{e \in \Omega_m} w_e \cdot e(\boldsymbol{h}), \tag{15}$$

where Ω_m denotes the set of activated experts and w_e are the normalized routing weights.

To ground explanation generation, we reuse pretrained user and item embeddings from the encoder by replacing placeholder tokens (e.g., *[UserToken]* and *[ItemToken]*) in the prompt, and append rating information and item attributes in natural language form; the prompt relies only on historical interactions, item-side information, and the predicted

item, without accessing any ground-truth review of the target interaction. An example of the prompt structure is illustrated in Figure 3. While prompts provide content grounding, cluster-guided routing explicitly controls the internal generation pathways of the LLM. The explanation generator is fine-tuned using LoRA [Hu *et al.*, 2022] adapters for parameter efficiency. Given the constructed prompt $\mathbf{x}_{u,i}$ and the target explanation $\mathbf{y}_{u,i}$, the explanation loss is defined as

$$\mathcal{L}_{\text{exp}} = -\log p(\mathbf{y}_{u,i} \mid \mathbf{x}_{u,i}, k^{\star}). \tag{16}$$

Due to space limits, the full training process and algorithm are provided in the supplementary material.

4 Experiments

4.1 Experimental Settings

Datasets We evaluate our method on three widely used real-world benchmark datasets for explainable recommendation: Amazon (Movies & TV), Yelp, and TripAdvisor [Tang *et al.*, 2024; Li *et al.*, 2025d]. These datasets span multiple application domains, including e-commerce, local business recommendation, and tourism, and provide user-item interactions with explicit ratings and accompanying textual reviews. Dataset statistics after preprocessing are reported in Table 1.

	Amazon (AZ)	TripAdvisor (TA)	Yelp
Users	7,506	9,765	27,147
Items	7,360	6,280	20,266
Reviews	441,783	320,023	1,293,247
Features	5,399	5,069	7,340
Avg. reviews / user	58.86	32.77	47.64
Avg. reviews / item	60.02	50.96	63.81
Avg. words / expl.	14.14	13.01	12.32

Table 1: Statistics of the datasets used in our experiments.

Evaluation Metrics. Following common practice in explainable recommendation and text generation, we evaluate explanation quality using a set of widely adopted automatic metrics, covering surface-level similarity, semantic relevance, and diversity. Specifically, we report BLEU and ROUGE for n-gram overlap, BERTScore (P/R/F) [Zhang *et al.*, 2020], BLEURT [Sellam *et al.*, 2020], and GPTScore [Wang *et al.*, 2023] to assess semantic similarity and fluency, and Distinct-n (Dist-1/2) together with Unique Sentence Ratio (USR) [Li *et al.*, 2021] to evaluate explanation diversity. All metrics are computed on the test set, where higher values indicate better performance.

Baselines. We evaluate HyperXRec on real-world benchmarks and compare it with representative state-of-the-art explainable recommendation methods, which can be categorized into three groups. **(1) Neural explanation generation methods:** PETER [Li *et al.*, 2021] and NETE [Li *et al.*, 2020]. **(2) Latent preference modeling-based explainable recommenders:** PEVAE [Cai and Cai, 2022] and GaVaMoE [Tang *et al.*, 2024]. **(3) LLM-based explainable recommendation methods:** PEPLER [Li *et al.*, 2023], XRec [Ma *et al.*, 2024], and G-Refer [Li *et al.*, 2025d].

Implementation Details. All experiments are implemented in Python with GPU acceleration using DeepSpeed ZeRO Stage-3 [Rajbhandari *et al.*, 2020], with detailed hardware configurations reported in the supplementary materials. For VAE pretraining, we adopt a learning rate of 1×10^{-3} with a batch size of 1024 for 300 epochs, followed by clustering fine-tuning with a learning rate of 1×10^{-5} for 30 epochs. For explanation generation, we fine-tune LLaMA-3 using LoRA [Hu *et al.*, 2022] with a learning rate of 3×10^{-5}, a batch size of 1 and gradient accumulation of 16, for 5 epochs. Additional experimental analyses on clustering behavior and computational cost are provided in the supplementary material.

