Why Now? Causally Grounded Explanations for Dynamic Recommendation

Anonymous ACL submission

Abstract

Explainable recommendation has gained increasing attention for its ability to build user trust through transparent and meaningful justifications. However, real-world user preferences and item attributes are inherently dynamic, yet most existing methods rely on static historical interactions, often producing outdated recommendations and implausible explanations. In this paper, we propose DyCEX (Dynamic Causal Explanation), a novel framework for generating causally grounded, temporally aware, and cognitively plausible explanations in dynamic recommendation scenarios. Specifically, we first design a causality-guided representation learner that models the temporal evolution of users and items through inferred cause-effect relationships, effectively filtering out obsolete signals to better reflect present-day interests. Second, we employ a dual-path gated fusion strategy that distinguishes stable thematic affinities from transient stylistic trends by adaptively reweighting features across time, yielding more coherent user and item representations. Third, we leverage a Large Language Model (LLM) guided by Chain-of-Thought (CoT) prompting to generate step-by-step natural language explanations that logically connect current user needs with relevant item attributes. Extensive experiments on three real-world datasets demonstrate that DyCEX significantly outperforms state-of-the-art baselines in explanation quality.

1 Introduction

Explainable recommendation hinges on a temporally aware modeling of user and item profiles, since both user interests and item characteristics undergo continuous evolution. Existing approaches (Lin et al., 2024; Zhang et al., 2024a) that depend solely on historical interactions often incorporate obsolete patterns, leading to diminished recommendation accuracy and less convincing explanations. To maintain effectiveness, recommender systems must therefore dynamically update their representations to generate timely and credible justifications from a natural language perspective.

We study the problem of temporally coherent and cognitively aligned explainable recommendation, where both user preferences and item attributes continuously evolve. Given a target user–item pair (u, i) with its rating score $r_{u,i}$, our goal is to generate a coherent natural language explanation $EX_{u,i}$ that justifies why this item is relevant to the user right now, not just based on what they liked in the past. For explainable recommendation in such a dynamic setting, three critical challenges must be addressed (as shown in Figure 1).

C1: Recommendations must reflect the user's present self, not their past shadow. People's tastes evolve as their lives change—whether through new responsibilities, shifting values, or personal growth. Take Lena, for instance: in her early twenties, she devoured fast-paced action thrillers and gritty crime dramas, drawn to their intensity and moral ambiguity. Now in her thirties, as a parent balancing work and family, she seeks uplifting stories with emotional depth and hopeful messages. If a system recommends another violent noir film with the explanation "You rated Heat 5 stars five years ago", the suggestion feels out of touch. That's not because her past preference was wrong, but because it no longer represents who she is. An effective explainable recommender must recognize this transformation and anchor its reasoning in Lena's current life stage, not her archival viewing history.

C2: Not all features age equally, i.e., some fade, others endure, and the system must tell them apart. In Lena's viewing history, early preferences like "high-speed chases" or "morally gray heroes" probably reflected a temporary desire for excitement and rebellion. By contrast, certain qualities, such as "well-developed characters", "dialogue with depth", and "stories about second

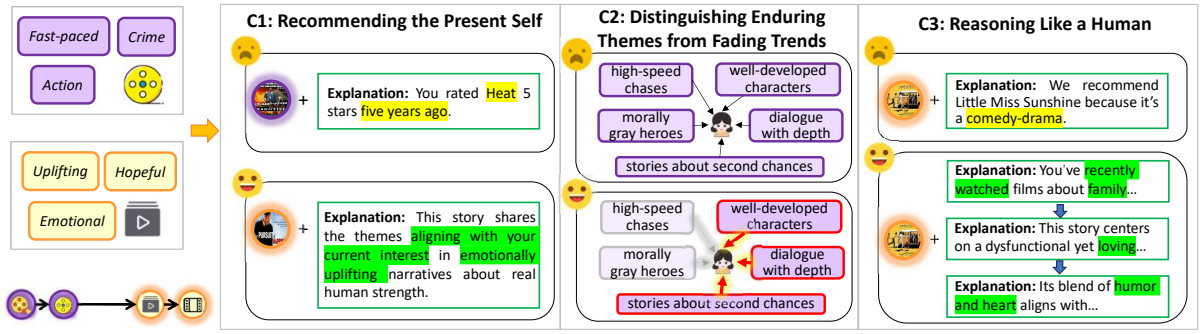

Figure 1: Motivation Examples.

chances", have remained consistent, appearing in both her past crime thrillers and her current indie films. Likewise, a movie once praised for its "cutting-edge visual effects" might today be appreciated more for its "emotional story" or "genuine acting". If the system weights all historical signals uniformly, it risks overemphasizing fleeting trends while overlooking enduring affinities. To stay relevant, it must distinguish between ephemeral stylistic preferences and stable thematic inclinations, and prioritize the latter in explanation.

C3: Explanations must reason like a thoughtful human, not recite a label. Simply telling Lena "We recommend Little Miss Sunshine because it's a comedy-drama" provides little justification. But a layered explanation "You've recently watched films about family resilience; this story centers on a dysfunctional yet loving family overcoming adversity together; its blend of humor and heart aligns with your preference for emotionally authentic narratives" creates a compelling causal chain. Users like Lena don't just want a title. They want to see how the recommendation connects to their evolving identity and current needs. This requires explanations that progress step by step, linking observed behavior to item attributes through plausible reasoning, just as a knowledgeable friend would.

To address the aforementioned challenges, we propose DyCEX (Dynamic Causal Explanation), a novel framework for explainable recommendation that effectively bridges temporal dynamics and cognitive reasoning. Specifically, DyCEX addresses the misalignment between static historical signals and users' evolving identities. It learns causality-informed representations that prioritize recent interactions. At the same time, it actively suppresses outdated preferences. This ensures that recommendations reflect who the user is now, not who they were in the past. It further distinguishes transient features from stable core traits through an adaptive feature weighting mechanism, thereby preserving meaningful affinities while discarding ephemeral noise. Finally, to generate truly persuasive justifications, DyCEX leverages a Large Language Model (LLM) guided by a Chain-of-Thought (CoT) prompting strategy. It constructs explanations as a step-by-step causal chain: first inferring the user's current need, then identifying relevant item attributes, and finally linking them through human-like reasoning. We summarise our contributions as follows.

- To align recommendations with users' current selves, we design a causality-guided representation learner that captures the temporal dynamics of users/items through an inferred causal structure. By focusing on cause–effect relationships instead of raw correlations, it filters out outdated signals from past interactions, producing representations that better reflect present-day interests.

