Explainable Recommendation with Topic-enhanced Personalized Prompt Learning

1st Rui Tang
State Key Laboratory of Media Convergence and Communication
Communication University of China
Beijing, China
aria@cuc.edu.cn

2nd Cheng Yang*
State Key Laboratory of Media Convergence and Communication
Communication University of China
Beijing, China
chy@cuc.edu.cn

3rd Peng Yi
State Key Laboratory of Media Convergence and Communication
Communication University of China
Beijing, China
yipeng@cuc.edu.cn

***Abstract*—Explainable recommendation systems aim to provide both recommendation results and corresponding explanations to enhance user trust and engagement. Existing post-hoc methods leverage pre-trained language models to generate textual explanations. However, these works hardly consider the latent topical structures in user reviews, resulting in generic and repetitive explanations with limited personalization. To address this issue, this paper introduces TPLER (Topic-enhanced Prompt Learning for Explainable Recommendation), a novel model that integrates topical preferences into prompt learning to generate more personalized and higher-quality explanations. Our approach employs a topic-enhanced feature extraction strategy based on Latent Dirichlet Allocation (LDA) to explicitly model user and item topical preferences from review texts. These topic-aware features are fed into a pre-trained transformer model via continuous and discrete prompts, guiding the generation process. Continuous prompts encode rating information to capture personalized user-item interactions, while discrete prompts are obtained from topic-enhanced features. A two-stage multi-task training strategy is proposed to jointly optimize the rating prediction and explanation generation tasks. Experiments on three benchmark datasets show that TPLER achieves an average improvement of 7.93% across text quality metrics (BLEU-1/4, ROUGE-1/2/L) and 21.63% for personalization metrics, verifying the effectiveness of our proposed TPLER model.**

***Index Terms*—explainable recommendation, prompt learning, pre-trained language model, topic model**

I. Introduction

Recommendation systems have developed into an essential part for various online applications, including e-commerce platforms [1], search engines [2], and interactive services [3]. However, conventional recommendation systems prioritize providing users with more accurate recommendation results. The lack of explanation for why these recommendations are made undermines user's trust and deprives developers of effective guidance for system optimization [4].

Explainable recommendation systems aim to present recommendation results while providing corresponding explanations. As a research area that has been attracting sustained attention, there are two main lines of existing work on explainable recommendation: embedded and post-hoc methods. Embedded methods enable explainability of recommendation models by revealing the decision-making process behind recommendation results. They represent user-item interactions with external knowledge, such as Tag-based Graphs [5], Knowledge Graphs [6] and Causal Graphs [7], with the optimization objective being to improve rating prediction performance. However, these methods rely on structured knowledge for model construction. The difficulty in obtaining high-quality datasets compromises both the accuracy and explainability of recommendation results. Post-hoc methods [8] separate the explainable recommendation task into a rating prediction task and an explanation generation task. Multi-task learning facilitates the joint optimization of both tasks, which utilizes shared latent personalized representations but employs individual models to output the final rating prediction and explanation generation [9]. Due to the abundance and accessibility of textual reviews online and the remarkable performance of pre-trained language models in text generation tasks, a predominant research of post-hoc explainable recommendation focus on the generation of personalized textual explanations in recent years. How to improve the personalization and quality of explanations is the key issue. For example, NRT [10] and Att2Seq [11] employed shared latent user and item ID embeddings to enhance personalization, while utilizing Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM) architectures to guarantee the quality of the generated text. Since a user might not be interested in all features of an item, PETER [12] and ERRA [13] integrated important features extracted from reviews that reflect user preferences into shared personalized representations and generated text based on Transformer. PEPLER [14] utilized the Generative Pre-Training (GPT) series with stronger language modeling capabilities as its generation model.

Despite the achieved progress, these methods still suffer from inappropriate feature extraction strategy, limiting the personalization of the generated text. As shown in Fig. 1, textual reviews exhibit a latent topical structure [15]. Users primarily describe a hotel through several topics, represented by different colors in the figure, such as "Price", "Location", "Room" and "Food". When extracting user and item features from reviews, existing methods like PETER [12] and PE-

This work was supported in part by a grant from the Fundamental Research Funds for the Central Universities (NO.CUC24BS28).

*Corresponding author.

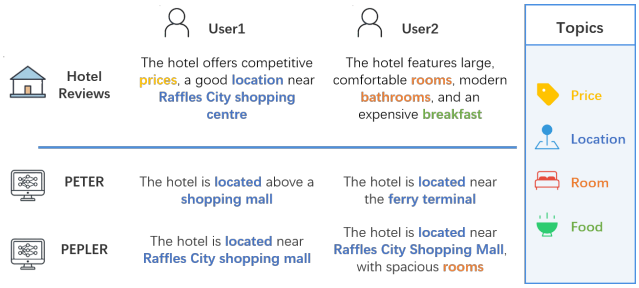


Fig. 1: Examples of different users' reviews for the same hotel and explanations generated by PETER and PEPLER.

PLER [14], do not consider the underlying topic distribution of the text, resulting in generated explanations that mostly consist of repetitive sentences with general features and same topics. For example, Fig. 1 demonstrates that the "Location" topic becomes the most frequently occurring topic in the generated sentences. Topic models, such as Latent Dirichlet Allocation (LDA) [16], are widely used in review-based recommender to improve rating prediction accuracy [15] [17]. The learned topic distributions are often used to enrich the user and item representations [18]. How to design a feature extraction strategy that integrates topic information to enhance the personalization and quality of generated explanations is our key focus.