4.2 Performance Evaluation

Performance of HyperXRec As shown in Table 2, HyperXRec achieves strong and consistent improvements over state-of-the-art baselines across both lexical- and semantic-level evaluation metrics, highlighting its effectiveness for explainable recommendation. In particular, HyperXRec demonstrates substantial gains in semantic consistency, as reflected by the BERT-F metric, outperforming the second-best baseline by +0.390, +0.431, and +0.144 on Amazon, TripAdvisor, and Yelp, respectively. These improvements stem from the proposed cluster-guided expert routing mechanism, which injects structured preference information into the LLM's generation process, enabling experts to specialize in distinct preference semantics and produce explanations better aligned with the recommendation logic. Notably, HyperXRec attains a perfect USR of 1.0 on all datasets, indicating that the model avoids degeneration into repetitive explanations.

Effect of the Number of Routers (G) Table 3 further studies the impact of varying the number of routers (G) in the cluster-guided MoE explainer. Overall, moving from coarse to moderate routing granularity improves explanation quality, as a larger G encourages expert specialization and enables the model to better capture heterogeneous user preferences. For instance, the best performance is achieved with $G = 4$ on Amazon and $G = 5$ on Yelp. However, the benefit is not monotonic. When G becomes too large, each router receives fewer interactions, resulting in insufficient expert training and weakened semantic coherence. This phenomenon is most evident on TripAdvisor, where increasing G from 3 to 4 causes a clear degradation (e.g., BLEU-1 drops from 22.658 to 14.963 and GPTScore from 80.43 to 60.31). These results indicate that G should be carefully chosen to balance routing granularity and effective expert utilization.

4.3 Ablation Studies

To examine the impact of hyperspherical modeling, we compare vMFMM with GMM under the same dual-VAE architecture. Results on the Amazon dataset (Table 4) show consistent improvements across both lexical and semantic metrics when using vMFMM, with similar trends observed on other datasets. To further understand these improvements, Figure 4 visualizes the learned latent spaces using t-SNE under different priors. The GMM prior yields overlapping clusters that blur cluster boundaries and destabilize routing. In contrast, the vMFMM prior enforces directional constraints on the hypersphere, producing compact, well-separated clusters that

Data	Method	N-gram metrics				Semantic & diversity metrics							
		BLEU-1	BLEU-4	ROUGE-1	ROUGE-L	Dist-1	Dist-2	BERT-P	BERT-R	BERT-F	GPTScore	BLEURT	USR
AZ	PEPLER	12.564	0.991	14.005	10.816	17.134	61.190	0.369	0.362	0.364	76.08	-0.2950	0.9563
	PETER	14.796	1.201	15.335	11.614	17.431	62.790	0.384	0.376	0.380	77.65	-0.2937	0.8480
	PEVAE	16.349	1.503	19.255	15.853	18.440	60.610	0.349	0.340	0.345	79.87	-0.3302	0.8715
	NETE	14.965	1.152	16.470	12.982	14.503	47.312	0.344	0.344	0.344	75.63	-0.4073	0.5413
	GaVaMoE	18.224	1.498	23.697	16.975	19.470	63.454	0.388	0.370	0.373	80.45	-0.1850	0.9632
	G-Refer	20.800	1.710	25.150	17.650	20.950	65.000	0.407	0.450	0.429	82.12	-0.1203	**1.0000**
	XRec	20.259	1.730	24.971	17.737	20.370	64.444	0.409	0.398	0.401	82.57	-0.1061	**1.0000**
	Ours	**21.514**	**1.850**	**25.402**	**18.607**	**28.340**	**66.958**	**0.822**	**0.816**	**0.819**	**88.69**	**-0.0835**	**1.0000**
TA	PEPLER	13.392	0.965	15.382	12.078	18.124	64.923	0.345	0.303	0.343	63.91	-0.2934	0.7583
	PETER	14.957	0.881	16.514	13.002	19.400	65.652	0.354	0.325	0.350	67.00	-0.3316	0.8750
	PEVAE	15.705	1.401	18.675	15.411	18.182	66.414	0.369	0.363	0.368	69.94	-0.3354	0.9143
	NETE	13.847	1.186	14.891	11.880	14.854	48.431	0.300	0.293	0.295	61.54	-0.4116	0.2677
	GaVaMoE	17.270	1.735	25.314	18.392	20.466	65.332	0.301	0.296	0.263	69.10	-0.3220	0.9050
	G-Refer	19.350	1.720	24.100	17.450	20.900	66.200	0.405	0.400	0.402	73.20	-0.2100	**1.0000**
	XRec	19.642	1.650	23.672	16.871	21.113	65.421	0.386	0.380	0.384	71.81	-0.2486	**1.0000**
	Ours	**22.658**	**2.747**	**26.601**	**18.517**	**27.442**	**67.573**	**0.831**	**0.834**	**0.833**	**80.43**	**-0.1150**	**1.0000**
Yelp	PEPLER	9.448	0.623	12.604	9.769	13.321	43.577	0.317	0.294	0.305	61.58	-0.2892	0.8660
	PETER	8.072	0.574	12.965	9.850	14.240	52.891	0.298	0.271	0.281	65.16	-0.3375	0.4757
	PEVAE	14.250	1.217	16.229	13.891	16.890	58.510	0.328	0.323	0.326	67.34	-0.2546	0.8446
	NETE	11.314	0.823	14.041	9.226	13.245	44.426	0.222	0.137	0.144	58.27	-0.4838	0.2533
	GaVaMoE	16.225	1.396	22.695	16.876	17.471	61.455	0.369	0.351	0.352	68.10	-0.2180	0.9950
	G-Refer	19.210	1.680	23.400	17.300	18.200	62.600	0.363	0.437	0.401	75.16	-0.1336	**1.0000**
	XRec	19.379	1.601	22.870	16.531	17.560	61.190	0.395	0.351	0.373	69.12	-0.2026	0.9993
	Ours	**20.284**	**1.942**	**24.568**	**18.227**	**22.365**	**65.194**	**0.512**	**0.589**	**0.545**	**83.25**	**-0.0761**	**1.0000**