- To distinguish enduring traits from transient trends, we propose a dual-path gated fusion strategy that leverages cross-attention to compare user/item features across time windows. The gate dynamically amplifies stable characteristics while attenuating outdated ones, resulting in more accurate and temporally coherent profiles.

- To produce human-like, logically sound explanations, we develop a cognition-inspired generation approach based on CoT prompting. This method decomposes the reasoning process into sequential steps, leading to natural language explanations that are both structured and persuasively grounded.

- We conduct extensive experimental evaluations on three real datasets from online platforms. The experiment results demonstrate the high effectiveness of our proposed model.

2 Related Work

We review existing literature on two topics closely related to our study, including explainable recommendation systems and representation learning for explainable recommendation.

2.1 Explainable Recommendation Systems

Explainable recommendation aims to reveal the principles behind recommendation results and generate human-comprehensible textual explanations. Current research in explainable recommendation primarily focuses on three directions: adopting advanced language models (Li et al., 2017, 2021, 2023; Ma et al., 2024), enhancing semantic representations (Zhao et al., 2024; Liu et al., 2025; Shimizu et al., 2025), and incorporating auxiliary information (Ma et al., 2024; Li et al., 2025). For example, advanced language models like GPT-2 (Li et al., 2023) and LLaMA 2 (Ma et al., 2024) have been applied to produce higher-quality explanations. The MMI model (Zhao et al., 2024) employs reinforcement learning to align the generated explanations with user ratings. Frameworks like XRec (Ma et al., 2024) enrich the process by constructing attribute-based profiles, providing deeper contextual grounding for explanation generation. However, current methods fail to capture the dynamic nature of user preferences and item attributes, and overlook deeper cognitive mechanisms. Consequently, the generated explanations often lack persuasiveness and user acceptance.

2.2 Representation Learning for Explainable Recommendation

Representation learning is central to explainable recommendation, encoding user preferences, item attributes, and interactions into vector spaces. Existing methods fall into three categories based on information sources: **ID-based** representation learning uses only user and item IDs, aiming to imbue discrete identifiers with explanation-relevant semantics. NRT (Li et al., 2017) jointly optimizes rating and tip generation via MLP-encoded IDs. PETER (Li et al., 2021) bridges IDs and words through a context prediction task in a personalized Transformer. PEPLER (Li et al., 2023) treats IDs as continuous prompts, optimized via sequential tuning with recommendation loss as regularization. NETE (Li et al., 2020) combines GRU-based ID encoding with neural templates to improve explanation controllability. **Content-enhanced** methods leverage textual or attribute features to enrich explainability. Att2Seq (Dong et al., 2017) encodes user-item-attribute triples with MLP and generates review-style explanations via LSTM. PEPLER-D (Li et al., 2023) replaces IDs with feature-derived discrete prompts, while GaVaMoE (Tang et al., 2024) integrates features into prompts, clusters users via GMM, and uses expert networks for personalized explanations. **Graph-augmented** approaches model interactions as graphs and use GNNs to capture high-order relations. XRec (Ma et al., 2024) aligns LightGCN-encoded graph representations with LLMs via a collaborative adapter. G-Refer (Li et al., 2025) retrieves structural and semantic CF paths from graphs and translates them into natural language to augment LLM inputs. Specifically, building on the paradigm of XRec, we enhance the representations learned by the Mixture of Experts (MoE) based on causal graph principles, thereby eliminating obsolete information embedded in these representations.

3 Framework of Our Solution

In this work, we propose a novel model named DyCEX, as illustrated in Fig. 2. Our model has three core components: causality-guided representation learner, dual-path gated fusion, and cognition-based explanation generation. First, the former captures evolving user preferences and item attributes by modeling their causal structures in dynamic interactions. It proceeds in three steps: **a**) obtaining initial user/item representations (rich in collaborative information) and their multi-stage feature semantic representations; **b**) fusing the previous-stage representations with current feature semantics via Dual-path Gated Fusion to generate multi-stage real-time representations; **c**) producing the final representation by attention-weighting these multi-stage representations. Second, the cognition-guided explanation generation component employs large language models to produce structured, logically sound and persuasive explanations. The training and inference workflows of the model are presented in the Appendix. We will detail these components in Section 4.

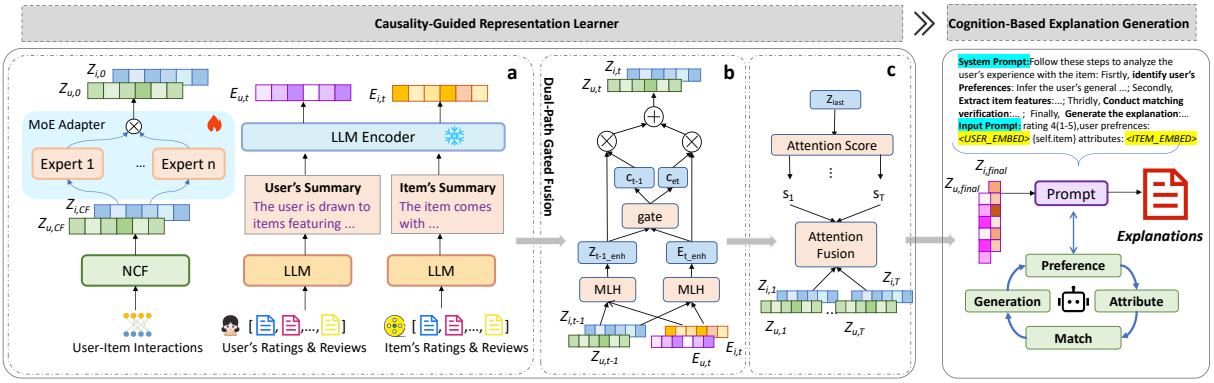

Figure 2: Overview of DyCEX.

4 Our Proposed Model: DyCEX

In this section, we will present our proposed model (DyCEX) that generates a natural language explanation grounded in causal, step-by-step reasoning.

4.1 Causality-Guided Representation Learner

Guided by causal theory (Pearl, 2009; Dahabreh and Bibbins-Domingo, 2024; Luo et al., 2024), we model user–item interactions via the causal graph in Fig. 3. We assume short-term preference stability and split interaction histories into consecutive time segments, each representing a distinct environment. In each environment t, user preferences $Z_{u,t}$ and item attributes $Z_{i,t}$ are shaped by their current features $E_{u,t}$ and $E_{i,t}$, respectively. Preference-attribute alignment between $Z_{u,t}$ and $Z_{i,t}$ determines the rating $R_{a,t}$ and review $R_{e,t}$. While representations maintain stability within a single environment, transitions between periods (e.g., from t-1 to t) may induce shifts in $Z_{u,t}$ and $Z_{i,t}$, driven by changes in $E_{u,t}$ vs. $E_{u,t-1}$ and $E_{i,t}$ vs. $E_{i,t-1}$. For instance, a user shifts from "rock" to "light music". Capturing these time-varying causal signals accurately is essential for dynamic representation learning.

