# Enhancing Explainable Rating Prediction through Annotated Macro Concepts

**Huachi Zhou**[1], **Shuang Zhou**[1], **Hao Chen**[1],
**Ninghao Liu**[2], **Fan Yang**[3], **Xiao Huang**[1,*]
[1]The Hong Kong Polytechnic University, Hong Kong, China
[2]University of Georgia, Georgia, USA
[3]University of Wake Forest, North Carolina, USA
[1]huachi.zhou@connect.polyu.hk, {csszhou,xiaohuang}@comp.polyu.edu.hk,
[1]sundaychenhao@gmail.com [2]ninghao.liu@uga.edu [3]yangfan@wfu.edu

## Abstract

Generating recommendation reasons for recommendation results is a long-standing problem because it is challenging to explain the underlying reasons for recommending an item based on user and item IDs. Existing models usually learn semantic embeddings for each user and item, and generate the reasons according to the embeddings of the user-item pair. However, user and item IDs do not carry inherent semantic meaning, thus the limited number of reviews cannot model users' preferences and item characteristics effectively, negatively affecting the model generalization for unseen user-item pairs. To tackle the problem, we propose the **Concept Enhanced Explainable Recommendation** framework (CEER), which utilizes macro concepts as the intermediary to bridge the gap between the user/item embeddings and the recommendation reasons. Specifically, we maximize the information bottleneck to extract macro concepts from user-item reviews. Then, for recommended user-item pairs, we jointly train the concept embeddings with the user and item embeddings, and generate the explanation according to the concepts. Extensive experiments on three datasets verify the superiority of our CEER model.

## 1 Introduction

In recommender systems, rating prediction is a crucial component for filtering user interest items from a vast pool of candidates. However, many recommender systems only offer rating prediction results without providing any explanations (Pan et al., 2022; Zhang et al., 2022). Personalized and appropriate explanations can greatly arouse user interest and enhance the overall user experience. Therefore, there is potential to deliver accurate rating predictions while generating personalized and suitable reasons for recommended items (Balog et al., 2019; Wang et al., 2022a).

Current explainable rating prediction models typically involve learning user and item embeddings and then generating explanations based on these embeddings (Chen et al., 2019, 2020b). For instance, they attribute semantic meanings to each user and item ID by reconstructing the associated review sentence using user and item embeddings (Wang et al., 2023; Gao et al., 2019). CETP (Li et al., 2021a) goes a step further by incorporating knowledge graphs as auxiliary information and leveraging graph neural networks to enhance the item embeddings. Similarly, MMCT utilizes multi-modal information to aid in training the user and item embeddings (Liu et al., 2023).

However, existing models still highly rely on the embeddings of user and item. Nevertheless, training user and item embeddings of explainable rate prediction face the following limitations: **(i). training sparsity:** The observed user-item pairs are sparse in comparison to all possible combinations. This sparsity issue prevents users and item embeddings from capturing sufficient semantic information by solely reconstructing the available reviews, thus negatively affecting the model's generalization in generating explanations for unseen user-item pairs. **(ii). Inference sparsity:** The absence of available information in the inference stage makes it challenging to generate appropriate explanations. There are some efforts that explicitly put the collected item features as prompting information to alleviate this issue (Li et al., 2020b; Cheng et al., 2023). However, it is hard to predict in advance which specific features will appear in the explanations.

It is challenging to solve the sparsity problems due to two reasons. **(i) ID insufficient semantics:** The initial embeddings for users and items lack semantic meanings, and are not effectively recognized by language models. The insufficient initial semantics exacerbate the learning of a detailed and accurate mapping to the semantic space.

[*]Xiao Huang is the corresponding author.

*Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11736–11748
August 11-16, 2024 ©2024 Association for Computational Linguistics

**(ii) Mixed high-level semantics:** The user reviews are complicated and may involve multiple high-level semantics. For example, a review that “I hate this guitar since the quality of the guitar is bad." could be identified by three high-level semantics: “negative emotion”, “instrument”, and “quality”. It is difficult to assist language models in recognizing user focus by providing an item feature.

To this end, we propose a framework named Concept Enhanced Explanation Recommendation (CEER). Our framework explicitly annotates high-level semantics, i.e., macro concepts that abstract item characteristics with similar semantics in reviews and embeds the potentially matched macro concepts into the user and item embeddings. The mined macro concepts enrich the semantic meaning of user and item embeddings without requiring additional data to learn. Our key contributions are summarized below:

- To improve the explanation generation, we leverage the macro concepts to address the issue of inadequate semantic information in user and item embeddings.
- To build macro concepts, we discover the micro characteristics of the item attended by the user from the reviews by maximizing the information bottleneck.
- We devise three tasks to enrich user and item embeddings motivated by integrating the macro concept into representations space.
- Extensive experiments on three real-world datasets prove the superiority of the proposed framework CEER.

## 2 Preliminary

**Problem Formulation.** We denote the user set and item set as $\mathcal{U} = \{u_1, u_2, ..., u_{|\mathcal{U}|}\}$ and $\mathcal{V} = \{v_1, v_2, ..., v_{|\mathcal{V}|}\}$, respectively. The rating score $r_{ij}$ assigned towards the interaction $(u_i, v_j)$ characterizes the degree of user preference. The observed reviews are organized within a matrix $\mathbf{X}$, where each element in the matrix represents a review and $\mathbf{X}_{ij}$ represents the review for the rating $r_{ij}$. $\mathbf{X}_{ij}$ is defined as a sentence denoted as $[x_1..., x_m, ..., x_M]$, where $x_m$ is the $m^{th}$ word in $\mathbf{X}_{ij}$. During inference, the available information for each interaction is limited to the user and item ID. In this work, we define explanation generation as a sequence-to-sequence prediction task. Specifically, given a user-item pair, we apply the user and item embeddings as soft prompts, asking the transformer to generate explanations, where each word in the explanation is predicted given the preceding words and soft prompt. For example, given the soft prompt $[u_i, v_j]$, the transformer is trained to predict the next word $x_1$. After multiple generation steps, the output from the transformer forms the sequence $[x_1..., x_m, ..., x_M]$. The notations are put into Appendix A.1

## 3 Methodology

This section introduces two primary components of our framework: a *macro concept annotator* and an *explanation generator*. The first component involves two steps: identifying the informative micro characteristics of items in the reviews and using LLMs to annotate macro concepts. The second component utilizes three tasks based on the annotated concepts to enhance user and item embeddings and facilitate explanation generation. The architecture sketch is provided in Figure 1.

### 3.1 Macro Concept Annotation

**Micro Characteristics Identification**. We represent each macro concept as a group of semantically similar micro characteristics of items that are carefully curated from the reviews. To serve as the micro characteristics, the selected words are required to be informative and capable of justifying the rationale behind recommendations.