Table 2: Overall performance on three datasets. Best and second-best results are highlighted with red and purple.

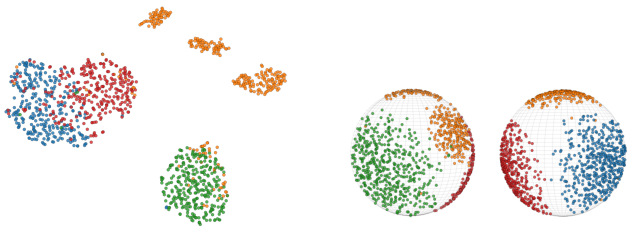
Data	G	N-gram metrics				Semantic metrics							
		BLEU-1	BLEU-4	ROUGE-1	ROUGE-L	Dist-1	Dist-2	BERT-P	BERT-R	BERT-F	GPTScore	BLEURT	USR
AZ	2	18.910	1.508	22.187	15.313	24.885	62.868	**0.822**	0.812	0.810	82.71	-0.1647	0.9342
	3	19.342	1.462	22.458	15.736	26.873	64.877	0.810	0.814	0.812	84.99	-0.1120	0.9572
	4	**21.514**	**1.850**	**25.402**	**18.607**	**28.340**	**66.958**	**0.822**	**0.816**	**0.819**	**88.69**	**-0.0835**	**1.0000**
	5	19.286	1.504	22.286	15.487	26.876	64.871	0.815	0.815	0.815	85.34	-0.1169	0.9545
TA	2	21.141	2.527	25.995	17.759	25.841	63.885	0.822	0.833	0.828	78.26	-0.1613	0.9591
	3	**22.658**	**2.747**	**26.601**	**18.517**	**27.442**	**67.573**	**0.8313**	**0.834**	**0.833**	**80.43**	**-0.1150**	**1.0000**
	4	14.963	1.358	21.964	13.965	16.760	50.873	0.821	0.824	0.823	60.31	-0.3310	0.9719
	5	19.863	2.365	25.546	17.016	22.824	65.886	0.817	0.832	0.825	71.54	-0.1906	0.9717
Yelp	2	16.784	1.365	21.532	15.873	19.426	60.774	0.368	0.352	0.360	73.25	-0.1831	0.9384
	3	17.942	1.529	22.614	16.723	20.417	62.143	0.381	0.366	0.374	76.52	-0.1540	0.9527
	4	18.585	1.687	23.231	17.104	21.392	63.432	0.389	0.375	0.382	79.06	-0.1246	0.9642
	5	**20.284**	**1.942**	**24.568**	**18.227**	**22.365**	**65.194**	**0.512**	**0.589**	**0.545**	**83.25**	**-0.0761**	**1.0000**