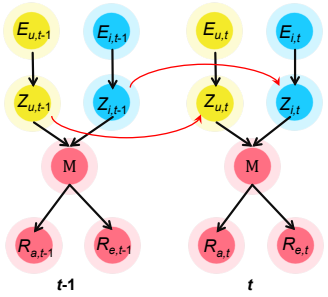

Figure 3: Time-Varying Causal Signals Modeling.

We initialize user and item representations using low-dimensional embeddings ($Z_{u,CF}$ and $Z_{i,CF}$) from a pre-trained neural collaborative filtering (NCF) model, which encode rich collaborative signals and group-based preferences. To bridge the gap between behavior-aware collaborative representations and language-based semantic representations, we adopt a Mixture of Experts (MoE) fusion mechanism to adaptively integrate structured collaborative tokens and unstructured language tokens through a gating network ($Z_{u/i,init}$ = MoE($Z_{u/i,CF}$)). The gating network computes expert weights using W_{gate} and injects noise ϵ during training for robustness:

$$g = \text{Softmax}(Z \cdot W_{\text{gate}} + \epsilon) \quad (1)$$

$$\text{MoE}(Z) = \sum_{i=1}^{K} g_i(Z) \cdot \text{E}_i(Z) \quad (2)$$

where g_i denotes the weight assigned to the i-th expert , $E_i(Z)$ is the output of the i_th expert network for input Z and K denotes the number of experts.

To model temporal preference shifts, we propose a causality-guided sequence processing method. Notably, this approach follows the field-recognized causal graph-guided representation learning paradigm, which focuses on learning representations based on predefined causal structures and modeling the generation mechanism of user-item interaction behaviors, as outlined in (Gao et al., 2024)'s survey and implemented in (Wang et al., 2023b). For each time step t, text embeddings (from user or item descriptions) are pooled into global semantic features. These are fused with previous state embeddings (with $Z_{u/i,init}$ serving as $Z_{u/i,t-1}$ when t = 1) via dual-path gated fusion (DPGF) strategy, leveraging causal dependencies ($(E_{u,t}, Z_{u,t-1}) \rightarrow Z_{u,t}$ and $(E_{i,t}, Z_{i,t-1}) \rightarrow Z_{i,t}$). For notational brevity, we denote $Z_{i,t}$ and $Z_{u,t}$

as Z_t, and similarly denote $E_{i,t}$ and $E_{u,t}$ as E_t throughout the paper, given that the vectors in each pair are subjected to identical downstream operations:

$$Z_t = DPGF(E_t, Z_{t-1}) \tag{3}$$

The fused representation is transformed through an MLP to update the current state embedding Z. Finally, to capture holistic evolution patterns, we aggregate embeddings across all time steps via a learnable temporal fusion strategy:

$$Z_{\text{final}} = \sum_{t=1}^{T} \text{Softmax}(f_\theta(Z_{\text{last}})_t) \cdot Z_t \tag{4}$$

where $f_\theta(Z_{\text{last}})_t$ computes the attention weight for each step t. This approach reduces bias from isolated time points and enhances the robustness of sequence modeling.

4.2 Dual-Path Gated Fusion

We have devised a dual-path gated fusion strategy. The aim is to efficiently integrate the previous-stage representations of users and items with their current-stage feature representations, thus yielding representations that accurately reflect the current-stage state $Z_t = DPGF(E_t, Z_{t-1})$. Guided by the causal graph theory, this strategy leverages a cross-attention mechanism to precisely identify stable and outdated features in representations across different time windows. Simultaneously, it employs a gating mechanism to purposefully strengthen stable features and effectively weaken outdated ones, ultimately generating user and item representations that better align with the actual current-stage scenario.

DPGF utilizes a dual-path gating mechanism that separately assesses the feature reliability (stable or outdated) of the previous-stage semantic representation and the current-stage feature representation.

It adopts a multi-head (MLH) cross-attention mechanism to precisely capture feature correlations between the previous-stage representation and the current-stage feature representation. This mechanism designates one representation as the query and the other as the key and the value, excavating their correlations to generate an enhanced representation where stable features are strengthened and outdated features are weakened. Moreover, the multi-head mechanism enables cross-attention to capture feature correlations from multiple dimensions, providing a reliable basis for the subsequent gating-based "strengthening/weakening" operations.

$$E_{t\_enh} = MLH(Z_{t-1}, E_t, E_t) \tag{5}$$

$$Z_{t-1\_enh} = MLH(E_t, Z_{t-1}, Z_{t-1}) \tag{6}$$

Based on the confidence scores output by the gating mechanism, the two enhanced representations are weighted and fused. Stable features are prioritized for strengthening due to their high confidence scores, while outdated features are effectively weakened given their low scores.

$$c_i = \sigma(W_2 \cdot ReLU(W_1 \cdot Z_{enh})) \tag{7}$$

Subsequently, residual fusion and layer normalization are applied to ensure the stability of the fusion process and the effectiveness of feature representation. Finally, a complete user/item representation that better matches the actual current-stage situation is output.

$$Z_{t\_fused} = c_{et} \cdot E_{t\_enh} + c_{t-1} \cdot Z_{t-1\_enh} \tag{8}$$

$$Z_t = LayerNorm(W_3 \cdot Z_{t\_fused} + Z_{t\_fused}) \tag{9}$$

4.3 Cognition-Based Explanation Generation

Existing prompt (Yang et al., 2024; Liu et al., 2024) designs in explainable recommendation often rely on superficial textual features or isolated correlations, limiting their ability to leverage causal-aware representations or systematically model preference-attribute relationships. This often results in implausible or unpersuasive explanations. In contrast, CoT prompting enhances complex reasoning by breaking down problems into intermediate steps and connecting underlying features (Stechly et al., 2024; Zhang et al., 2024b). Aligned with our objective, CoT's multi-step reasoning capability enables effective use of causal evolutionary representations, transforms abstract preference-attribute matches into interpretable rationales, and ultimately improves explanation quality.