To this end, we propose to adjust the existing pre-trained language models (GPT-2) for explainable recommendation. By designing a topic-enhanced feature extraction strategy, user and item ID embeddings, combined with topic-enhanced features, are incorporated as prompts to guide the generation of more personalized and higher-quality explanations.

The main contributions are as follows:

- By analyzing user-item interaction reviews, we design a topic-enhanced feature extraction strategy based on the LDA model to explicitly capture user and item preferences. This alleviates the problem of insufficient personalization in explainable recommendation models caused by the lack of topic information mining.
- We propose the model, denoted as TPLER (Topic-enhanced Prompt Learning for Explainable Recommendation). A topic-enhanced prompt learning module is designed. Ratings are incorporated into the continuous prompts to capture personalized features, while a topic-enhanced strategy is applied to the discrete prompts to model topic preferences.
- Extensive experiments conducted on three benchmark datasets demonstrate the superiority of our TPLER model over various state-of-the-art baselines, showcasing its effectiveness in improving recommendation performance.

II. Related Work

A. Explainable Recommendation

The relevant research on explainable recommendation can be classified into two types: embedded and post-hoc methods [4]. Since embedded methods rely on structured knowledge for model construction, the explainable recommendation model proposed in this paper employs the post-hoc method. Recommendation explanations take various forms, such as aspect-based [19], textual [20] [21], and visual explanations [22]. Due to the widespread availability of textual information on the internet, such as reviews, posts, and product descriptions, text has become the optimal carrier for explanations. Relevant approaches can be classified into text extraction-based models [20] and text generation-based models [23]. Advances in deep learning have spurred significant interest in generation-based explainable recommendation. While recurrent neural networks like GRU [10] and LSTM [11], as well as Transformers [12] [13], have been widely utilized to generate personalized explanations, the adaptation of advanced pre-trained language models for personalization tasks is still in early stages.

B. Pre-trained Language Models and Prompt Learning

Pre-trained language models (PLMs) [24] have demonstrated exceptional proficiency in a wide range of natural language understanding and generation applications, such as question answering systems [25] and conversational systems [3]. For example, LLM2ER-EQR [26] adapted InstructGPT [27] to generate explanations in a pretraining-finetuning paradigm. Since retraining a large transformer model demands substantial computational resources, prompt learning [28] has become as a new paradigm. Studies have shown that the potential of PLMs can be elicited through textual prompts or in-context demonstrations, particularly in data-scarce environments. PEPLER [14] treats the user and item ID embeddings as continuous prompts and the extracted feature words as discrete prompts. These two types of prompts are fed into GPT-2 to generate personalized explanations. However, existing prompts fail to account for the latent topic structure of textual reviews, resulting in generated explanations with limited personalization and suboptimal quality.

C. Review-based Rating Prediction

Rating prediction methods for recommendation can typically be categorized into three types: collaborative filtering (CF), content-based filtering (CBF), and hybrid methods [29]. CF leverages observed rating matrix between users and items to make predictions. CBF employs auxiliary information to alleviate the rating matrix sparsity problems in CF. Reviews, as a common auxiliary information, can be utilized to extract semantic features [15] [30]. Most exsiting methods derived from LDA and treat a corpus as a mixture of topics (aspects) [18]. TopicMF [31] employed the learned topic distributions as model regularization to enhance the accuracy of rating prediction. TopicVAE [32] integrated attention-based topic extraction and variational inference to achieve fine-grained disentangled representation learning. However, researchers have rarely focused on leveraging topic information to simultaneously optimize the accuracy of rating predictions and the quality of generated explanations.

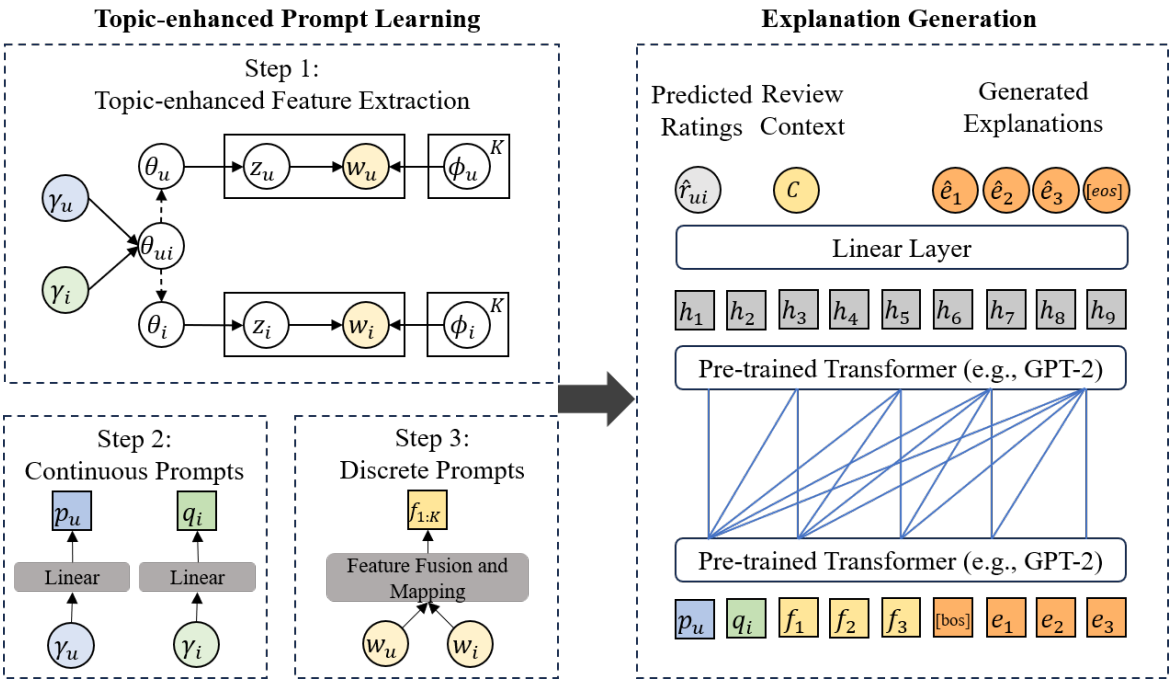


Fig. 2: The overall framework of our proposed TPLER Model.