Table 3: Varying number of routers G on three datasets. Best and second-best results are highlighted with red and purple.

stabilize downstream expert routing. Since cluster labels explicitly control expert routing in our model, these consistent improvements indicate that higher-quality clustering directly contributes to better explanation generation.

Prior	BLEU-4	ROUGE-L	Dist-1	BERT-F
GMM	1.529	17.357	19.820	0.403
vMFMM	**1.850**	**18.607**	**28.340**	**0.819**

Table 4: Comparison of GMM and vMFMM priors on Amazon.



(a) GMM Prior (b) vMFMM Prior

Figure 4: t-SNE visualization of latent representations on Amazon.

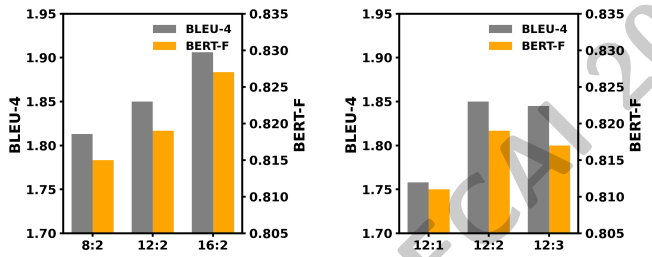
4.4 Hyperparameter Sensitivity Analysis

Experts E and Activated Experts m. We study the impact of expert configuration on explanation quality on the Amazon dataset, where E denotes the total number of experts and m denotes the number of activated experts. Figures 5 (a) and 5 (b) show that increasing E from small to moderate values consistently improves BLEU-4 and BERT-F across the settings shown in the figure, while further increasing E yields limited gains at higher computational and memory cost. Activating multiple experts is beneficial: $m = 2$ consistently outperforms the single-expert setting, whereas larger m provides no further improvement. We therefore adopt $(E = 12, m = 2)$ as the default configuration.

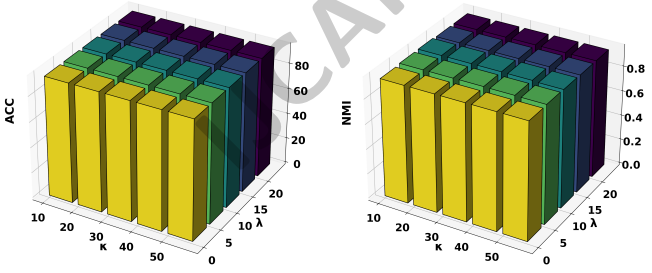
Sparse Setting (ds1)	Dense Setting (ds5)
Ground truth: the animation is superb and quality of the voice acting is also great.	**Ground truth:** the scenery in this movie is breathtaking and it alone is worth the price of admission.
Ours: the animation is visually impressive, and the voice acting adds emotional depth to the characters. The story feels cohesive and engaging, making it one of the most memorable animated films I've seen recently.	**Ours:** the scenery is breathtaking from start to finish, with sweeping landscapes and rich colors that captivate the viewer. The cinematography masterfully captures the emotional tone of each scene, making the visual experience alone worth the watch.
XRec: the animation looks vibrant, and families will enjoy the catchy tunes.	**XRec:** gorgeous locations with postcard-style vistas; perfect for a family night.
NETE: the animation is fine for this kind of film.	**NETE:** the scenery looks nice in parts.
PEPLER: a very good movie, really enjoyable.	**PEPLER:** looks great and is worth seeing.

Table 5: Case study comparison under sparse (ds1) and dense (ds5) interaction settings on the Amazon dataset.