We design a four-stage CoT prompt framework, incorporating principles from cognitive psychology (Al-Aly and Rosen, 2024; Newell et al., 1972) and tailored to the "match visualization" requirement in explanations. Based on findings that using explicit (rather than predicted) ratings improves emotional accuracy, we integrate the actual rating $r_{u,i}$ explicitly into the prompt. The explanation generation process is defined as: $EX_{u,i} = f\Big([p_1, \ldots, p_u, p_i, r_{u,i}, \ldots, p_n, \ldots, p_i, \ldots, p_e, \ldots]\Big)$,

where p_u, p_i, and p_e are special tokens marking user and item representations and explanation position, respectively.

To enhance the ability of LLMs to generate contextually and syntactically coherent explanations, we aim to minimize the loss between the predicted probabilities of the next tokens and the actual next tokens in the sequences. We employ negative log-likelihood (NLL) (Yao et al., 2019) as our training loss:

$$\mathcal{L} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C_n} y_{nc} \cdot \log(\hat{y}_{nc}) \qquad (10)$$

where N denotes the number of explanations, C_n represents the number of tokens in each explanation, and y_{nc} and $\hat{y}_{nc}$ correspond to the actual and predicted tokens, respectively.

5 Experiments

5.1 Experimental Settings

Datasets. We conduct experiments on two public datasets: Movies&TV, Yelp and Books. Movies contains 7,636 items, 6,790 users, and 335,854 interactions. Yelp includes 8,280 items, 9,900 users, and 659,759 interactions. Books includes 16,077 items, 13,147 users, and 551,297 interactions. All three datasets include user ratings, textual reviews and review time. We partition data into environments by jointly considering absolute time and interaction density. Since all three datasets show steadily increasing interaction volume over time, we adaptively set longer time spans for early sparse years and shorter ones for recent dense years, ensuring each environment has roughly balanced interactions and enabling stable learning under temporal shifts.

Metrics. We utilize a comprehensive set of metrics tailored to assess both the semantic explainability and stability, such as GPTScore (Wang et al., 2023a), BERTScore (Zhang et al., 2019), BARTScore (Yuan et al., 2021) and USR (Li et al., 2021).

Baselines. We compare our model with five state-of-the-art baselines: **NRT** (Li et al., 2017), **PETER** (Li et al., 2021), **PEPLER** (Li et al., 2023), **XRec** (Ma et al., 2024), and **G-Refer** (Li et al., 2025). These models are built upon representative language architectures such as GRU, GPT, and LLaMA.

Implementation Details. Following the design paradigm of XRec (Ma et al., 2024), we adopt a pre-trained NCF model to learn initial representations that encapsulate rich collaborative information for both users and items, with the embedding dimension set to 128. Our MoE configuration employs 7 experts with a dropout rate of 0.2 and a noise factor of 0.01 in the gating router. We adopt LLaMA2-7B as the backbone model for DyCEX. We adopt the DeepSeek-V3.1 model, which serves dual purposes: acting as a generator to process reviews and generate explanation ground truth, and functioning as a discriminator to calculate GPTScore. Both ground truth and generated explanations are limited to a maximum of 50 words. We set the learning rate as 5e-5 for DyCEX, which is trained on NVIDIA A100. All source code has been provided on https://anonymous.4open.science/r/DyCEX.

5.2 Performance Comparison.

Evaluation results are shown in Table 1, where superscripts P, R, and F1 denote Precision, Recall, and F1-Score, respectively, and the subscript std indicates the standard deviation. The best and second-best results are highlighted in bold and underlined, respectively. Our model consistently outperforms all baselines in both explainability and stability. This improvement can be attributed to two main factors: 1) its ability to capture evolving user preferences and item attributes over time, and to effectively preserve stable features while suppressing outdated ones across different time spans, producing embeddings that reflect up-to-date and temporally adaptive characteristics; 2) its effective use of these embeddings to generate explanations that accurately reflect the preference-attribute matching process. Notably, our model achieves slightly lower $\text{BERTScore}^{\text{R}}$ than the most powerful baseline, G-Refer, on all datasets. This is because G-Refer's hybrid retrieval strategy accesses richer external information compared to our embeddings, which primarily encode implicit collaborative signals. However, this added retrieval also introduces noise, resulting in a comparatively lower $\text{BERTScore}^{\text{P}}$ for G-Refer. Moreover, our model scores slightly lower on diversity metric (USR). This is primarily because it employs causality-guided representation learner and COT prompting to prioritize stable, essential characteristics of user preferences and item attributes. While this design enhances explanatory fidelity by filtering out redundant or noisy variations in the feature space, it comes at the cost of a modest reduction in output diversity.

Table 1: Overall Performance Comparison in Terms of Explainability and Stability.

Metrics	Explainability ↑						Stability ↓				
	GPTScore	BERTScore^P	BERTScore^R	BERTScore^{F1}	BARTScore	USR	GPT$_{\text{std}}$	BERT$^P_{\text{std}}$	BERT$^R_{\text{std}}$	BERT$^{F1}_{\text{std}}$	BART$_{\text{std}}$
Movies&TV											
NRT	61.08	0.2300	0.2971	0.2596	-4.5397	0.3281	21.76	0.2632	0.1975	0.2134	0.5843
PETER	66.99	0.3022	0.3579	0.3179	-4.2965	0.8622	20.00	0.4011	0.2031	0.2301	0.5955
PEPLER	67.73	0.3597	0.3743	0.3677	-4.3001	0.9121	18.43	0.1882	0.1869	0.1805	0.5721
XRec	73.10	0.4472	0.4075	0.4201	-4.1439	**1.0000**	15.93	**0.1488**	0.1873	0.1577	0.5732
G-Refer	73.31	0.4084	**0.4432**	0.4399	-3.8873	**1.0000**	**15.29**	0.1640	**0.1696**	0.1601	0.5572
Ours	**75.28**	**0.4543**	0.4284	**0.4520**	**-3.8618**	0.8793	16.38	0.1640	0.1851	**0.1521**	**0.5553**
Yelp											
NRT	71.49	0.4064	0.3454	0.3311	-3.5538	0.2537	18.2741	0.1979	0.1968	0.2247	0.6090
PETER	72.63	0.4211	0.3687	0.3447	-3.5476	0.8366	17.3387	0.2455	0.2046	0.2186	0.5888
PEPLER	73.04	0.4501	0.3981	0.3917	-3.5400	0.8781	16.3782	0.1771	0.1879	0.1701	0.5431
XRec	77.43	0.4600	0.4369	0.4494	-3.5392	**1.0000**	14.7332	**0.1521**	0.1883	0.1677	0.5333
G-Refer	79.22	0.4631	**0.4424**	0.4500	-3.4201	**1.0000**	**14.3143**	0.1632	0.1933	0.1602	0.5225
Ours	**80.79**	**0.4800**	0.4377	**0.4648**	**-3.4005**	0.8333	14.7117	0.1643	**0.1830**	**0.1586**	**0.5067**
Books											
NRT	76.03	0.4283	0.3676	0.3535	-3.4233	0.3835	17.1744	0.1762	0.1759	0.2034	0.5883
PETER	78.24	0.4433	0.3895	0.3705	-3.4171	0.8794	16.0384	0.2247	0.1838	0.1985	0.5677
PEPLER	78.03	0.4724	0.4199	0.4137	-3.4104	0.9187	15.0785	0.1562	0.1663	0.1493	0.5225
XRec	81.95	0.4843	0.4573	0.4701	-3.4191	**1.0000**	13.5330	**0.1323**	0.1681	0.1475	0.5131
G-Refer	83.74	0.4813	**0.4652**	0.4703	-3.3003	**1.0000**	**13.1142**	0.1431	0.1637	0.1401	0.5023
Ours	**84.27**	**0.4950**	0.4597	**0.4859**	**-3.2806**	0.8754	13.3116	0.1443	**0.1632**	**0.1385**	**0.4925**