To measure the informativeness of the words, we need to first quantify the informative level of the words in the corpus. Pre-trained language models (PLMs) are trained on vast amounts of text data and can recognize which words are more informative in conveying the explanation. To learn the informative level quantification, we maximize the following objective, i.e., *information bottleneck* (TISHBY, 2000) $M(\mathbf{X}_{ij}, r_{ij})$:

$$\max M(\mathbf{X}_{ij}, r_{ij}) = \max I(\mathbf{T}_{ij}, r_{ij}) - \tau \cdot I(\mathbf{T}_{ij}, \mathbf{X}_{ij}), \quad (1)$$

where $I(\cdot, \cdot)$ represents mutual information between two variables; $\tau$ is a trade-off hyper-parameter; $\mathbf{T}_{ij} \in \mathbb{R}^{M \times d}$ signifies the intermediate word representation from PLMs whose informative levels can be measured. Here, $d$ denotes the word dimension in the frozen PLM, which, for instance, is 768 when using BERT (Devlin et al.,

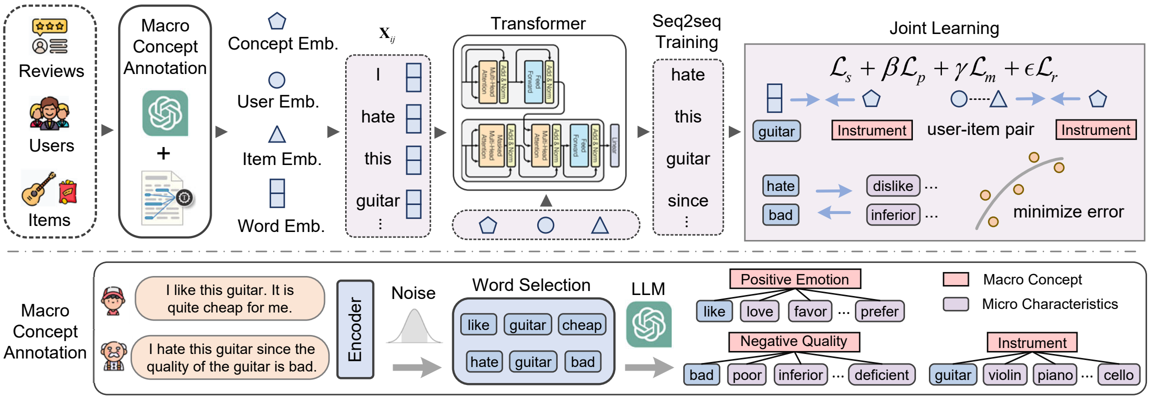


Figure 1: The figure shows the proposed CEER framework. The lower part demonstrates how we annotate the macro concepts. The upper part shows how to leverage the annotated concepts to enrich user and item embeddings.

2019) as the frozen PLM. In Eq. (1), the first term maximally preserves the information related to rating prediction while the second term (Alemi et al., 2016) serves as the regularization term that constrains the amount of information in intermediate representations encoded from $\mathbf{X}_{ij}$.

The variational approximation of the information bottleneck of traditional approaches applies restrictive constraints to the pre-trained language models as the second term (Tu and Li, 2022). The posterior distribution of the intermediate representations may not be standard Gaussian distribution (Tu et al., 2022). Some recent efforts have tried more flexible distributions, e.g., sample-based representations of variational distributions (Fang et al., 2019, 2022). They could learn expressive intermediate representations but are unable to explicitly quantify the informative level of the words. To achieve this goal, we inject noise into word representations (Guan et al., 2019). The extent to which these representations tolerate noise can serve as a reflection of their informativeness, as informative words play an important role in understanding the corpus and are sensitive to the introduced noise.

To facilitate the second-term calculation, we synthesize the word representations with noise as $\mathbf{T}_{ij}$:

$$\mathbf{T}_{ij} = (1 - \sigma) \odot PLM(\mathbf{X}_{ij}) + \sigma \odot \epsilon, \quad (2)$$

where $PLM(\cdot)$ provides the original word representations from PLMs, and $\epsilon \in \mathbb{R}^{M \times d}$ denotes the random noise independently sampled from standard Gaussian distribution while the learnable vector $\sigma \in \mathbb{R}^{M}_{[0,1]}$ controls the magnitude of noise. The minimization of $I(\mathbf{T}_{ij}, \mathbf{X}_{ij})$ is calculated as:

$$\min\ H(\mathbf{X}_{ij}) - H(\mathbf{X}_{ij}|\mathbf{T}_{ij}) \approx \max\ \log \sigma, \quad (3)$$

where we assume $P(\mathbf{X}_{ij}|\mathbf{T}_{ij}) \approx P(\mathbf{T}_{ij})$ and follows the multivariate Gaussian distribution with zero covariance. Next we could use $1 - \sigma_{x_m}$ to identify the informative level of word $x_m$ in $\mathbf{X}_{ij}$. Then the informative level for each word in $\mathbf{X}_{ij}$ is:

$$[1 - \sigma_{x_1}, ..., 1 - \sigma_{x_m}, ..., 1 - \sigma_{x_M}]\,. \quad (4)$$

To calculate the first term, we transform word representations into ratings and minimize the difference between predictions and labels. In this process, we employ a rating-specific message-passing encoder for transformation (Shuai et al., 2022).

After obtaining $\sigma$, we exploit the LLMs to perform the macro concept annotation for top $k$ important micro characteristics. Here we use LLM ChatGPT, i.e., gpt-3.5-turbo and take one demonstration following (Lou et al., 2023; Zhang et al., 2023) as an example:

> ***[Task Description]*** *User: Organize the following keywords into groups based on their semantic relationships, and give a concept name to each group.* ***[Demonstrations]*** *Input: [Expensive, Cheap]. Output: {Economy: [Expensive, Cheap]}* ***[Test Instance]*** *Input: [Guitar, Hate, Quality]. Output: {Instrument: [Guitar], Negative Emotion: [Hate], Quality: [Quality]}.*

In this way, we obtain the macro concept labels for $k$ informative micro characteristics from the historical reviews and create a padded concept label for the remaining words. The words in $\mathbf{X}_{ij}$ are labeled as $\{\mathbf{c}_{x_1}, ..., \mathbf{c}_{x_m}, ..., \mathbf{c}_{x_M}\}$, where $\mathbf{c}_{x_m}$ is the macro concept ID for word $x_m$.

### 3.2 Explanation Generator

In this component, our goal is to utilize the annotated macro concepts to enrich user/item embeddings. Specifically, we embed the relationships between users/items and concepts, as well as concepts and reviews, in representations through three tasks. These three tasks collaborate to utilize macro concepts as an intermediary to align user and item embeddings with the review embeddings.

