Reason Generation for Point of Interest Recommendation via a Hierarchical Attention-based Transformer Model

Yuxia Wu, Guoshuai Zhao, Mingdi Li, Zhuocheng Zhang, and Xueming Qian, *Member, IEEE*

***Abstract*—Existing point-of-interest (POI) recommendation methods only show the direct recommendation results and lack the proper reasons for recommendation. In recent years, explainable recommendation has become an increasingly important subfield in recommendation systems. The aim of explainable recommendation is to provide a reason why an item is recommended to a user. In this way, it helps to improve the transparency, persuasiveness and user satisfaction of recommendation systems. The explainable recommendation should indicate users' preferences for POIs, such as the category and the price. In addition, to increase the diversity of the results, we take emotional intensity into account in our model to generate more vivid reasons. To this end, we propose a hierarchical attention-based transformer model to generate reasons with specific topics and different emotions. With a hierarchical attention mechanism, we can capture the word-level and attribute-level preferences of users. In addition, we also learn the latent representation of the emotion score to generate diverse recommendation reasons. We evaluate the proposed model on a new real-world dataset collected from three travel service websites. The experimental results demonstrate that our method outperforms the related approaches for reason generation.**

***Index Terms*—Explainable recommendation, natural language generation, personalization, recommender system**

I. Introduction

POI is a specific location that someone finds interesting, such as a restaurant, a shopping mall and so on. POI recommendation in travel is crucial for helping people discover interesting attractions [22], [35], [40], [52]. Envision a user planning a tour in a new city. Traditional POI recommendation systems may provide a list of recommended POIs without any explicit reasons or may show the same rigid reasons for all users, such as "users who visited A also visited B". This lack of explanations negatively affects user experience and reduces the likelihood of users accepting the recommendations. Therefore, we focus on explainable recommendation, which provides a reason as to why an item is recommended to a user to enhance transparency, persuasiveness, and user satisfaction [26], [49], [56]. For example, the system may recommend a specific historical landmark with a persuasive reason like this: "Explore an iconic location with rich historical significance. Witness stunning architecture, captivating sculptures, and immerse yourself in centuries of history".

An effective recommendation reason should be able to reflect the user's personalized interests, such as the city, the category tags and cost [38], [42], [47]. We believe that the semantic information is essential to capture users' preferences [38], [47]. Furthermore, in order to make the recommendation reasons more vivid and diverse, we also focus on generating reasons with different emotional intensities. The diversity in traditional POI recommendation mainly refers to recommending different POIs to the same user. The same POI recommended to different users is simply displayed as a recommendation result. In our work, diversity focuses on generating recommendation reasons with different emotional intensities. Our motivation is that users have varied emotional preferences and may seek different experiences even when visiting the same POI. For example, regarding the Great Wall of China, the system may provide different reasons such as "Great Wall: Awe-inspiring marvel of engineering, panoramic views, rich historical significance, and a captivating journey through ancient times" with strong positive emotion and "Great Wall: Tranquil beauty, winding paths through lush landscapes, serene escape, and breathtaking vistas that evoke peace and harmony" with mild emotion emphasizing the sense of peace. In this way, the system caters to tourists with varying preferences, allowing them to connect with the Great Wall of China on different emotional levels and providing a more personalized experience.

Therefore, we attempt to generate personalized reasons with specific topics and emotional intensity for different users. As shown in Fig. 1, given the information of the POI, the preferred topics of users and the emotion score, our task is to generate the corresponding sentence as the personalized recommendation reason. The POI information contains the city name, the POI name and the tags. The emotion score represents the emotional intensity and is used to enhance the diversity of the recommendation reasons. For the same POI and the same topics, our model generates different reasons with different emotion scores. We can observe that when the emotion score is lower, the emotional intensity of the recommendation reason is not as strong. We leverage Transformer [37] as our basic model and apply the word vector in BERT [23] as the initial embedding vector.

There are several challenges to generating personalized

This work was supported in part by the NSFC under Grant 61732008 and 61772407

Yuxia Wu, Mingdi Li and Zhuocheng Zhang are with the School of Information and Communication Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: wuyuxia@stu.xjtu.edu.cn; limingdi@stu.xjtu.edu.cn; zhuocheng_zhang@163.com)

Guoshuai Zhao is with the School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: guoshuai.zhao@xjtu.edu.cn)

Xueming Qian (corresponding author) is with the Ministry of Education Key Laboratory for Intelligent Networks and Network Security, the School of Information and Communication Engineering, and SMILES LAB, Xi'an Jiaotong University, Xi'an 710049, China. (e-mail: qianxm@mail.xjtu.edu.cn)