Hyperparameters λ and κ. We analyze the sensitivity of HyperXRec to the regularization weight λ and the vMF concentration parameter κ on the Amazon dataset. We evaluate clustering quality using accuracy (ACC) and normalized mutual information (NMI), which measure the agreement between inferred clusters and ground-truth labels [Luo *et al.*, 2025]. As shown in Figures 5 (c) and 5 (d), clustering performance remains stable across a broad range of values. Moderate κ encourages compact and well-separated hyperspherical clusters, whereas excessively large values over-constrain the latent space and degrade clustering quality. Similarly, intermediate λ achieves the best trade-off between reconstruction fidelity and regularization. We therefore set $\lambda = 10$ and $\kappa = 30$ in all experiments.



(a) Varying E, fixed m=2 (b) Varying m, fixed E=12



(c) ACC over (κ, λ) (d) NMI over (κ, λ)

Figure 5: Hyperparameter study of HyperXRec.

4.5 Robustness Under Data Sparsity

We evaluate the robustness of HyperXRec under varying levels of data sparsity on the Amazon dataset by grouping users according to their interaction frequency (ds1-ds5), as shown in Figure 6. As user activity increases, explanation quality improves for all methods; notably, HyperXRec consistently outperforms all baselines, even in the sparsest setting (ds1). In sparse regimes, baseline methods degrade faster due to unstable representation learning and cluster formation, which lead to inconsistent MoE routing. By contrast, HyperXRec leverages structured hyperspherical preference representations to preserve stable clustering even under limited interactions, and couples this structure with cluster-guided expert routing. Representative examples under sparse and dense settings are shown in Table 5, with additional results on other datasets provided in the supplementary material.

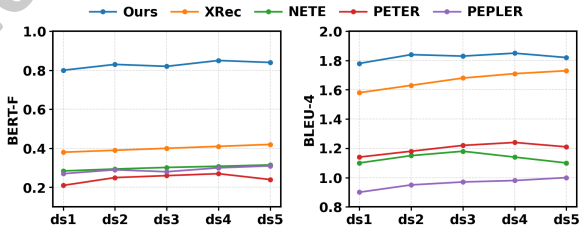


Figure 6: Explanation quality under varying interaction sparsity.

5 Conclusion

We proposed HyperXRec, a unified framework for explainable recommendation that tightly couples hyperspherical latent preference modeling with cluster-guided expert routing in an MoE-enhanced language model. By learning semantically structured user-item representations via a vMFMM prior and using the inferred cluster posteriors to steer expert activation, HyperXRec produces explanations that are more faithful to the underlying collaborative evidence, while remaining coherent and personalized. Extensive experiments on multiple real-world datasets show that HyperXRec outperforms strong baselines, particularly under sparse interactions, while maintaining strong semantic alignment and diversity. In future work, we plan to extend HyperXRec by exploring finer-grained and hierarchical preference structures to capture multi-facet user intents beyond a single clustering layer.

Acknowledgments

The completion of this work was supported by the National Natural Science Foundation of China (62276106), the Guangdong Basic and Applied Basic Research Foundation (2024A1515011767), and the Guangdong Provincial Key Laboratory IRADS (2022B1212010006).

Contribution Statement

Xue Han and Zhiwen Luo contributed equally: Conceptualization, Methodology, Investigation, Writing–original draft; **Zhixiang Li, Nizar Bouguila, and Weifeng Su**: Validation, Writing–review & editing; **Wentao Fan**: Conceptualization, Supervision, Funding acquisition.

References

[Brita *et al.*, 2025] Catalin E. Brita, Hieu Nguyen, Lubov Chalakova, and Nikola Petrov. Revisiting XRec: How collaborative signals influence LLM-based recommendation explanations. *Transactions on Machine Learning Research*, 2025.

[Cai and Cai, 2022] Zefeng Cai and Zerui Cai. Pevae: A hierarchical vae for personalized explainable recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*, pages 692–702, 2022.

[Dai *et al.*, 2024] Damai Dai, Chengqi Deng, Chenggang Zhao, R.x. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y.k. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*, pages 1280–1297, 2024.

[Davidson *et al.*, 2018] Tim R. Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M. Tomczak. Hyperspherical variational auto-encoders. In *Proceedings of the 34th Conference on Uncertainty in Artificial Intelligence (UAI 2018)*, pages 856–865, 2018.