#### 3.2.1 User/item-concept Relationship Modeling

The first task user/item-concept relationship modeling utilizes user and item embeddings to aid in predicting macro concepts in the associated review. To train the prediction of macro concept distribution, we need to craft macro concept labels $\mathbf{W}_{ij} = \left[w_1, ..., w_q, ..., w_{|\mathcal{C}|}\right] \in \mathbb{R}^{|\mathcal{C}|}$ for review $\mathbf{X}_{ij}$. The macro concept label $w_q$ is determined by the cumulative informative level of the words in $\mathbf{X}_{ij}$ associated with that concept $c_q$:

$$\sum_{m=1}^{M} \mathbb{I}(c_{x_m} = c_q)(1 - \sigma_{x_m}), \qquad (5)$$

where the indicator $\mathbb{I}(\cdot)$ take value $1$ if the condition holds otherwise $0$.

After obtaining the macro concept labels for review $\mathbf{X}_{ij}$, we come to train the user/item embeddings to predict the macro concept distribution $\hat{\mathbf{W}}_{ij}$ in the associated review, defined as:

$$\hat{\mathbf{W}}_{ij} = MLP(\mathbf{e}_{u_i}, \mathbf{e}_{v_j}), \qquad (6)$$

where $\mathbf{e}_{u_i}, \mathbf{e}_{v_j}$ denotes the user and item embeddings respectively. Next, we use Kullback-Leibler divergence loss to penalize the divergence between the predicted macro concept distribution and the labels, defined as:

$$\mathcal{L}_p = \mathcal{L}_{KL}(\hat{\mathbf{W}}_{ij} || \mathbf{W}_{ij}) = \sum_{q=1}^{|\mathcal{C}|} w_q \cdot \log \frac{w_q}{\hat{w}_q}. \qquad (7)$$

After training, the user and item embeddings could be utilized to recognize the potential macro concepts in the inference. Then, we obtain a composite concept embedding by the weighted sum of all macro concept embeddings with the predicted importance: $\sum_{q=1}^{|\mathcal{C}|} \hat{w}_q \mathbf{e}_{c_q} / \sum_{q=1}^{|\mathcal{C}|} \hat{w}_q$. The composite macro concept embedding is appended behind the user and item embedding to instruct the transformer to generate relevant explanations.

#### 3.2.2 Concept-review Relationship Modeling

The second task concept-review relationship modeling brings the words in reviews and associated macro concept embeddings closer together. Through this task, we proceed to enhance the words in reviews' awareness of the associated macro concept within the representation space. To reach this objective, we reduce the uncertainty about each word mapping to its associated macro concept. For simplicity, we use the inner product to compute the uncertainty and optimize it with the cross-entropy loss, driving the movement of word embeddings to the macro concept embedding:

$$\mathcal{L}_m = -\sum_{m=1}^{M} \sum_{q=1}^{|\mathcal{C}|} \mathbb{I}(c_{x_m} = c_q) \cdot \log(p_{x_m, c_q}), \qquad (8)$$

where $\log(p_{x_m, c_q})$ is the uncertainty between the word $x_m$ and the macro concept $c_q$, computed by inner product of embedding: $p_{x_m, c_q} = e_{x_m}^{\top} \cdot e_{c_q}$. It encourages words and associated concepts to exhibit proximate representations.

#### 3.2.3 User/item-review Relationship Modeling

The third task user/item-review relationship modeling assists the transformer in considering words belonging to the same macro concepts when generating the next word. Traditional approaches employ the maximum likelihood function to reconstruct the original reviews. However, it ignores the diversity of the language expression. In practice, each position in the sentence could have multiple word choices, e.g., semantically similar words. To achieve this, we substitute the word in the review with multiple alternatives in the same macro concept. The loss can be reformulated as:

$$\mathcal{L}_s = -\frac{1}{M} \sum_{m=1}^{M} (\log p(x_m) + \alpha \log p(x'_m)), \qquad (9)$$

where $x'_m$ denotes a randomly selected word within the same macro concept as $x_m$ and $\alpha$ denotes a hyper-parameter to control the loss term. Here, we only incorporate one substitution. More substitutions may take the risk of introducing noise and generating contextually inappropriate explanations.

#### 3.2.4 Training Objective

Besides the above tasks, we jointly perform rating prediction following previous efforts (Li et al., 2023b) using the loss function defined as:

$$\mathcal{L}_r = \left(MLP(\mathbf{e}_{u_i}, \mathbf{e}_{v_j}) - r_{ij}\right)^2. \qquad (10)$$

Finally, we unify these tasks in a linear combination form to train the transformer. The joint learning objective is indicated by:

$$\mathcal{L} = \mathcal{L}_s + \beta\mathcal{L}_m + \gamma\mathcal{L}_p + \mathcal{L}_r, \tag{11}$$

where $\beta, \gamma$ are trade-off hyper-parameters to balance the effect of different objectives of the model.

## 4 Experiments

In this section, we perform comprehensive experiments to validate the effectiveness of CEER and gain insights into its behavior. We aim to answer the following research questions: **RQ1:** How well do the explanations and ratings generated by CEER and baselines align with the actual labels in terms of text quality? **RQ2:** How does the quality of explanations generated by all models in terms of interpretability? **RQ3:** How do different tasks contribute to the whole model performance? **RQ4:** What impact do different hyper-parameter settings of the proposed tasks have on CEER? **RQ5:** How does information bottleneck perform compared with other micro characteristics identification methods for macro concept annotation?

Table 1: Detailed datasets statistics.

| Datasets | # Users | # Items | # Interactions | # Reviews | # Concepts |
|---|---|---|---|---|---|
| Instruments | 8,292 | 695 | 22,831 | 7,380 | 425 |
| Home | 66,519 | 27,679 | 539,403 | 538,204 | 551 |
| Automotive | 20,886 | 1,573 | 57,722 | 16,866 | 477 |

### 4.1 Dataset Processing

In our experiments, we use three real-world Amazon datasets (He and McAuley, 2016), i.e., Musical Instruments, Home and Kitchen, and Automotive, to evaluate the overall performance. The dataset statistics are shown in Table 1. The detailed description of these datasets is illustrated in Appendix A.2.

### 4.2 Experimental Setup

We implement our framework by PyTorch using NLG4RS ecosystem (Li et al., 2020a) and run it on NVIDIA Tesla V100S PCIe, 32GB memory. The learning rate is set as 0.001. All models are optimized by the Adam optimizer and the embedding dimension is set as 64 for a fair comparison. The batch size equals 256. We truncate explanations with a maximum length of 20 and the budget $k$ is set as 5000 for each dataset. We search the number of transformer layers and attention heads in each layer for the base explanation generator from $\{1, 2, 3, 4, 5\}$. And we also search the trade-off factor $\alpha, \beta, \gamma$ from $\{1e-5, 1e-4, 1e-3, 1e-2, 1e-1, 1\}$ and the dropout rate from $\{0, 0.2, 0.4, 0.6, 0.8\}$. The search strategy is grid search. We also set the patience threshold on the validation set to stop model training earlier and the optimal performance on the validation set is selected to report performance on the test set. We run each model five times and report the average performance.