III. Problem Formulation

Given a user set U, an item set I, a user-item raing matrix R, and a user-item review text corpus C, the objective of topic-enhanced explainable recommendation is to predict the rating $\hat{r}_{ui}$ of a user $u \in U$ for an item $i \in I$, while also generating a fluent, personalized, and semantically topic-aware textual explanation $e_{ui} = \{\hat{e}_1, \hat{e}_2, ...\hat{e}_n\}$ to clarify the rationale behind the recommendation.

IV. Methodology

A. Overall Framework

The overall framework of our proposed TPLER model is shown in Fig. 2, mainly consists of two components: topic-enhanced prompt learning and explanation generation. The topic-enhanced prompt learning component first extracts personalized topic features by linking user and item ID embeddings with their associated topic information. Subsequently, the obtained implicit ID embeddings and explicit topic features are adapted to the pre-trained transformer via continuous prompt and discrete prompt, respectively. Finally, a two-stage multi-task joint optimization strategy is employed to achieve topic-enhanced personalized recommendation explanation generation.

B. Topic-enhanced Prompt Learning

Given the review words set as $V = \{w_1, ..., w_v, ...w_{|V|}\}$, where $|V|$ is the total number of words. The review of user u is denoted as x_u, which can be composed of a subset of the above word set. z_u represents the topic assigned to each word in x_u. Therefore, the user review X_U can be represented as a matrix $X_U = [x_{uv}]_{UV}$. Similarly, for all items, the review of item i is denoted asx_i, which also can be composed of a subset of the word set V. z_i represents the topic assigned to each word in x_i. Thus, the item review content X_I can be represented as a matrix $X_I = [x_{iv}]_{IV}$. Assume that the review content of both users and items shares a common latent topic space, with the number of topics being K.

When users comment on items, their reviews tend to focus more on describing the attributes of the items themselves rather than expressing personal preference tendencies. When users provide ratings, the scores they give reflect their degree of preference for various attributes of the items. Based on the above observations, the latent topic parameter $\theta \in \mathbb{R}^{IU \times K}$ is defined to represent how each topic is emphasized in users' comment on items. The topic-word matrices for the user and item domains are defined as $\phi_u \in \mathbb{R}^{V \times K}$ and $\phi_i \in \mathbb{R}^{V \times K}$, respectively. Let $\gamma_u \in \mathbb{R}^{U \times K}$ and $\gamma_i \in \mathbb{R}^{I \times K}$ denote the user and item ID embeddings learned from rating information.

a) Topic-enhanced Feature Extraction: To associate the ID embeddings γ_u, γ_i and θ, intuitively, γ_i represents the attributes of an item, and users decide whether to give a high rating based on whether they like these attributes according to their personal preferences γ_u. On the other hand, the topics θ, ϕ_u, and ϕ_i define specific distributions of words appearing in reviews. The objective of the transformation function is as follows: if a user or item exhibits a particular rating preference attribute (i.e., a high value of γ_{uk} or γ_{ik}), it should correspond to a specific topic described in the reviews (i.e., high values of θ_{uk} and θ_{ik}).

Therefore, inspired by the assumptions in TopicMF [31], the transformation function in the topic-enhanced feature extraction strategy proposed in this paper should satisfy the

following conditions: (1) γ_{uk} and γ_{ik} can take any real values, while $\sum_{k\in K}\theta_{iuk}=1$. (2) The transformation function should be monotonic, ensuring that the maximum value of γ_i corresponds to the maximum value of θ_i, and similarly for γ_u and θ_u. (3) The contributions of users and items to the topic θ are different. Specifically, the transformation function is formulated as:

$$\begin{aligned}\theta_{iuk}=\alpha\frac{exp(\sigma_1\gamma_{ik})}{\sum_{k'=1}^{K}exp(\sigma_1\gamma_{ik'})}\\+(1-\alpha)\frac{exp(\sigma_2\gamma_{uk})}{\sum_{k'=1}^{K}exp(\sigma_2\gamma_{uk'})}\end{aligned}\tag{1}$$

As shown in (1), the parameter α controls the contribution of item ratings and user ratings to the topic θ. The parameters σ_1 and σ_2 control the shape of the softmax function. When $\sigma_1\to 1$, users tend to focus more on the core topics of the target item; when $\sigma_1\to 0$, users tend to discuss all topics of the target item uniformly. Similarly, when $\sigma_2\to 1$, users always tend to discuss the important topics they care about; when $\sigma_2\to 0$, users are willing to discuss all topics they are concerned about in their reviews.