5.3 Ablation Analysis.

We conduct ablation studies on the Movies&TV and Yelp datasets to evaluate the contribution of key components in our model: Causality-Guided Representation Learner, Dual-Path Gated Fusion, and Cognition-Based CoT. We compare four model variants: 1) **DyCEX w/o Causal**: uses only MoE-mapped representations; 2) **DyCEX w/o DPGF**: fuses z_{et} and z_{t-1} using an attention mechanism; 3) **DyCEX w/o CoT**: employs concise task description prompts without detailed reasoning; 4) **DyCEX w/o Causal & DPGF & CoT**: combines all the above limitations. To rigorously assess explainability and stability, we evaluate these variants on the Movies&TV and Yelp datasets using GPTScore and BERTScore (including their standard deviations), which helps clarify the critical role of each component in shaping the model's performance and capabilities.

As shown in Fig. 4, DyCEX outperforms other variants in both explainability and stability, highlighting its superior capability. Removing any single component leads to a significant performance drop, with the worst results observed when all three are removed. 1) DyCEX(w/o Causal) exhibits the largest drop in BERTScore, as this metric primarily evaluates the semantic alignment and key feature fidelity between generated explanations and reference labels. The Causality-Guided Representation Learner enables dynamic updates of user preferences and item attributes by capturing time-varying causal signals and filtering out outdated information—thereby grounding explanations in accurate, relevant features. Removing this component results in explanations that either miss critical signals or incorporate irrelevant content, substantially degrading BERTScore performance. 2) DyCEX(w/o DPGF) also shows a significant drop in BERTScore, similar to DyCEX(w/o Causal). This indicates that DPGF is more effective than standard attention mechanisms at identifying stable and outdated attributes in the representations, thereby enabling better fusion of z_{et} and z_{t-1}. Without DPGF, the resulting explanations are less accurate and relevant, leading to a drop in BERTScore. 3) The drop in GPTScore is most pronounced for DyCEX(w/o COT), primarily because GPTScore emphasizes the persuasiveness and cognitive plausibility of explanations—qualities that CoT enhances by generating structured, logically coherent rationales through step-by-step reasoning aligned with human cognition.

5.4 Effect of Multiple Environments.

As illustrated in Section 5.1, the choice of the environment number T is essential. To simplify the setup and avoid separately tuning the number of environments for training and inference on each dataset, we use the same number of environments in both periods. Fig.5 presents the results under varying numbers of environments, from which we draw the following findings.

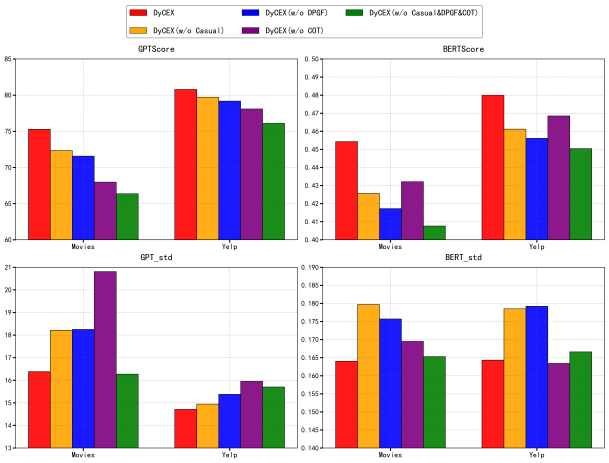

Figure 4: Ablation Study on Variant Models.

Table 2: Examples of the Generated Explanations.

Reference	The Hong Kong-style **milk tea** is rich and aromatic, and their new **truffle mac and cheese** is an absolute must-try delicacy. With its **lively atmosphere**and **great value**, this place has totally won me over.
NRT	Exceptionally **tasty mil**k.
PETER	This is a really **great Hong Kong cafe**, and pineapple buns are perfectly crispy.
PEPLER	This is an authentic Hong Kong-style cafe with **delicious milk tea** and **great value for money**.
XRec	Its **lively atmosphere**, paired with a priced milk tea and bread combo, makes it an **excellent choice**.
G-Refer	The **milk tea** is rich and aromatic. With its **lively atmosphere** and **great value**, this place has won me over.
DyCEX	Authentic Hong Kong **cafe** with **vibrant atmosphere**, **great value**, and delicious **truffle mac and cheese**.

First, The performance rises at first and then drops with an increase in the number of environments. This rise validates the effectiveness of the causality-guided representation learner that captures user preferences and item attributes across multiple environments, as opposed to conventional single-environment representation learning paradigms. Besides, the performance drop also verifies the arguments in Section 5.1: an increasing number of environments will lead to sparse interaction data within each individual environment. This, in turn, hinders the model's disentangled learning of time-invariant versus outdated user preferences and item attributes within each environment, thereby compromising the accuracy of the learned representations.

Second, The effectiveness of environment number increasing is more significant on Movies&TV and Yelp than on the Books. The fundamental reason is that Movies&TV and Yelp domains exhibit more pronounced temporal shifts, and using more environments allows the model to capture finer-grained variations across them. In contrast, user preferences on Books are comparatively stable, so DyCEX employs fewer environments to more effectively learn the invariant preference within each one.