### 4.3 Evaluation Metrics

To evaluate the explanations, we categorize all the adopted metrics into two groups, which evaluate the text quality and the quality of interpretation, respectively. In the first text quality evaluation group, we have BLEU-1, BLEU-4, MAE, and RMSE. In the second interpretability evaluation group, we have USR, FCR, FMR, `Entail`, and `Consistency`. The detailed introduction of these metrics is illustrated in Appendix A.3.

### 4.4 Baseline Methods

We integrate the following baselines to conduct the comparison: RGCL (Shuai et al., 2022) applies a graph contrastive learning framework to learn user-item rating prediction; NRT (Li et al., 2017) learns explanation generation via gated recurrent units; NETE (Li et al., 2020b) generates explanations with controlled neural templates; PETER (Li et al., 2021b) introduces contextualization loss and item features to enhance the quality of the explanation; SAER (Yang et al., 2022) adopts an extract-and-refine architecture to effectively generate explanations; PEPLER (Li et al., 2022) fuses user/item IDs into soft prompt and fine-tunes GPT-2.

To demonstrate the effectiveness of different components, we also introduce three variants of the CEER: 1) P-CEER: it removes all the proposed tasks and adopts the pure transformer to generate explanations. 2) C-CEER: it adds the concept-review relationship modeling task to the P-CEER. 3) CW-CEER: it further adds the user/item-review relationship modeling task to the C-CEER.

To examine the usefulness of the information bottleneck in micro characteristics identification, we incorporate the following variants: (1) GPT-3.5: Directly annotate the macro concepts in the historical reviews with ChatGPT and remove the information bottleneck. (2) $\beta$-VAE: Replace the information bottleneck with the $\beta$-VAE (Higgins et al., 2017) and use the learned variance as the informative level.

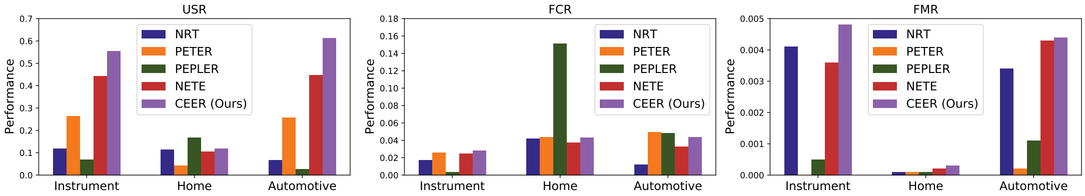


Figure 2: Explanation quality in terms of USR, FCR, FMR.

Table 2: Text quality comparison of all methods in terms of BLEU-1 and BLEU-4.

| Methods | Instruments | | Home | | Automotive | |
|---|---|---|---|---|---|---|
| | BLEU-1 | BLEU-4 | BLEU-1 | BLEU-4 | BLEU-1 | BLEU-4 |
| NRT (2017) | 12.8507 | 0.7109 | 12.1058 | 0.6764 | 15.3298 | 0.8062 |
| PETER (2021) | 14.9794 | 0.8345 | 11.5656 | 0.5737 | 15.9109 | 0.8116 |
| PEPLER (2023) | 4.5601 | 0.2289 | 8.8215 | 0.4074 | 2.4393 | 0.0706 |
| NETE (2020) | 13.8053 | 0.8949 | 10.6288 | 0.6059 | 14.6157 | 0.6916 |
| SAER (2022) | 13.7924 | 0.1210 | 12.5891 | 0.2763 | 13.9785 | 0.0000 |
| CEER | **16.1862** | **1.0489** | **12.6839** | **0.6887** | **16.1159** | **0.9010** |

Table 3: Rating prediction of all methods in terms of RMSE and MAE.

| Methods | Instruments | | Home | | Automotive | |
|---|---|---|---|---|---|---|
| | RMSE | MAE | RMSE | MAE | RMSE | MAE |
| RGCL (2022) | 0.9224 | **0.6425** | **1.0612** | 0.7703 | 0.9883 | 0.6842 |
| NRT (2017) | 1.0930 | 0.6751 | 1.3312 | 0.8508 | 1.2184 | 0.7312 |
| PETER (2021) | 0.9412 | 0.7229 | 1.1073 | 0.8487 | 1.0259 | 0.7801 |
| PEPLER (2023) | 1.0183 | 0.6556 | 1.0792 | 0.7610 | 1.2007 | 0.7006 |
| NETE (2020) | 0.9400 | 0.7139 | 1.1207 | 0.8691 | 1.0261 | 0.7740 |
| SAER (2022) | 1.0970 | 0.9156 | 1.1731 | 0.9409 | 1.0429 | 0.8131 |
| CEER | **0.9210** | 0.6680 | 1.0789 | **0.7490** | **0.9693** | **0.6771** |

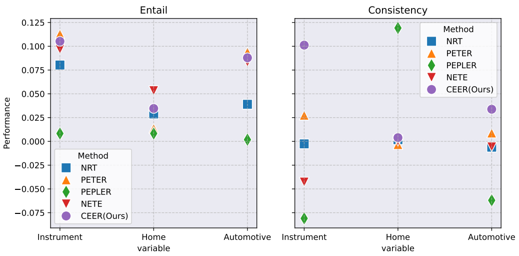


Figure 3: Explanation quality w.r.t `Entail` and `Consistency`.

## 4.5 Text Quality Analysis (RQ1)

### 4.5.1 Word-level Relevance

First, we evaluate the text quality of explanations in terms of BLEU-1 and BLEU-4 and exhibit the results in Table 2. We have several observations.

1) **The performance of baselines does not consistently improve with increasing complexity across the three datasets.** The NRT model is simple and designed for generating abstractive tips, while the PETER, NETE, and SAER have more complicated architectures. Surprisingly, we find that the complex models do not consistently outperform the simpler NRT model in all cases. Sparse data during inference and training hinders the effective learning of user and item embeddings and limits the ability of advanced language models to generate accurate explanations. Although PEPLER, which fine-tunes GPT-2, fails to yield satisfactory results. This may be because they do not have harvest knowledge about ID during pre-training. 2) **The proposed CEER outperforms all baselines across the three datasets.** Compared to the baseline methods, the CEER framework consistently produces better results. We infer that this improvement is due to the inclusion of macro concepts, which offer higher-level semantics that enhance the learning of ID embeddings and provide additional information for generating explanations.

### 4.5.2 Rating Prediction

We also examine the recommendation performance and present the results in Table 3. **We find that the proposed CEER performs consistently better than the explainable recommendation baselines.** And it performs even better than the dedicated rating prediction baseline RGCL in most cases. We infer that incorporating the high-level semantics, i.e., macro concepts refines the user and item representations, leading to improved recommendation performance. As a result, our model exhibits strong potential for real-world applications, as it not only offers explanations with high text quality but also delivers accurate rating predictions, addressing the specific requirements of the system.