After obtaining the review topic distribution θ_{iuk}, the user topic-word distribution ϕ_u, and the item topic-word distribution ϕ_i, the probability of generating a word from a review is as follows:

$$p(w^{(k)}|x_u)=p(w|k)p(k|x_u)=\sum_{i=1}^{I}\theta_{iuk}\phi_{uk}\tag{2}$$

$$p(w^{(k)}|x_i)=p(w|k)p(k|x_i)=\sum_{u=1}^{U}\theta_{iuk}\phi_{ik}\tag{3}$$

Therefore, for each topic $k\in K$, the words are sorted in descending order according to $p(w|x_u)$ and $p(w|x_i)$. The word with the highest probability is selected as the feature for user u and item i under topic k, as shown below:

$$w_{uk}=\arg\max_{w\in V}p(w^{(k)}|x_u)\tag{4}$$

$$w_{ik}=\arg\max_{w\in V}p(w^{(k)}|x_i)\tag{5}$$

b) Continuous Prompts: Continuous Prompts are composed of vector sequences that cannot be directly understood by humans. For user and item ID embeddings, if they were all treated as unknown tokens and added to the pre-trained model, the model would face high computational costs when predicting explanation text from a vast number of unknown tokens. Therefore, similar to the handling of continuous prompts in [14], we treat user and item ID embeddings as two additional token types. Since the embeddings and the GPT-2 token embeddings have different dimensions, we use a linear projection to conform the dimension of γ_u and γ_i to that of the GPT-2 token embeddings. The continuous prompt vectors p_u for user u and q_i for item i can be obtained by the following equations:

$$p_u=Linear(\gamma_u)g_u(u)\tag{6}$$

$$q_i=Linear(\gamma_i)g_i(i)\tag{7}$$

Where $g_u(u)\in\{0,1\}^{|U|}$, $g_i(i)\in\{0,1\}^{|I|}$. Non-zero elements represent the position of u and i. The values of the embeddings γ_u and γ_i are randomly initialized and updated during training through backpropagation.

c) Discrete Prompts: The token representations of word-based discrete prompts f could be obtained by mapping each word to its corresponding vector via token embedding operation in GPT-2. Furthermore, to derive the K feature embeddings that capture the aspects a user emphasizes in reviews, we integrate the user and item features via a linear fusion function. This step consolidates the number of discrete prompts from $2K$ to K.

$$f=Linear([Emb(w_u);Emb(w_i)])\tag{8}$$

Where Emb is the word embedding layer of GPT-2.

C. Explanation Generation

Our proposed TPLER model is built upon a pre-trained Transformer backbone (GPT-2). The input sequence is denoted as $S_{ui}=[p_u,q_i,f_1,...,f_K,e_1,...,e_{|E_{ui}|}]$, where p_u and q_i are continuous prompts obtained from the user and item ID embeddings. $f_{1:K}$ are discrete prompts containing topic-enhanced personalized features. $e_{1:|E_{ui}|}$ denote the token representations of the explanation text sequence. K is the number of topics, and $|E_{ui}|$ denotes the number of explanation tokens. Then the positional encoding representation $[p_1,...,p_{|S_{ui}|}]$ is added to the token representation S_{ui}, forming the input sequence to the pre-trained language model. This input sequence is passed into the pre-trained language model, producing an output sequence representation denoted as $h=[h_1,...,h_{|S_{ui}|}]$. Finally, each token in h is projected through a linear matrix into a word probability distribution, as shown in the following equation:

$$c_t=Softmax(W^v h_t+b^v)\tag{9}$$

Where c_t represents the probability distribution over vocabulary V. $W^v\in\mathbb{R}^{|V|\times len_t}$ and $b^v\in\mathbb{R}^{|V|}$ are weight parameters. len_t denotes the dimension of the token embeddings.

The TPLER model generates the explanation text auto-regressively word by word, employing a greedy decoding strategy during inference stage. We utilize the Negative Log-Likelihood Loss (NLL) as the loss function for text generation, which ensures the similarity between generated words and ground-truth words.

$$\mathcal{L}_G=\frac{1}{|\mathcal{T}|}\sum_{(u,i)\in\mathcal{T}}\frac{1}{|E_{ui}|}\sum_{t=1}^{|E_{ui}|}-logc_{2+K+t}^{e_t}\tag{10}$$

$\mathcal{T}$ denotes the training set. Unlike the generation loss of existing models, the input contains two continuous prompts and K discrete prompts, resulting in a position shift of $2+K$ for the output distribution $c_t^{e_t}$.

For the rating prediction task, TPLER employs Matrix Factorization (MF) as the backbone model to enhance the personalized representation capability of continuous prompts.

The predicted rating is obtained from the dot product of the continuous prompts of user u and item i, as shown in the following equation:

$$\hat{r}_{ui} = p_u q_i^\top \tag{11}$$

The rating prediction loss $\mathcal{L}_R$ is given by:

$$\mathcal{L}_R = \frac{1}{|\mathcal{T}|} \sum_{(u,i)\in\mathcal{T}} (\hat{r}_{ui} - r_{ui})^2 \tag{12}$$

Additionally, based on (1), TPLER establishes the association between ID embeddings and topic features by fitting the review context. The loss function for the review corpus, denoted as $\mathcal{L}_C$, is formulated as follows:

$$\begin{aligned}\mathcal{L}_C &= (\theta\phi_u^\top - X_U)^2 + (\theta\phi_i^\top - X_I)^2 \\ &= \sum_{u=1}^{U}\sum_{v=1}^{V}\left(\sum_{i=1}^{I}\theta_{iu}\phi_u^\top - x_{uv}\right)^2 \\ &\quad + \sum_{i=1}^{I}\sum_{v=1}^{V}\left(\sum_{u=1}^{U}\theta_{iu}\phi_i^\top - x_{iv}\right)^2\end{aligned} \tag{13}$$

D. Two-Stage Multi-Task Training

The objective of our proposed model is to jointly optimize both the recommendation task and the explanation generation task. These two tasks are linked via personalized prompts, based on rating-related parameters $\Theta = \{\gamma_u, \gamma_i\}$ and topic-related parameters $\Phi = \{\theta, \phi_u, \phi_i\}$. Therefore, the loss functions from (10), (12), and (13) are integrated into a multi-task learning problem, formulated as follows:

$$\mathcal{L} = \mathcal{L}_G + \lambda_R\mathcal{L}_R + \lambda_C\mathcal{L}_C \tag{14}$$

Where λ_C and λ_R are tradef-off parameters.