[El-Kishky *et al.*, 2023] Ahmed El-Kishky, Thomas Markovich, Kenny Leung, Frank Portman, Aria Haghighi, and Ying Xiao. knn-embed: Locally smoothed embedding mixtures for multi-interest candidate retrieval. In *Advances in Knowledge Discovery and Data Mining: Proceedings of the 27th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2023)*, pages 374–386, 2023.

[Fan *et al.*, 2026] Wentao Fan, Wenchuan Zhang, Xiao Dong, and Nizar Bouguila. Clustering-based brain functional segmentation via deep collapsed nonparametric von mises-fisher mixture models. *Expert Systems with Applications*, 298:129739, 2026.

[Fedus *et al.*, 2022] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

[Guo *et al.*, 2026a] Dayu Guo, Zhiwen Luo, Nizar Bouguila, and Wentao Fan. Multimodal topic discovery in web media via von mises-fisher mixture neural topic models. In *Proceedings of the ACM Web Conference 2026*, page 8749–8752, 2026.

[Guo *et al.*, 2026b] Dayu Guo, Zhiwen Luo, Nizar Bouguila, and Wentao Fan. Neural topic modeling on hyperspheres: Spherical representation learning with von mises-fisher mixtures. *Neural Networks*, 200:108792, 2026.

[Han *et al.*, 2025] Xue Han, Zhixiang Li, Wenchuan Zhang, Hanyuan Huang, and Wentao Fan. Robust generalized zero-shot learning via dual-stream variational autoencoders and out-of-distribution detection. In *IEEE International Conference on Multimedia and Expo, ICME 2025, Nantes, France, June 30 - July 4, 2025*, pages 1–6, 2025.

[Hu *et al.*, 2022] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *Proceedings of the 10th International Conference on Learning Representations (ICLR 2022)*, 2022.

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

[Lepikhin *et al.*, 2021] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. In *Proceedings of the 9th International Conference on Learning Representations (ICLR 2021)*, 2021.

[Li *et al.*, 2020] Lei Li, Yongfeng Zhang, and Li Chen. Generate neural template explanations for recommendation. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20)*, pages 755–764, 2020.

[Li *et al.*, 2021] Lei Li, Yongfeng Zhang, and Li Chen. Personalized transformer for explainable recommendation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL 2021)*, pages 4947–4957, 2021.

[Li *et al.*, 2023] Lei Li, Yongfeng Zhang, and Li Chen. Personalized prompt learning for explainable recommendation. *ACM Transactions on Information Systems*, 41(4):103:1–103:26, 2023.

[Li *et al.*, 2025a] Haodong Li, Lianyong Qi, Weiming Liu, Xiaolong Xu, Wanchun Dou, Yang Cao, Xuyun Zhang, Amin Beheshti, and Xiaokang Zhou. Balancing user–item structure and interaction with large language models and optimal transport for multimedia recommendation. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2025)*, pages 3027–3035, 2025.

[Li *et al.*, 2025b] Qingfeng Li, Wei Liu, Zaiqiao Meng, and Jian Yin. Decision-aware preference modeling for multi-

behavior recommendation. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2025)*, pages 3063–3071, 2025.

[Li *et al.*, 2025c] Xiaodong Li, Jiawei Sheng, Jiangxia Cao, Xinghua Zhang, Wenyuan Zhang, Yong Sun, Shirui Pan, Zhihong Tian, and Tingwen Liu. Personalized multi-interest modeling for cross-domain recommendation to cold-start users. In *Proceedings of the 41st IEEE International Conference on Data Engineering (ICDE 2025)*, pages 4092–4105, 2025.

[Li *et al.*, 2025d] Yuhan Li, Xinni Zhang, Linhao Luo, Heng Chang, Yuxiang Ren, Irwin King, and Jia Li. G-refer: Graph retrieval-augmented large language model for explainable recommendation. In *Proceedings of the ACM Web Conference 2025 (WWW '25)*, pages 240–251, 2025.

[Li *et al.*, 2025e] Zhixiang Li, Zhiwen Luo, Nizar Bouguila, Weifeng Su, and Wentao Fan. Disentangled representation learning for multi-view clustering via von mises-fisher hyperspherical embedding. *Neural Networks*, page 107802, 2025.