## 4.6 Interpretability Quality Analysis (RQ2)

### 4.6.1 Personalization

We also evaluate the degree of informativeness and diversity, i.e., personalization of explanations, and report the results in Figure 2. Notably, in Subsection 4.6, we only select the explainable baselines with superior performance for the RQ1 (i.e., PEPLER, PETER, NETE, and NRT) for comparison. From the results, we have several observations. **First, the higher diversity of explanations does not indicate that the features are accurately**

**included in each explanation.** It indicates that improving informativeness and diversity simultaneously is a challenging task. **Second, the proposed model generally outperforms the selected baselines in most cases, especially on the Instrument and Automotive datasets**. It indicates that the proposed model can not only generate more diverse explanations but also match the corresponding features for the explanations.

#### 4.6.2 Entail & Consistency

We evaluate the interpretability from the sentiment perspective and display the results in Figure 3. We notice that **the CEER achieves relatively better performance in producing emotionally coherent and factually accurate explanations in most cases.** During training, words become associated with the relevant concepts they belong to. This alignment between positive or negative concepts and the corresponding words enables CEER to consistently generate emotionally coherent explanations. Additionally, user and item representations tend to align with the representations of related concepts. It facilitates the incorporation of related facts into the explanation generation process.

### 4.7 Ablation Study (RQ3)

To explore the individual contributions of different tasks to the overall model performance, we analyze three variants and report the results in Table 4. Notably, in Subsections 4.7 and 4.8, we select a subset of metrics for evaluating text quality and interpretability rather than including all available metrics to save layout space. We have several observations: 1) **The P-CEER has the worst performance.** It does not include any of the designed tasks. The bad performance underscores the necessity of integrating the designated task to enhance model performance. 2) **By taking each designed task gradually, CEER, CW-CEER, and C-CEER demonstrate performance improvements compared with its predecessor CW-CEER, and C-CEER, P-CEER correspondingly.** It demonstrates the effectiveness of each task in addressing the issue of limited generalization caused by sparsity. The overall results validate the effectiveness of the proposed model.

### 4.8 Hyper-parameter Sensitivity (RQ4)

We further analyze the hyper-parameters $\alpha$, $\beta$, $\gamma$ in the proposed regularization terms. We conduct experiments on the Automotive dataset and present the results in Figure 4. We have several observations. 1) **Across all selected metrics, the model is sensitive to extreme hyper-parameter values.** Hyper-parameters that are too large, such as 1, or too small, such as $1e-5$, do not lead to the best performance of the model. 2) **Different evaluation metrics require different hyper-parameters to reach their optimal values.** This observation implies that it is difficult to have a hyper-parameter setting that could achieve optimal results considering multiple evaluation metrics together for the whole model. 3) **Compared with $\beta$ and $\gamma$, $\alpha$ generally prefers relatively small values.** We infer that the user/item-review relationship modeling task may act as noise and hinder the modeling of the language pattern in the explanations when $\alpha$ is large. All the results imply that the CEER is sensitive to the hyper-parameter values and it is important to choose appropriate values.

### 4.9 Characteristics Identification Study (RQ5)

In this subsection, we first conduct the comparison between different micro characteristics identification methods and display the results in Table 5. Then we present an example from the Amazon Instrument dataset to intuitively show the identified micro characteristics, annotated macro concept, and the generated explanation in Figure 5.

**Performance Comparison.** We compare the information bottleneck method with two other choices using the same annotation prompt, and explanation generator. The two choices are compared on the Amazon Instrument dataset and the Amazon Home dataset, respectively. From the results, we find that **direct annotation without information bottleneck negatively affects the model performance.** We infer that although LLMs are proficient at text comprehension, they are not trained to assign consistent macro concepts to similar words, which leads to poor performance. And the $\beta$-VAE also performs worse than the information bottleneck in most cases. $\beta$-VAE does not quantitatively define the importance of the words. And the learned variance may not be a good indicator to differentiate word importance although the small variance may mean that the meaning of the word does not change a lot across contexts, such as stop words. These results jointly demonstrate the effectiveness of the information bottleneck method.

**Case Study.** We showcase the informative level of each word in the historical reviews associated with

Table 4: Ablation study results on three datasets.

| Methods | Instruments | | | | | Home | | | | | Automotive | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | BLEU-1 | BLEU-4 | USR | FCR | FMR | BLEU-1 | BLEU-4 | USR | FCR | FMR | BLEU-1 | BLEU-4 | USR | FCR | FMR |
| P-CEER | 12.8870 | 0.7512 | 0.2138 | 0.0158 | 0.0038 | 11.6297 | 0.6568 | 0.0577 | 0.0299 | 0.0002 | 14.5324 | 0.7367 | 0.4688 | 0.0311 | 0.0012 |
| C-CEER | 14.6413 | 0.9538 | 0.3303 | 0.0250 | **0.0051** | 12.3248 | 0.6598 | 0.0652 | 0.0330 | **0.0003** | 14.9229 | 0.8626 | 0.5402 | 0.0424 | 0.0030 |
| CW-CEER | 15.8464 | 0.8565 | 0.4904 | 0.0261 | 0.0038 | 12.6099 | 0.6344 | 0.0791 | **0.0475** | 0.0002 | 15.4576 | 0.7608 | **0.6217** | **0.0473** | 0.0027 |
| CEER | **16.1862** | **1.0489** | **0.5552** | **0.0281** | 0.0048 | **12.6839** | **0.6887** | **0.0890** | 0.0431 | **0.0003** | **16.1159** | **0.9010** | 0.6120 | 0.0435 | **0.0044** |

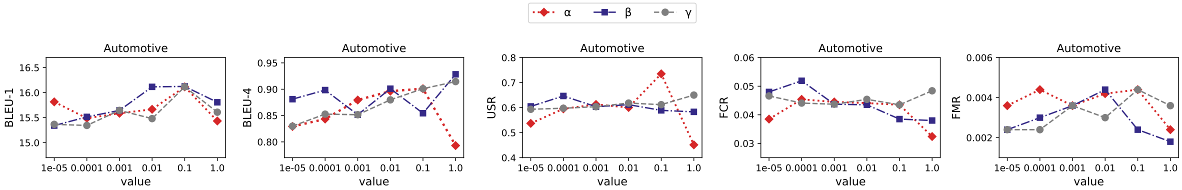
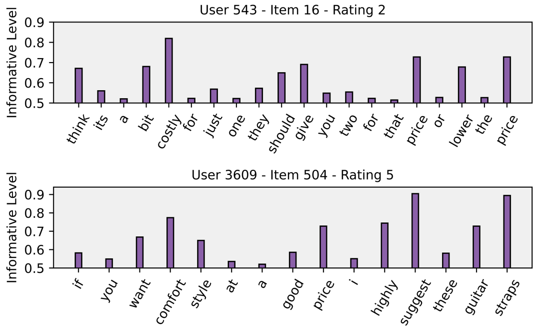


Figure 4: Hyper-parameter sensitivity results of different regularization terms on Automotive dataset.