Typically, the parameters in the generation model (10) and the recommendation model (11) (12) are fitted via gradient descent, while the parameters in the topic model (13) are estimated using Gibbs sampling. Since the TPLER model integrates both types, we designs an alternating training strategy to optimize these two sets of parameters. The procedure consists of the following two stages:

$$\Theta^{(t)}, \Phi^{(t)} = \arg\min_{\Theta,\Phi} \mathcal{L} \tag{15}$$

$$p(z^{(t)} = k) = \theta_k \phi_{k,v}^{(t)} \tag{16}$$

In the first stage (15), the topic assignment z (a latent variable) for each word is fixed. Then, the remaining parameters are fitted via gradient descent.

In the second stage (16), we iterate on the words in the user and item review corpora to update their topic assignments. Specifically, each word is randomly assigned to a topic $k = 1...K$, with a probability proportional to the likelihood of that topic generating the word. This procedure is similar to the LDA. The main difference that in TPLER the topic representation θ does not follow a Dirichlet distribution but is determined by the parameters in $\Theta^{(t)}$ obtained from the previous stage via (1). Consequently, in the current stage, only a single pass through the corpus is required to update the topic assignments z. That is, after updating $\Theta^{(t)}$ according to (15), we sample the new topic assignments z.

Finally, the two stages are alternated, updating the parameters until the value of the loss function $\mathcal{L}$ converges.

TABLE I: Detailed Statistics of the Datasets

Datasets	Amazon	TripAdvisor	Yelp
Number of Users	7,506	9,765	27,147
Number of Items	7,360	6,280	20,266
Number of Reviews	441,783	320,023	1,293,247
Number of Features	5,399	5,069	7,340
Avg. Reviews per User	58.86	32.77	47.64
Avg. Reviews per Item	60.02	50.96	63.81
Avg. Words per Review	14.14	13.01	12.32

V. Experiment

A. Experimental Setting

a) Datasets: This paper employs three widely-used benchmark datasets [14] in recommender systems for the experiments: TripAdvisor[1] (hotels), Amazon[2] (Movies & TV), and Yelp [3] (restaurants). Detailed statistical information provided is shown in Table I. The datasets are split into training, validation, and test sets with ratio 8:1:1.

b) Baseline Methods: For the explanation generation task, to evaluate the explainability of the model, we conduct comparisons with text generation-based models.

- NRT [10] Based on predicted ratings and word distributions in reviews, GRUs are used to generate explanations word by word.
- CAML [9] The most relevant reviews and words are selected through a mutual attention mechanism to generate recommendation explanations.
- PETER [12] ID embeddings and feature embeddings are integrated into the Transformer, which serves as the initial vector to generate recommendation explanations.
- PEPLER [14] Recommendation explanations are generated using pre-trained language models (GPT-2) based on prompt learning.

For the rating prediction task, the model's accuracy is primarily compared with traditional matrix factorization-based recommendation models and deep learning-based recommendation models. All comparative models utilize reviews as auxiliary information to enhance rating prediction performance.

- TopicMF [31] A topic-enhanced matrix factorization rating prediction model which considers the ratings and accompanied review texts.
- Deep learning-based models, like NRT [10], CAML [9], and PETER [12], are based on multi-task learning to simultaneously optimize both rating and explanation tasks.

[1]https://www.tripadvisor.com
[2]https://jmcauley.ucsd.edu/data/amazon
[3]https://www.yelp.com/dataset

TABLE II: Explanation generation performance comparison

Dataset	Model	Personalization				Text Quality				
		FMR ↑	FCR ↑	DIV ↓	USR ↑	B1 ↑	B4 ↑	R1 ↑	R2 ↑	RL ↑
Amazon	NRT	0.08	0.12	2.98	0.13	10.94	0.74	11.93	1.06	9.17
	CAML	0.06	0.09	3.09	0.21	10.21	0.73	11.50	0.92	8.92
	PETER	0.08	0.09	2.87	0.20	10.78	0.79	12.90	1.56	9.77
	PEPLER	0.11	0.19	**2.45**	0.37	12.12	0.82	13.10	1.65	9.94
	TPLER	**0.12**	**0.24**	2.58	**0.46**	**12.66**	**0.94**	**13.93**	**1.92**	**10.47**
TripAdvisor	NRT	0.05	0.10	4.02	0.09	13.07	0.75	13.56	1.70	10.25
	CAML	0.05	0.13	3.90	0.12	13.28	0.76	14.25	1.59	10.56
	PETER	**0.07**	0.18	3.57	0.16	14.33	0.89	14.02	1.94	11.61
	PEPLER	**0.07**	**0.19**	2.86	0.18	14.32	1.09	14.13	2.03	12.14
	TPLER	**0.07**	**0.19**	**2.76**	**0.24**	**14.59**	**1.10**	**14.35**	**2.10**	**12.46**
Yelp	NRT	0.05	0.09	2.31	0.08	9.50	0.52	9.89	1.10	8.35
	CAML	0.06	0.08	2.95	0.07	9.25	0.45	10.02	1.07	8.23
	PETER	0.07	0.10	2.68	0.08	10.09	0.49	10.34	1.00	8.46
	PEPLER	**0.08**	0.11	2.18	0.09	10.11	0.54	10.81	1.06	9.06
	TPLER	**0.08**	**0.23**	**2.10**	**0.14**	**11.39**	**0.68**	**11.32**	**1.15**	**9.93**