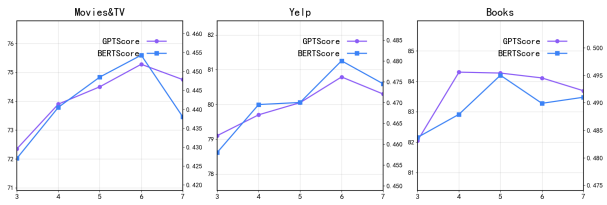

Figure 5: Effect of environment numbers.

5.5 Case Study.

To further assess DyCEX's ability to produce personalized and relevant explanations, we perform a case study comparing its outputs with baseline methods on the Yelp dataset. Key comparisons are summarized in Table 2. Our model successfully identifies salient features (such as the authentic Hong Kong-style café atmosphere, vibrant ambience, and affordability), while also incorporating newly introduced menu items like truffle macaroni. This demonstrates the efficacy of our key modules, which together capture up-to-date user/item representations and achieve accurate preference-attribute alignment. In contrast, while strong baselines such as XRec and G-Refer can produce reasonable explanations, they fail to capture the latest features.

6 Conclusion

In this paper, we propose a novel recommendation model that integrates causal representation learning with cognitive explanation generation. It consists of three core modules: a causality-guided representation learner that captures evolving user preferences/item attributes through causal modeling, a dual-path gated fusion strategy that precisely identifies stable and outdated features across different time windows, strengthens the former and weakens the latter to generate user and item representations better suited to the current stage, and a cognition-based explanation generation that guides LLMs to produce persuasive explanations.

Limitations

This study has carried out systematic work in fusing collaborative signals and text semantic information for recommendation explanation generation by constructing the DyCEX model. The proposed core framework and key technologies provide valuable references for research in related fields. Meanwhile, due to the exploratory nature and inherent limitations of the research, there are several directions that can be further improved, as follows:

First, there is room for improvement in the mining and utilization of collaborative text semantic information. The DyCEX model proposed in this study has achieved the core task of fusing collaborative information by obtaining implicit collaborative signals from low-dimensional embeddings encoded by a pre-trained NCF. However, it has not conducted in-depth mining and integration of structured explicit collaborative information, resulting in insufficient explicitness and richness in the expression of collaborative text semantic information. In contrast, G-Refer (Li et al., 2025) extracts explicit path/node-level collaborative signals through hybrid retrieval and converts them into human-readable text, providing ideas for the utilization of explicit collaborative information. Future research can further supplement this dimension of work to more fully reflect the association basis between users and items and enhance the depth and persuasiveness of explanations.

Second, the model's inference efficiency requires further optimization to adapt to a wider range of application scenarios. The DyCEX model integrates core modules such as the MoE fusion mechanism, causality-guided sequence processing, and multi-stage CoT prompting, which effectively guarantees the quality of recommendation explanations. However, the sequential execution of each module inevitably introduces a certain amount of computational overhead. This leads to relatively long inference latency when the model is applied to large-scale user-item interaction datasets, making it temporarily difficult to fully adapt to real-time recommendation scenarios that require strict low-latency responses. In subsequent work, the inference efficiency can be improved by optimizing the model architecture and simplifying redundant computational steps, thereby expanding the practical application scope of the model.

Finally, the diversity of generated explanations can be further enhanced by introducing multi-modal information. This study has fully utilized two types of core data, namely textual profiles and structured interaction data, to complete the explanation generation task. However, it has not covered the rich multi-modal information in real-world scenarios (e.g., item images, audio features, contextual scene descriptions). As a result, the currently generated explanations are text-centric, lacking multi-dimensional information support, and thus cannot fully meet users' diverse information consumption preferences. In the future, multi-modal data can be incorporated for fusion modeling to further improve the comprehensiveness and persuasiveness of explanations.

References

Ziyad Al-Aly and Clifford J Rosen. 2024. Long covid and impaired cognition—more evidence and more work to do.

Issa J Dahabreh and Kirsten Bibbins-Domingo. 2024. Causal inference about the effects of interventions from observational studies in medical journals. *Jama*, 331(21):1845–1853.

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

Chen Gao, Yu Zheng, Wenjie Wang, Fuli Feng, Xiangnan He, and Yong Li. 2024. Causal inference in recommender systems: A survey and future directions. *ACM Transactions on Information Systems*, 42(4):1–32.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182.

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*, pages 426–434.

Lei Li, Yongfeng Zhang, and Li Chen. 2020. Generate neural template explanations for recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 755–764.

Lei Li, Yongfeng Zhang, and Li Chen. 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)*.

Lei Li, Yongfeng Zhang, and Li Chen. 2023. Personalized prompt learning for explainable recommendation. *TOIS*, pages 1–26.

Piji Li, Zihao Wang, Zhaochun Ren, Lidong Bing, and Wai Lam. 2017. Neural rating regression with abstractive tips generation for recommendation. In *SIGIR*, pages 345–354.

Yuhan Li, Xinni Zhang, Linhao Luo, Heng Chang, Yuxiang Ren, Irwin King, and Jia Li. 2025. G-refer: Graph retrieval-augmented large language model for explainable recommendation. In *WWW*, pages 240–251.

Xinyu Lin, Wenjie Wang, Yongqi Li, Shuo Yang, Fuli Feng, Yinwei Wei, and Tat-Seng Chua. 2024. Data-efficient fine-tuning for llm-based recommendation. In *SIGIR*, pages 365–374.

Shijie Liu, Ruixin Ding, Weihai Lu, Jun Wang, Mo Yu, Xiaoming Shi, and Wei Zhang. 2025. Coherency improved explainable recommendation via large language model. In *AAAI*, pages 12201–12209.

Xu Liu, Tong Yu, Kaige Xie, Junda Wu, and Shuai Li. 2024. Interact with the explanations: Causal debiased explainable recommendation system. In *WSDM*, pages 472–481.

Huishi Luo, Fuzhen Zhuang, Ruobing Xie, Hengshu Zhu, Deqing Wang, Zhulin An, and Yongjun Xu. 2024. A survey on causal inference for recommendation. *The Innovation*, 5(2).

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

Andriy Mnih and Russ R Salakhutdinov. 2007. Probabilistic matrix factorization. *Advances in neural information processing systems*, 20.

Allen Newell, Herbert Alexander Simon, and 1 others. 1972. *Human problem solving*, volume 104. Prentice-hall Englewood Cliffs, NJ.

Judea Pearl. 2009. *Causality*. Cambridge University Press.