(a) The informative level of words in reviews.

| | |
|---|---|
| P-CEER | I love this one of the best I have ... |
| Ground-truth | This real strap is nice. The price is even better ... |
| CEER (Ours) | I am very happy with this strap. The price is pretty nice and it is a very good quality ... |

(b) Explanation comparison.

Measurement

quantity, pounds, gallons, quarter, unit, volume, length, inches, dozen, diameter, weights, square, degrees, ounces, size, level, bulk, shape ...

Positive quality

superiority, fabulous, distinctive, loose, precious, perfect, terrific, fine, valuable, helpful, flawless, skinny, incredible, exceptional, special ...

(c) Examples of the annotated words in the concepts.

Figure 5: Examples from the Instrument dataset.

Table 5: Model performance comparison with different micro characteristics identification methods.

| Methods | BLEU-1 | BLEU-4 | USR | FCR | FMR |
|---|---|---|---|---|---|
| GPT-3.5 | 15.9572 | 0.8701 | 0.4763 | 0.0233 | 0.0038 |
| CEER | **16.1862** | **1.0489** | **0.5552** | **0.0281** | **0.0048** |
| $\beta$-VAE | 12.3631 | **0.7028** | 0.0819 | 0.0389 | **0.0003** |
| CEER | **12.6839** | 0.6887 | **0.0890** | **0.0431** | **0.0003** |

the given user-item pair. Then, we demonstrate the annotated concepts that drive the model to generate the explanation involved by the user and item in this example. First, we visualize the informative level of each word representation of two historical reviews associated with user 543 and item 504, respectively. We observe that those keywords that explain the reason for the rating are assigned significantly lower amounts of noise. For example, in the first review of subfigure (a), the words "costly" and "price" describe the reason for user dissatisfaction and their informative levels are very high. It demonstrates that the first component could select the informative words. That facilitates the cost reduction for LLMs to annotate concepts.

Second, we compare the explanations generated by the CEER and the P-CEER, i.e., the transformer model for the interaction between user 543 and item 504, and show it in subfigure (b). The target rating is 5. We observe that the P-CEER only describes the user's positive sentiment towards the item and does not contain any useful information. The generated explanation from CEER reveals the reason for the high rating is due to the price and the positive quality. We also exhibit the top two prioritized concepts and words associated with them in the subfigure (c). It shows to a certain extent that the high-level concept does drive the transformer to generate semantically relevant sentences. And the associated words for the concept demonstrate that the LLMs could appropriately assign a concept label to semantic similar words.

## 5 Related Work

### 5.1 Sentence-based Explanation Generation

Compared with examining which feature plays a more important role in driving the interaction (Wang et al., 2018a, 2022b), sentence-based explanations for rating prediction offer natural language explainable information about why the user assigns particular ratings (Hua et al., 2023; Wu et al., 2024). Some efforts only manipulate the historical reviews to generate the explanations (Le and

Lauw, 2020; Pugoy and Kao, 2021), other efforts attempt to integrate varied auxiliary information into explanation generation. Knowledge graph provides an organized way to access facts (Chen et al., 2024; Shengyuan et al., 2024). CETP (Li et al., 2021a) selects the triples from the knowledge graph and incorporates them into explanation generation. Further, METER (Geng et al., 2022; Liu et al., 2023) incorporates multi-modality information to encourage the model to write more faithful and diverse explanations. LLMs have strong text manipulation abilities (Dong et al., 2024; Zhang et al., 2024). PRAG (Xie et al., 2023) leverages LLMs and reformulates the explanation generation as a question-answering task. Additionally, the tips written by users show how they feel about the interaction and can also be utilized to generate a comprehensive explanation (Zhu et al., 2023). To generate coherent explanations, AESG tries to incorporate the syntax graph dependency tree to improve the quality of explanations (Hu et al., 2023).

### 5.2 Review-based Rating Prediction

Review-based rating prediction involves predicting a numerical rating for a user-item pair by modeling the text of a review provided by a user (Harrag et al., 2019). Early effort introduces the review-level attention mechanism to model the review for rating prediction (Chen et al., 2018). It ignores the fine-grained information in the review. NPA (Wu et al., 2019) leverages user ID to incorporate word-level and review-level information to improve performance attentively. Further, DAML (Liu et al., 2019) exploits the mutual information between the user and the item extracted from information that the convolutional neural network attends to. Similarly, CARP (Li et al., 2019) models mutual information and uses the capsule network to extract high-level information. EDMF (Liu et al., 2022) extracts the useful feature from the review text to further improve performance. Graph neural networks (GNNs) capture global relationships between nodes that are indirectly connected (Dong et al., 2023; Chen et al., 2020a). Efforts adopt GNNs to model user and item relationships and the review features can be regarded as edge features (Qiao et al., 2022; Shuai et al., 2022).

## 6 Conclusion and Future Work

In this paper, we present the CEER framework, designed to enrich user and item embeddings with high-level semantics, i.e., macro concepts to improve the explanation generation. The framework identifies micro characteristics of items from associated reviews and utilizes LLMs to annotate macro concepts. These macro concepts act as intermediaries to align the user and item embeddings with review embeddings to obtain more semantic information. And we achieve this through three tasks that focus on establishing relationships between users/items, macro concepts, and reviews. We conduct extensive experiments, and the results confirm the effectiveness of CEER. Our future work will involve adapting our method to annotate concepts from various data sources, such as images, and enhancing the explanation generation process (Li et al., 2023a).

## Acknowledgments

The work described in this paper was fully supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. PolyU 25208322).

## Limitations

In this study, we annotate the macro concept from the reviews to assist the explanation generation. The existing framework is constrained to annotating macro concepts solely from text and lacks the capability to expand to additional data sources like images. And we annotate macro concepts from a series of micro item characteristics, i.e., words. However, the semantic meaning of words differs under different contexts. Moreover, we employ LLMs for annotating macro concepts and training a small language model for explanation generation instead of fine-tuning LLMs. While fine-tuned GPT-2 may outperform the suggested framework when available reviews are rich, our framework demonstrates relatively better effectiveness in situations with extremely sparse reviews and does not incur significant computational costs.

## Ethics Statement

In this study, all the Amazon datasets are publicly available and have been extensively used in research related to recommendation systems. All the baseline codes are open-sourced. And we adhere to the ACL Code of Ethics[1].

[1] https://www.acm.org/code-of-ethics

## References

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. 2016. Deep variational information bottleneck. In *International Conference on Learning Representations*.

Krisztian Balog, Filip Radlinski, and Shushan Arakelyan. 2019. Transparent, scrutable and explainable user models for personalized recommendation. In *SIGIR*, pages 265–274.

Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. 2018. Neural attentional rating regression with review-level explanations. In *WWW*, pages 1583–1592.

Hao Chen, Yue Xu, Feiran Huang, Zengde Deng, Wenbing Huang, Senzhang Wang, Peng He, and Zhoujun Li. 2020a. Label-aware graph convolutional networks. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 1977–1980.