The best performance on each dataset is highlighted in bold. B1 and B4 represent BLEU-1 and BLEU-4. R1, R2 and RL represent recall score for ROUGE-1, ROUGE-2 and F1 score for ROUGE-L. Arrows indicate the direction of better performance (↑: higher is better; ↓: lower is better).

c) Evaluation Metrics: We adopt two common metrics, BLEU [33] and ROUGE [34], to evaluate explanation quality. Precision is measured by BLEU-1 and BLEU-4, recall by ROUGE-1 and ROUGE-2, and the F1-score of ROUGE-L provides an overall measure. Following PEPLER [14], we also measure explanation personalization using the Feature Matching Ratio (FMR), Feature Coverage Ratio (FCR), Feature Diversity (DIV), and Unique Sentence Ratio (USR). We also adopt Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) to evaluate the rating prediction performance.

d) Implementation Details: Our experiments are conducted on an NVIDIA GeForce RTX 4090 GPU. The model utilizes pre-trained GPT-2 [35] from huggingface as its backbone network. GPT-2 employs Byte Pair Encoding (BPE) for vocabulary construction. The embedding dimension in GPT-2 is 768. The TPLER model is optimized using AdamW, with the batch size set to 128 and the learning rate set to 0.001. For the proposed model, the contribution parameter α is set to 0.5. The balancing parameters λ_C and λ_R are set to 1 and 0.1. The latent topics are initialized to $K = 10$.

B. Experimental Results and Analysis

The experiments mainly aim at addressing the following research questions:

RQ 1: Does the proposed TPLER model outperform the baseline models on both the explanation generation task and the rating prediction task?

RQ 2: Does the topic-enhanced feature extraction strategy contribute to the two tasks? Does multi-task learning of the rating prediction task and the explanation generation task benefit the performance of both tasks?

RQ 3: How does the number of topics affect the model's performance?

RQ 4: Can topic-enhanced prompts generate explanations that better adhere to the topics of ground-truth texts?

a) Performance Comparison (RQ1): The performance comparison of explanation generation results on different datasets are shown in Table II. The proposed TPLER achieves an average improvement of 7.93% across text quality metrics (BLEU-1/4, ROUGE-1/2/L) and 21.63% across personalization metrics over the best baseline, PEPLER. In terms of text quality, TPLER consistently achieves higher scores on BLEU and ROUGE metrics. Improvements on BLEU-4 are more pronounced than BLEU-1, especially on the Yelp dataset, where B4 improves by 25.9%. This indicates better fluency and higher-order n-gram matching in the generated explanations. For personalization, TPLER shows gains in Feature Coverage Ratio (FCR) and Unique Sentence Ratio (USR), demonstrating a superior ability to incorporate diverse features and generate unique explanations. The experimental results demonstrate the effectiveness of the proposed multi-task training strategy and the topic-enhanced personalized prompts for the explanation generation task, leading to explanations that are more personalized and higher in quality.

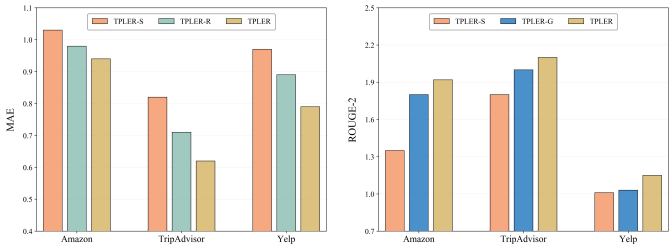
As shown in Table III, the proposed TPLER model demonstrates superiority in rating prediction, achieving the best performance on three datasets for both RMSE and MAE. From the overall results, deep learning-based models generally achieve higher accuracy than traditional matrix factorization-based models. This is primarily because deep learning models, with their more complex network architectures, capture the feature representations of users and items more effectively. Compared to the PETER model, the TPLER model reduces RMSE by approximately 8.56% and MAE by approximately 9.19% on average in the rating prediction task, validating the effectiveness of its topic-enhanced prompts and multi-task training strategy in enhancing rating prediction accuracy.

b) Ablation Study (RQ2): This section conducts ablation study to analyze the effectiveness of key modules in the model, including topic-enhanced feature extraction and multi-task learning strategy. Fig. 3 shows the results for the MAE

TABLE III: Rating prediction performance comparison

Dataset	Model	RMSE ↓	MAE ↓
Amazon	TopicMF	1.31	1.12
	NRT	1.23	1.07
	CAML	1.24	1.06
	PETER	1.21	1.01
	TPLER	**1.19**	**0.94**
TripAdvisor	TopicMF	1.17	0.88
	NRT	0.91	0.73
	CAML	0.94	0.69
	PETER	0.92	0.70
	TPLER	**0.81**	**0.62**
Yelp	TopicMF	1.57	0.96
	NRT	1.19	0.92
	CAML	1.18	0.89
	PETER	1.16	0.87
	TPLER	**1.02**	**0.79**

The best performance on each dataset is highlighted in bold. ↓ indicates lower values are better.