Ryotaro Shimizu, Takashi Wada, Yu Wang, Johannes Kruse, Sean O'Brien, Sai HtaungKham, Linxin Song, Yuya Yoshikawa, Yuki Saito, Fugee Tsung, and 1 others. 2025. Disentangling likes and dislikes in personalized generative explainable recommendation. In *WWW*, pages 4793–4809.

Kaya Stechly, Karthik Valmeekam, and Subbarao Kambhampati. 2024. Chain of thoughtlessness? an analysis of cot in planning. *NeurIPS*, 37:29106–29141.

Fei Tang, Yongliang Shen, Hang Zhang, Zeqi Tan, Wenqi Zhang, Zhibiao Huang, Kaitao Song, Weiming Lu, and Yueting Zhuang. 2024. Gavamoe: Gaussian-variational gated mixture of experts for explainable recommendation. *arXiv preprint arXiv:2410.11841*.

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

Wenjie Wang, Xinyu Lin, Liuhui Wang, Fuli Feng, Yunshan Ma, and Tat-Seng Chua. 2023b. Causal disentangled recommendation against user preference shifts. *ACM Transactions on Information Systems*, 42(1):1–27.

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

Hengshuai Yao, Dong-lai Zhu, Bei Jiang, and Peng Yu. 2019. Negative log likelihood ratio loss for deep neural network classification. In *FTC*, pages 276–282.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. In *NIPS*, pages 27263–27277.

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

Xiaoyu Zhang, Shaoyun Shi, Yishan Li, Weizhi Ma, Peijie Sun, and Min Zhang. 2024a. Feature-enhanced neural collaborative reasoning for explainable recommendation. *TOIS*, 43:1–33.

Xuan Zhang, Chao Du, Tianyu Pang, Qian Liu, Wei Gao, and Min Lin. 2024b. Chain of preference optimization: Improving chain-of-thought reasoning in llms. *NeurIPS*, 37:333–356.

Yurou Zhao, Yiding Sun, Ruidong Han, Fei Jiang, Lu Guan, Xiang Li, Wei Lin, Weizhi Ma, and Jiaxin Mao. 2024. Aligning explanations for recommendation with rating and feature via maximizing mutual information. In *CIKM*, pages 3374–3383.

A Appendix

The supplementary materials offer a thorough exposition of the methodological and experimental foundations of our work, covering implementation specifics, evaluation metrics, prompt engineering, and hyperparameter study.

A.1 Model Training & Inference

For a clearer illustration of the training and inference process, we conclude them in Algorithm 1. First, the hyper-parameters are specified (lines 1-2). Then, input the user and item IDs to obtain the low-dimensional user and item representations rich

Algorithm 1 Model training and inference

1: Indicate the number of MoE K.
2: Indicate the number of environments T.
Training Process
3: **while** Not converged **do**
4: **for** a batch B in training set **do**
5: Get the pre-trained NCF-encoded $Z_{u,CF}$ and $Z_{i,CF}$.
6: Get $Z_{u,init}$ and $Z_{i,init}$ with the same dimension as LLM embeddings via MoE.
7: Get the LLM-encoded $E_{u,t}$ and $E_{i,t}$ for all stages $t = 1, 2, ..., T$.
8: Get the causality-enhanced $Z_{u,final}$ and $Z_{i,final}$ via the Causality-Guided Representation Learner.
9: Derive the loss L for the prediction.
10: **end for**
11: **end while**
Inference Process
12: Load MoE and Causality-Guided Representation Learner.
13: Input u, i and rating to obtain the explanation $E_{u,i}$.

in collaborative information(line 5). Input the user and item IDs to obtain two types of representations: one is the user and item representations rich in collaborative information, which are mapped to the LLM embedding dimension via MoE(line 5-6); the other is the user and item feature representations encoded by the large language model (LLM)(line 7). Next, input the existing user and item representations into the Causality-Guided Representation Learner module to obtain the representations that reflect the latest features of users and items(line 8). Finally, input the obtained user and item representations into the designed chain-of-thought (CoT) prompt template to generate explanations, calculate losses, and optimize the parameters of DyCEX(line 9). During the inference, load the saved optimal parameters, and generate the corresponding explanations based on the user IDs, item IDs, and the ratings generated by the pre-trained model(line 12-13).

A.2 Details of Metrics

- **GPTscore** (Wang et al., 2023a) leverages large language models to evaluate text quality, providing a context-aware assessment. Fig. 6 illustrates the prompt designed to compute GPTscore on the Yelp dataset.

System Prompt: Score the given explanation against the ground truth on a scale from 0 to 100, focusing on the alignment of meanings rather than the formatting. Provide your score as a number and do not provide any other text.

Input Prompt: "prediction": The Hong Kong-style milk tea is rich and aromatic, and their new truffle mac and cheese is an absolute must-try delicacy. With its lively atmosphere and great value, this place has totally won me over. "reference": Authentic Hong Kong cafe with vibrant atmosphere, great value, and delicious truffle mac and cheese.

Response: 83

Figure 6: Prompt for GPTscore calculation.

- **BERTscore** (Zhang et al., 2019) computes the similarity between reference and generated texts using contextual embeddings from BERT. Given a reference sentence $x =< x_1, x_2, ..., x_n >$ and a generated sentence $\hat{x} = \langle \hat{x}_1, \hat{x}_2, \ldots, \hat{x}_m \rangle$, A sequence of word embeddings are first generated using BERT:

$$\begin{aligned} \text{BERT}(\langle x_1, x_2, ..., x_n \rangle) &= \langle \mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n \rangle \\ \text{BERT}(\langle \hat{x}_1, \hat{x}_2, ..., \hat{x}_m \rangle) &= \langle \hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, ..., \hat{\mathbf{x}}_m \rangle \end{aligned} \tag{11}$$

The similarity between two individual embeddings $(x_\mathbf{i}, \hat{x}_\mathbf{j})$ is measured using cosine similarity, which simply reduces to $x_\mathbf{i}^\top \hat{x}_\mathbf{j}$ since both embeddings are pre-normalized. With these definitions, the Precision, Recall, and F1-score are calculated as follows:

$$\text{BERT}^{\text{P}}_{\text{score}} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_\mathbf{j} \in \hat{x}} \underbrace{\max_{x_\mathbf{i} \in x} \overbrace{x_\mathbf{i}^\top \hat{x}_\mathbf{j}}^{\text{cosine similarity}}}_{\text{greedy matching}} \tag{12}$$

$$\text{BERT}^{\text{R}}_{\text{score}} = \frac{1}{|x|} \sum_{x_\mathbf{i} \in x} \underbrace{\max_{\hat{x}_\mathbf{j} \in \hat{x}} \overbrace{x_\mathbf{i}^\top \hat{x}_\mathbf{j}}^{\text{cosine similarity}}}_{\text{greedy matching}} \tag{13}$$

$$\text{BERT}^{\text{F1}}_{\text{score}} = 2 \times \frac{\text{BERT}^{\text{P}}_{\text{score}} \times \text{BERT}^{\text{R}}_{\text{score}}}{\text{BERT}^{\text{P}}_{\text{score}} + \text{BERT}^{\text{R}}_{\text{score}}} \tag{14}$$

- **BARTscore** (Yuan et al., 2021) conceptualizes the evaluation as a text generation task, assigning scores based on the probability of regenerating reference texts using the BART model.