Shengyuan Chen, Qinggang Zhang, Junnan Dong, Wen Hua, Qing Li, and Xiao Huang. 2024. Entity alignment with noisy annotations from large language models.

Zhongxia Chen, Xiting Wang, Xing Xie, Mehul Parsana, Akshay Soni, Xiang Ao, and Enhong Chen. 2020b. Towards explainable conversational recommendation. In *IJCAI*, pages 2994–3000.

Zhongxia Chen, Xiting Wang, Xing Xie, Tong Wu, Guoqing Bu, Yining Wang, and Enhong Chen. 2019. Co-attentive multi-task learning for explainable recommendation. In *IJCAI*, pages 2137–2143.

Hao Cheng, Shuo Wang, Wensheng Lu, Wei Zhang, Mingyang Zhou, Kezhong Lu, and Hao Liao. 2023. Explainable recommendation with personalized review retrieval and aspect learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 51–64. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186.

Junnan Dong, Qinggang Zhang, Xiao Huang, Keyu Duan, Qiaoyu Tan, and Zhimeng Jiang. 2023. Hierarchy-aware multi-hop question answering over knowledge graphs. In *Proceedings of the ACM Web Conference 2023*, pages 2519–2527.

Junnan Dong, Qinggang Zhang, Chuang Zhou, Hao Chen, Daochen Zha, and Xiao Huang. 2024. Cost-efficient knowledge-based question answering with large language models.

Le Fang, Chunyuan Li, Jianfeng Gao, Wen Dong, and Changyou Chen. 2019. Implicit deep latent variable models for text generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3946–3956.

Xianghong Fang, Jian Li, Lifeng Shang, Xin Jiang, Qun Liu, and Dit-Yan Yeung. 2022. Controlled text generation using dictionary prior in variational autoencoders. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 97–111.

Jingyue Gao, Xiting Wang, Yasha Wang, and Xing Xie. 2019. Explainable recommendation through attentive multi-view learning. In *AAAI*, pages 3622–3629.

Shijie Geng, Zuohui Fu, Yingqiang Ge, Lei Li, Gerard de Melo, and Yongfeng Zhang. 2022. Improving personalized explanation generation through visualization. In *ACL*, pages 244–255.

Chaoyu Guan, Xiting Wang, Quanshi Zhang, Runjin Chen, Di He, and Xing Xie. 2019. Towards a deep and unified understanding of deep neural models in nlp. In *International conference on machine learning*, pages 2454–2463. PMLR.

Fouzi Harrag, AbdulMalik Al-Salman, and Alaa Alqahtani. 2019. Prediction of reviews rating: A survey of methods, techniques and hybrid architectures. *Journal of Digital Information Management*, 17(3):164.

Ruining He and Julian J. McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *WWW*, pages 507–517.

Irina Higgins, Loic Matthey, Arka Pal, Christopher P Burgess, Xavier Glorot, Matthew M Botvinick, Shakir Mohamed, and Alexander Lerchner. 2017. beta-vae: Learning basic visual concepts with a constrained variational framework. *ICLR (Poster)*, 3.

Yidan Hu, Yong Liu, Chunyan Miao, Gongqi Lin, and Yuan Miao. 2023. Aspect-guided syntax graph learning for explainable recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):7768–7781.

Wenyue Hua, Lei Li, Shuyuan Xu, Li Chen, and Yongfeng Zhang. 2023. Tutorial on large language models for recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, pages 1281–1283. ACM.

Trung-Hoang Le and Hady W. Lauw. 2020. Synthesizing aspect-driven recommendation explanations from reviews. In *IJCAI*, pages 2427–2434.

Chenliang Li, Cong Quan, Li Peng, Yunwei Qi, Yuming Deng, and Libing Wu. 2019. A capsule network for recommendation and explaining what you like and dislike. In *SIGIR*, pages 275–284.

Junyi Li, Wayne Xin Zhao, Zhicheng Wei, Nicholas Jing Yuan, and Ji-Rong Wen. 2021a. Knowledge-based review generation by coherence enhanced text planning. In *SIGIR*, pages 183–192.

Lei Li, Li Chen, and Yongfeng Zhang. 2020a. Towards controllable explanation generation for recommender systems via neural template. In *WWW Demo*.

Lei Li, Yongfeng Zhang, and Li Chen. 2020b. Generate neural template explanations for recommendation. In *CIKM*, pages 755–764.

Lei Li, Yongfeng Zhang, and Li Chen. 2021b. Personalized transformer for explainable recommendation. In *ACL*, pages 4947–4957.

Lei Li, Yongfeng Zhang, and Li Chen. 2022. Personalized prompt learning for explainable recommendation. *arXiv*, abs/2202.07371.

Lei Li, Yongfeng Zhang, and Li Chen. 2023a. Personalized prompt learning for explainable recommendation. *ACM Trans. Inf. Syst.*, 41(4):103:1–103:26.

Lei Li, Yongfeng Zhang, and Li Chen. 2023b. Prompt distillation for efficient llm-based recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, pages 1348–1357. ACM.

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.

Donghua Liu, Jing Li, Bo Du, Jun Chang, and Rong Gao. 2019. DAML: dual attention mutual learning between ratings and reviews for item recommendation. In *KDD*, pages 344–352.

Hai Liu, Chao Zheng, Duantengchuan Li, Xiaoxuan Shen, Ke Lin, Jiazhang Wang, Zhen Zhang, Zhaoli Zhang, and Neal N. Xiong. 2022. EDMF: efficient deep matrix factorization with review feature learning for industrial recommender system. *IEEE Transactions on Industrial Informatics*, 18(7):4361–4371.

Zhuang Liu, Yunpu Ma, Matthias Schubert, Yuanxin Ouyang, Wenge Rong, and Zhang Xiong. 2023. Multimodal contrastive transformer for explainable recommendation. *IEEE Transactions on Computational Social Systems*.

Renze Lou, Kai Zhang, and Wenpeng Yin. 2023. Is prompt all you need? no. A comprehensive and broader view of instruction learning. *arXiv*, abs/2303.10475.

Jianmo Ni, Jiacheng Li, and Julian J. McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *EMNLP*, pages 188–197.

Sicheng Pan, Dongsheng Li, Hansu Gu, Tun Lu, Xufang Luo, and Ning Gu. 2022. Accurate and explainable recommendation via review rationalization. In *WWW*, pages 3092–3101.

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

Pengpeng Qiao, Zhiwei Zhang, Zhetao Li, Yuanxing Zhang, Kaigui Bian, Yanzhou Li, and Guoren Wang. 2022. Tag: Joint triple-hierarchical attention and gcn for review-based social recommender system. *IEEE Transactions on Knowledge and Data Engineering*.