[Luo *et al.*, 2025] Zhiwen Luo, Wentao Fan, Manar Amayri, and Nizar Bouguila. Dynamic deep clustering of high-dimensional directional data via hyperspherical embeddings with bayesian nonparametric mixtures. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2025)*, pages 938–949, 2025.

[Luo *et al.*, 2026] Zhiwen Luo, Wentao Fan, Manar Amayri, and Nizar Bouguila. Adaptive deep clustering via disentangled hyperspherical vaes with contrastive geometry optimization. *Neurocomputing*, page 132625, 2026.

[Ma *et al.*, 2024] Qiyao Ma, Xubin Ren, and Chao Huang. Xrec: Large language models for explainable recommendation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 391–402, 2024.

[McAuley and Leskovec, 2013] Julian J. McAuley and Jure Leskovec. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the Seventh ACM Conference on Recommender Systems (RecSys '13)*, pages 165–172, 2013.

[Pazzani and Billsus, 2007] Michael J. Pazzani and Daniel Billsus. Content-based recommendation systems. In *The Adaptive Web: Methods and Strategies of Web Personalization*, pages 325–341, 2007.

[Qiu *et al.*, 2025] Yilun Qiu, Tianhao Shi, Xiaoyan Zhao, Fengbin Zhu, Yang Zhang, and Fuli Feng. Latent inter-user difference modeling for llm personalization. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP 2025)*, pages 10599–10617, 2025.

[Rajbhandari *et al.*, 2020] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC 2020)*, pages 1–16, 2020.

[Sellam *et al.*, 2020] Thibault Sellam, Dipanjan Das, and Ankur Parikh. Bleurt: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, pages 7881–7892, 2020.

[Sinha and Dieng, 2021] Samarth Sinha and Adji B. Dieng. Consistency regularization for variational auto-encoders. In *Proceedings of the 35th International Conference on Neural Information Processing Systems (NeurIPS 2021)*, 2021.

[So *et al.*, 2021] David So, Wojciech Mańke, Hanxiao Liu, Zihang Dai, Noam Shazeer, and Quoc V. Le. Searching for efficient transformers for language modeling. In *Proceedings of the 35th International Conference on Neural Information Processing Systems (NeurIPS 2021)*, pages 6010–6022, 2021.

[Tang *et al.*, 2024] F. Tang, Y. Shen, H. Zhang, Z. Tan, W. Zhang, G. Hou, K. Song, W. Lu, and Y. Zhuang. Gavamoe: Gaussian-variational gated mixture of experts for explainable recommendation. *arXiv preprint arXiv:2410.11841*, 2024.

[Tintarev and Masthoff, 2007] Nava Tintarev and Judith Masthoff. A survey of explanations in recommender systems. In *Proceedings of the 2007 IEEE International Conference on Data Engineering Workshops (ICDEW 2007)*, pages 801–810, 2007.

[Wang *et al.*, 2023] Jiaan Wang, Yunlong Liang, Fandong Meng, Zengkui Sun, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. Is chatgpt a good nlg evaluator? a preliminary study. In *Proceedings of the 4th New Frontiers in Summarization Workshop*, pages 1–11, 2023.

[Yang *et al.*, 2021] Lin Yang, Wentao Fan, and Nizar Bouguila. Deep clustering analysis via dual variational autoencoder with spherical latent embeddings. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–10, 2021.

[Zhang and Chen, 2020] Yongfeng Zhang and Xu Chen. Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval*, 14(1):1–101, 2020.

[Zhang *et al.*, 2020] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *Proceedings of the 8th International Conference on Learning Representations (ICLR 2020)*, 2020.

[Zhou *et al.*, 2022] Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Y. Zhao, Andrew M. Dai, Zhifeng Chen, Quoc V. Le, and James Laudon. Mixture-of-experts with expert choice routing. In *Proceedings of the 36th International Conference on Neural Information Processing Systems (NeurIPS 2022)*, 2022.

[Zhu *et al.*, 2024] Yaochen Zhu, Liang Wu, Qi Guo, Liangjie Hong, and Jundong Li. Collaborative large language model for recommender systems. In *Proceedings of the ACM Web Conference 2024 (WWW '24)*, pages 3162–3172, 2024.