- **USR** (Li et al., 2021) assesses the uniqueness of generated explanations by calculating the ratio of unique sentences to total sentences.

System Prompt: You will see a series of explanation summaries that describe what users liked and disliked about a specific movie. Each explanation reflects a different user's opinion about the movie. Based on all the explanations, please summarize the movie's overall strengths and weaknesses. Separate the movie's strengths and movie's weaknesses clearly, and provide a concise list for each. The strengths should highlight what most users appreciated about the movie, and the weaknesses should focus on common issues or criticisms that users pointed out. Please keep your summary under 200 words."

Input Prompt: Explanations as follows : Explanation1: **"**When will directors leave well enough alone. I haven't read the book ""Do Androids Dream of Electric Sheep"", but everyone loved the original version of this movie or the director would not have had the opportunity to do his cut version. Well I like the orignal and I'm glad I still have the VHS of it because when I bought the DVD and it didn't have Deckard's narration or the final scene I was truly disappointed. I can only hope this studio comes out with Classic Blade Runner, and the Directors version goes the way of New Coke." Explanation2:....

Response: "Strengths: - Thought-provoking exploration of humanity, androids, and identity.
- Groundbreaking visuals and immersive world-building.
- Harrison Ford's captivating performance.
- Dark, atmospheric dystopian setting with a haunting soundtrack.
Weaknesses: - Director's cut omits the original narration and final scene, which some users prefer.
- Some versions (e.g., director's cut) may lack elements that enhance the story for certain viewers."

Figure 7: Prompt for Item Attribute Feature Summarization.

A.3 Prompt Design

We design two dedicated prompts: one to summarize user preference features from users' reviews on all items within a single stage, and the other to summarize the attribute features of an item based on all user reviews for that item within the same stage. Only the prompt for item attribute feature summarization is illustrated in Fig. 7, as the two prompts differ merely by a minor adjustment of active-passive voice.

A.4 Baselines

We compare our model with five state-of-the-art baselines:

- **NRT** (Li et al., 2017) employs multi-task learning to predict ratings and generate tips for recommendations simultaneously based on user and item IDs. The generation component is a GRU.
- **PETER** (Li et al., 2021) is a personalized Transformer model that maps user and item IDs to generated explanations. It bridges IDs and words through a "context prediction" task. PETER is used instead of PETER+ due to the absence of word features in the datasets.
- **PEPLER** (Li et al., 2023) proposes sequential tuning and recommendation as regularization strategies to bridge the gap between prompts (incorporating user and item ID vectors) and the pre-trained transformer model for generating explanations.
- **XRec** (Ma et al., 2024) utilizes the encoded user/item embeddings from GNNs as implicit collaborative signals, which are then integrated into each layer of LLMs, enabling the generation of explanations.
- **G-Refer** (Li et al., 2025) utilizes a hybrid graph retrieval mechanism to extract explicit collaborative filtering signals from structural and semantic perspectives, which are then translated into human-understandable text and integrated into LLMs via knowledge pruning and retrieval-augmented fine-tuning, enabling the generation of personalized and interpretable explanations for recommendation.

A.5 Recommendation Performance

To evaluate the recommendation performance, we adopt two commonly used metrics: Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). Key comparisons are summarized in Table 3. Our model leverages a pre-trained NCF (He et al., 2017) to obtain user and item representations rich in collaborative information, and we further compare the recommendation performance of NCF against two categories of baseline models: pure recommendation models and explainable-recommendation multi-task models.:

- **PMF** (Mnih and Salakhutdinov, 2007) is a standard probabilistic matrix factorization method that characterizes users and items by latent factors.
- **SVD++** (Koren, 2008) leverages a user's interacted items to enhance the latent factors.
- **NETE** (Li et al., 2020) is a tailored GRU that incorporates a given feature into the decoding process to generate template-like explanations. It can also make recommendations.
- **PETER** (Li et al., 2021) is a Transformer-based model: it generates recommendations by mapping user-item sequential representations (via MLP) to scalar ratings.

Table 3: Recommendation performance comparison in terms of RMSE and MAE.

	Movies&TV		Yelp		Books	
	R↓	M↓	R↓	M↓	R↓	M↓
PMF	1.050	0.803	1.080	0.831	1.030	0.791
SVD++	0.972	0.732	1.040	0.779	0.993	0.762
NETE	0.957	0.725	1.050	0.765	0.985	0.751
PETER	0.961	0.727	1.020	0.771	0.987	0.748
NCF	**0.948**	**0.711**	**0.993**	**0.756**	**0.979**	**0.735**

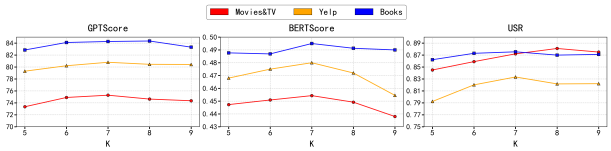

Figure 8: Impact of MoE Expert Number on Model Performance.

A.6 Hyperparameter Study

We investigate the variation in DyCEX's performance with respect to the number of MoE experts. Fig. 8 illustrates the impact of k (ranging from 5 to 9) on GPTScore, BERTscore precision and USR. A Comprehensive analysis of these three metrics reveals that the model achieves optimal performance when k = 7, a configuration that balances high semantic similarity and diversity of generated explanations. Specifically, insufficient experts limit MoE's feature learning capability, failing to effectively map complex user/item features to the LLM semantic space and thus reducing the semantic alignment between generated and ground-truth explanations. Excessive experts, by contrast, impair gating network efficiency: the gating mechanism struggles to make precise selections, leading to scattered weights and underutilized experts. Regarding diversity, it shows a non-linear trend with k: more experts bring diverse generation perspectives and richer sentences, but further increasing k to 9 or 10 yields negligible improvements, as newly added experts learn highly similar features without introducing novel patterns.