Chen Shengyuan, Yunfeng Cai, Huang Fang, Xiao Huang, and Mingming Sun. 2024. Differentiable neuro-symbolic reasoning on large-scale knowledge graphs. *Advances in Neural Information Processing Systems*, 36.

Jie Shuai, Kun Zhang, Le Wu, Peijie Sun, Richang Hong, Meng Wang, and Yong Li. 2022. A review-aware graph contrastive learning framework for recommendation. In *SIGIR*, pages 1283–1293.

N TISHBY. 2000. The information bottleneck method. In *Proceedings of the 37-thAnnual Allerton Conference on Communication, 2000*.

Haoqin Tu and Yitong Li. 2022. An overview on controllable text generation via variational auto-encoders. *CoRR*, abs/2211.07954.

Haoqin Tu, Zhongliang Yang, Jinshuai Yang, Siyu Zhang, and Yongfeng Huang. 2022. Adavae: Exploring adaptive gpt-2s in variational auto-encoders for language modeling. *CoRR*, abs/2205.05862.

Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2018a. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *CIKM*, pages 417–426.

Linlin Wang, Zefeng Cai, Gerard de Melo, Zhu Cao, and Liang He. 2023. Disentangled cvaes with contrastive learning for explainable recommendation. In *AAAI*, pages 13691–13699.

Peng Wang, Renqin Cai, and Hongning Wang. 2022a. Graph-based extractive explainer for recommendations. In *WWW*, pages 2163–2171.

Shendi Wang, Haoyang Li, Caleb Chen Cao, Xiao-Hui Li, Ng Ngai Fai, Jianxin Liu, Xun Xue, Hu Song, Jinyu Li, Guangye Gu, and Lei Chen. 2022b. Tower bridge net (tb-net): Bidirectional knowledge graph aware embedding propagation for explainable recommender systems. In *ICDE*, pages 3268–3279.

Xiting Wang, Yiru Chen, Jie Yang, Le Wu, Zhengtao Wu, and Xing Xie. 2018b. A reinforcement learning framework for explainable recommendation. In *ICDM*, pages 587–596.

Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019. NPA: neural news recommendation with personalized attention. In *KDD*, pages 2576–2584.

Xuansheng Wu, Haiyan Zhao, Yaochen Zhu, Yucheng Shi, Fan Yang, Tianming Liu, Xiaoming Zhai, Wenlin Yao, Jundong Li, Mengnan Du, et al. 2024. Usable xai: 10 strategies towards exploiting explainability in the llm era. *arXiv preprint arXiv:2403.08946*.

Zhouhang Xie, Sameer Singh, Julian J. McAuley, and Bodhisattwa Prasad Majumder. 2023. Factual and informative review generation for explainable recommendation. In *AAAI*, pages 13816–13824.

Aobo Yang, Nan Wang, Renqin Cai, Hongbo Deng, and Hongning Wang. 2022. Comparative explanations of recommendations. In *WWW*, pages 3113–3123.

Aobo Yang, Nan Wang, Hongbo Deng, and Hongning Wang. 2021. Explanation as a defense of recommendation. In *WSDM*, pages 1029–1037.

Qinggang Zhang, Junnan Dong, Hao Chen, Xiao Huang, Daochen Zha, and Zailiang Yu. 2023. Knowgpt: Black-box knowledge injection for large language models. *arXiv preprint arXiv:2312.06185*.

Qinggang Zhang, Junnan Dong, Hao Chen, Wentao Li, Feiran Huang, and Xiao Huang. 2024. Structure guided large language model for sql generation. *arXiv preprint arXiv:2402.13284*.

Wei Zhang, Junbing Yan, Zhuo Wang, and Jianyong Wang. 2022. Neuro-symbolic interpretable collaborative filtering for attribute-based recommendation. In *WWW*, pages 3229–3238.

Jihua Zhu, Yujiao He, Guoshuai Zhao, Xuxiao Bu, and Xueming Qian. 2023. Joint reason generation and rating prediction for explainable recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(5):4940–4953.

# A Appendix

## A.1 Notations

In this work, matrices are symbolized by uppercase bold letters (e.g., $\mathbf{E}$), vectors are represented by lowercase bold letters (e.g., $\mathbf{e}$), and scalars are represented by lowercase letters (e.g., $r_{ij}$). We denote sets with calligraphic letters (e.g., $\mathcal{U}$) and denote their size using cardinality (e.g., $|\mathcal{U}|$). The $i^{th}$ row of matrix $\mathbf{E}$ is denoted as $e_i$, whereas the element locating in the $i^{th}$ row and $j^{th}$ column is denoted as $\mathbf{E}_{ij}$.

## A.2 Dataset Details

These datasets consist of customer reviews and ratings specifically related to items. We regard the sentence in the review that can explain the purchase motivation as the explanation following the previous practice (Li et al., 2020b; Ni et al., 2019). We extract explanations and item features (Yang et al., 2021) from the given reviews using ChatGPT. Both of them will be included in the evaluation metrics, which will be introduced later. To ensure that users and items under evaluation receive training, we reserve users and items that have at least one record in the training set (Li et al., 2021b). The interactions are initially sorted based on their timestamps and then divided into training, validation, and test sets with a proportion of $8 : 1 : 1$. The rating score range in three datasets based on the sentiment from negative to positive is $\{1, 2, 3, 4, 5\}$.

## A.3 Evaluation Metrics

Here, the symbol ↑ followed by the metric means that the higher the score, the better the performance, while ↓ represents the lower the score, the better the performance. In the first group, we view the explanations as purely plain text and evaluate the text quality. (*i*) We adopt the widely used BLEU-1↑ and BLEU-4↑ metrics to measure the word-level relevance of the generated explanations to the ground-truth explanations. (*ii*) User and item serve as the soft prompt to generate explanations, with their representations containing textual information. We adopt two classic metrics, MAE↓ and RMSE↓ to evaluate the predicted ratings using user and item representations with Eq. (10) (Li et al., 2017, 2020b, 2021b).

In the second group, we examine the interpretability quality of generated explanations from two aspects. (*i*) To measure the degree of personalization, we adopt Unique Sentence Ratio (USR)↑, Feature Coverage Ratio (FCR)↑, and Feature Matching Ratio (FMR)↑ by following previous efforts (Li et al., 2020b, 2021b). Specifically, USR assesses the diversity of generated interpretations, while FCR and FMR measure the diversity and accuracy of generated features. (*ii*) To examine the semantic-level relevance, we adopt `Entail`↑ (Xie et al., 2023) to estimate the extent of whether the generated explanation entails the label explanation. To measure the emotional consistency between the predicted rating and the generated explanations, we adopt `Consistency`↑ (Wang et al., 2018b).

Consistent with prior practices (Xie et al., 2023), we employ a pre-trained entailment model to assess whether the generated explanations entail the ground truths. Similarly, following previous work (Wang et al., 2018b), we employ a sentiment classification model to estimate sentiment scores based on the generated explanations and calculate the correlation with the predicted rating.