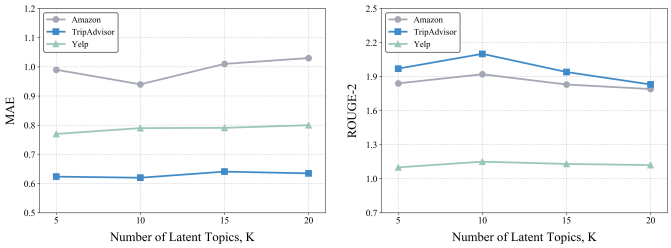
(a) Rating prediction task (b) Explanation generation task

Fig. 3: Ablation study

and ROUGE-2 metrics, respectively.

Three variant models TPLER-S, TPLER-R and TPLER-G are designed in our experiments. Compared with TPLER, TPLER-S removes the association between continuous prompts and discrete prompts; meanwhile, the review corpus loss $\mathcal{L}_C$ is also removed from (14). The TPLER-R model only considers the rating prediction task by retaining the loss $\mathcal{L}_R$. The TPLER-G model retains only the explanation generation loss $\mathcal{L}_G$. As shown in Fig. 3a and 3b, the TPLER multi-task learning model outperforms both single-task models $\mathcal{L}_R$ and $\mathcal{L}_G$ across all datasets. Furthermore, The results show that TPLER-S performs worse than TPLER and the single-task variants TPLER-R and TPLER-G in terms of both recommendation accuracy and explainability. This finding clearly verifies that there exists a correlation between the rating prediction task and the explanation generation task. The improvement in model performance is attributed to the topic-enhanced feature extraction strategy, which effectively captures the intrinsic correlation between the two tasks, rather than merely incorporating auxiliary features.

c) Hyper-parameter Sensitivity Analysis (RQ3): As revealed in Fig. 4 that the results of both recommendation and explanation generation are influenced by the number of latent topics K. Both tasks achieve their best performance when $K = 10$.



(a) Rating prediction task (b) Explanation generation task

Fig. 4: Trends of MAE and ROUGE-2 values with different number of topics

TABLE IV: Comparison of explanations generated by PEPLER and TPLER models

Example 1	**hotel, location, airport**
Ground-Truth	great **hotel** in great **location**
PEPLER	the **hotel** is very nice and the staff is very helpful
TPLER	the **hotel** is very close to the **airport**
Example 2	**rooms, floor, service**
Ground-Truth	**rooms** are spacious (even the cheaper ones) and nicely furbished
PEPLER	the hotel is connected to the airport by a direct underground
TPLER	the **rooms** are very nice and the **service** is very good
Example 3	**bath, bed, hotel**
Ground-Truth	**bathrooms** supplied with excellent toiletries
PEPLER	the apartment was clean and spacious
TPLER	the **bathroom** was very clean and the amenities were of a high standard

d) Explainability Study (RQ4): Table IV displays explanation examples generated by the PEPLER and our proposed TPLER on TripAdvisor dataset. Ground-Truth denotes the user's actual review label for the hotel, and the words highlighted in bold represent the discrete prompts learned by the model. It is clear that compared with PEPLER, the explanations generated by TPLER contain more topic-enhanced prompt words and are more informative, resulting in higher-quality and more personalized explanations. Furthermore, the explanations generated by TPLER are topically closer to the ground-truth explanations.

VI. Conclusion

In this paper, we propose an explainable recommendation method TPLER based on pre-trained transformer model, which improves the personalization and quality of explanations by mining topical structure of the textual review. We first design a topic-enhanced feature extraction strategy based on the LDA model. Then continuous prompt and discrete prompt are obtained from ID embeddings and extracted features. Finally the continuous prompts and discrete prompts are integrated as the input of pre-trained transformer model. Two-stage multi-task training strategy is proposed to jointly optimize the

rating prediction and explanation generation tasks. Extensive experiments conducted on Amazon, TripAdvisor and Yelp datasets demonstrate the superiority of our TPLER model over various state-of-the-art baselines. For future work, we will adopt a more capable pre-trained language model to improve explanation quality. In addition, beyond considering topic structures, exploring other information such as sentiment and opinions in reviews will also help enhance the personalization of explanations.

REFERENCES

[1] Y. Meng, C. Guo, Y. Cao, T. Liu, and B. Zheng, "A generative re-ranking model for list-level multi-objective optimization at taobao," in *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2025, pp. 4213–4218.

[2] Y. Zhou, Q. Zhu, J. Jin, and Z. Dou, "Cognitive personalized search integrating large language models with an efficient memory mechanism," in *Proceedings of the ACM Web Conference 2024*, 2024, pp. 1464–1473.

[3] Z. He, Z. Xie, R. Jha, H. Steck, D. Liang, Y. Feng, B. P. Majumder, N. Kallus, and J. McAuley, "Large language models as zero-shot conversational recommenders," in *Proceedings of the 32nd ACM international conference on information and knowledge management*, 2023, pp. 720–730.

[4] Y. Zhang, X. Chen *et al.*, "Explainable recommendation: A survey and new perspectives," *Foundations and Trends® in Information Retrieval*, vol. 14, no. 1, pp. 1–101, 2020.

[5] O. Seton, P. S. Haghighi, M. Alshammari, and O. Nasraouli, "Tag-boosted explainable matrix factorization methods for multi-style explanations," *ACM ISBN*, pp. 978–1, 2021.

[6] G. Balloccu, L. Boratto, G. Fenu, and M. Marras, "Hands on explainable recommender systems with knowledge graphs," in *Proceedings of the 16th ACM Conference on Recommender Systems*, 2022, pp. 710–713.

[7] S. Xu, J. Ji, Y. Li, Y. Ge, J. Tan, and Y. Zhang, "Causal inference for recommendation: Foundations, methods, and applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 16, no. 3, pp. 1–51, 2025.

[8] G. Peake and J. Wang, "Explanation mining: Post hoc interpretability of latent factor models for recommendation systems," in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018, pp. 2060–2069.

[9] Z. Chen, X. Wang, X. Xie, T. Wu, G. Bu, Y. Wang, and E. Chen, "Co-attentive multi-task learning for explainable recommendation." in *IJCAI*, 2019, pp. 2137–2143.

[10] P. Li, Z. Wang, Z. Ren, L. Bing, and W. Lam, "Neural rating regression with abstractive tips generation for recommendation," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2017, pp. 345–354.

[11] L. Dong, S. Huang, F. Wei, M. Lapata, M. Zhou, and K. Xu, "Learning to generate product reviews from attributes," in *Conference of the European Chapter of the Association for Computational Linguistics*, 2017, pp. 623–632.

[12] L. Li, Y. Zhang, and L. Chen, "Personalized transformer for explainable recommendation," in *Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 2021, pp. 4947–4957.

[13] H. Cheng, S. Wang, W. Lu, W. Zhang, M. Zhou, K. Lu, and H. Liao, "Explainable recommendation with personalized review retrieval and aspect learning," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 2023, pp. 51–64.

[14] L. Li, Y. Zhang, and L. Chen, "Personalized prompt learning for explainable recommendation," *ACM Transactions on Information Systems*, vol. 41, no. 4, pp. 1–26, 2023.

[15] E. Hasan, M. Rahman, C. Ding, J. X. Huang, and S. Raza, "Review-based recommender systems: A survey of approaches, challenges and future perspectives," *ACM Computing Surveys*, vol. 58, pp. 1–41, 2024.

[16] U. Chauhan and A. Shah, "Topic modeling using latent dirichlet allocation: A survey," *ACM Computing Surveys*, vol. 54, pp. 1–35, 2021.

[17] Z. Cheng, Y. Ding, X. He, L. Zhu, X. Song, and M. Kankanhalli, "A3ncf: an adaptive aspect attention model for rating prediction," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018, pp. 3748–3754.

[18] P. Sun, L. Wu, K. Zhang, Y. Su, and M. Wang, "An unsupervised aspect-aware recommendation model with explanation text generation," *ACM Transactions on Information Systems (TOIS)*, vol. 40, no. 3, pp. 1–29, 2021.

[19] Y. Hou, N. Yang, Y. Wu, and P. S. Yu, "Explainable recommendation with fusion of aspect information," *World Wide Web*, vol. 22, no. 1, pp. 221–240, 2019.

[20] J. Shuai, L. Wu, K. Zhang, P. Sun, R. Hong, and M. Wang, "Topic-enhanced graph neural networks for extraction-based explainable recommendation," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023, pp. 1188–1197.

[21] S. Liu, R. Ding, W. Lu, J. Wang, M. Yu, X. Shi, and W. Zhang, "Coherency improved explainable recommendation via large language model," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 11, 2025, pp. 12 201–12 209.

[22] S. Geng, Z. Fu, Y. Ge, L. Li, G. De Melo, and Y. Zhang, "Improving personalized explanation generation through visualization," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022, pp. 244–255.

[23] H. Zhang, H. Song, S. Li, M. Zhou, and D. Song, "A survey of controllable text generation using transformer-based pre-trained language models," *ACM Computing Surveys*, vol. 56, no. 3, pp. 1–37, 2023.

[24] B. Min, H. Ross, E. Sulem, A. P. B. Veyseh, T. H. Nguyen, O. Sainz, E. Agirre, I. Heintz, and D. Roth, "Recent advances in natural language processing via large pre-trained language models: A survey," *ACM Computing Surveys*, vol. 56, no. 2, pp. 1–40, 2023.

[25] Y. Zhuang, Y. Yu, K. Wang, H. Sun, and C. Zhang, "Toolqa: A dataset for llm question answering with external tools," *Advances in Neural Information Processing Systems*, vol. 36, pp. 50 117–50 143, 2023.

[26] M. Yang, M. Zhu, Y. Wang, L. Chen, Y. Zhao, X. Wang, B. Han, X. Zheng, and J. Yin, "Fine-tuning large language model based explainable recommendation with explainable quality reward," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 8, 2024, pp. 9250–9259.

[27] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.

[28] N. Ding, S. Hu, W. Zhao, Y. Chen, Z. Liu, H. Zheng, and M. Sun, "Openprompt: An open-source framework for prompt-learning," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2022, pp. 105–113.

[29] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE transactions on knowledge and data engineering*, vol. 17, no. 6, pp. 734–749, 2005.

[30] F. J. Peña, D. O'Reilly-Morgan, E. Z. Tragos, N. Hurley, E. Duriakova, B. Smyth, and A. Lawlor, "Combining rating and review data by initializing latent factor models with topic models for top-n recommendation," in *Proceedings of the 14th ACM conference on recommender systems*, 2020, pp. 438–443.

[31] Y. Bao, H. Fang, and J. Zhang, "Topicmf: Simultaneously exploiting ratings and reviews for recommendation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 28, no. 1, 2014.

[32] Z. Guo, G. Li, J. Li, and H. Chen, "Topicvae: Topic-aware disentanglement representation learning for enhanced recommendation," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 511–520.

[33] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[34] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[35] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.