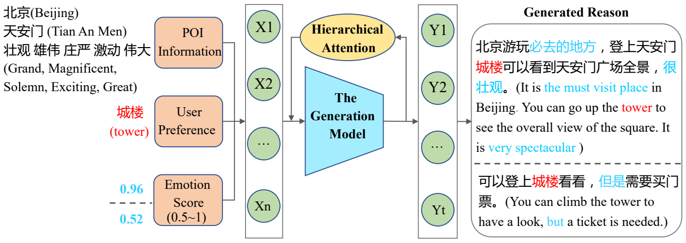

北京(Beijing)
天安门 (Tian An Men)
壮观 雄伟 庄严 激动 伟大
(Grand, Magnificent,
Solemn, Exciting, Great)
城楼
(tower)
0.96
0.52
POI Information
User Preference
Emotion Score (0.5~1)
X1
X2
…
Xn
Hierarchical Attention
The Generation Model
Y1
Y2
…
Yt
Generated Reason
北京游玩必去的地方，登上天安门城楼可以看到天安门广场全景，很壮观。(It is the must visit place in Beijing. You can go up the tower to see the overall view of the square. It is very spectacular)
可以登上城楼看看，但是需要买门票。(You can climb the tower to have a look, but a ticket is needed.)

Fig. 1. Flowchart of our reason generation model.

recommendations. (1) Lack of appropriate datasets with acceptable reasons. We only have some travel website comments, of which only a portion can be considered valid reasons. (2) Existing methods have difficulty in capturing users' preferences and generating reasons based on different attributes. (3) Generating diverse recommendation reasons is still much less explored.

To address the aforementioned challenges, we construct a new dataset and propose a new reason generation model. The reviews imply the preferences of users [24], [51]. Therefore, we scrape users' reviews from travel sites such as Ctrip, Qunar and Baidu Travel and select suitable comments as recommendation reasons. The topics of each POI are extracted to represent the preference of users. Then, the relevant comments are retrieved by the target topics as the output of the generation model in Fig. 1. The emotion score of each comment is obtained by emotional classification. After preprocessing, we design a new reason generation model with a hierarchical attention mechanism to learn the fine-grained preferences of users. We use word-level and attribute-level attention to learn preferences at different levels. In the decoder stage, encoder-decoder attention helps us learn the weight of each attribute term. This enables us to understand specific preferences and generate personalized reasons. To enhance diversity, we generate varied reasons using different emotion scores.

It is worth noting that our model is not limited by the type of input information. In practical applications, we can use different information as input depending on the practical situation. In addition, the input information is easy to obtain in the real-world scenario of POI recommendation. The POI information can be obtained from the meta information of the backend database of the candidate POIs. The tags can be extracted from the category, tips or reviews by the commonly used NLP tools or the preprocessing method used in our paper. The user preference can be predicted by another recommendation model in reality, which is out of our scope in this paper due to the difficulty in obtaining the interaction histories of the users. Therefore, we only focus on the recommendation reason generation part and regard the user preference and the target POI as known information. Moreover, it is reasonable if we integrate the preference prediction model with our reason generation model. In future work, we will also work to make our model more practical and flexible. 3) The emotion score in the input is used to improve the emotional diversity of the generated reasons. In the real-world scenario of POI recommendation, we can obtain the score by sentiment analysis of the ground truth reasons as in our paper during the training process. For the testing process, we can set different scores as input to obtain diverse recommendation reasons.

Our contributions can be summarized as follows:

(1) We formulate a new problem to generate recommendation reasons for POIs based on the preference of each user. To achieve this, we also propose a filtering method for a large-scale dataset for our task. Through preprocessing, we obtain the attribute representations of the POI, the preferred topics of each user and the emotion score. Our task is to generate appropriate reasons to recommend the POI to the specific user and increase the probability of accepting the recommendation.

(2) We propose a new method named the hierarchical attention transformer (HAT) to generate personalized recommendation reasons. By incorporating the word-level and attribute-level attention mechanism, we can better learn users' preferences for different attributes.

(3) The reasons generated by our model are able to better address the personalized preferences of users than the recommendation reasons on travel websites. Furthermore, the learning of emotional representation increases the diversity of reasons. In this way, we can generate reasons more like real interpersonal communication and then improve user experience.

II. Related Work

In this section, we briefly review the recent progress in the areas of explainable recommendation, personalized tour recommendation and textual generation, which are highly relevant to our work.

A. Explainable Recommendation

Generally, explainable recommendation can be divided into four categories [49]: 1) explanations based on relevant users or items; 2) feature-based explanations; 3) textual sentence explanations; and 4) visual explanations. Our work is more related to textual sentence explanations [8], [25], [26], [36], [50], [57]. Li et al. [25] proposed a multitask learning model to simultaneously predict ratings and generate abstractive tips for an item. Chen et al. [8] focused on verifying the usefulness of online reviews and proposed a neural attentional regression model with review-level explanations (NARRE). They applied an attention mechanism to automatically learn the weights of reviews and select highly useful reviews. The selected reviews were utilized for learning the latent vectors of users and items and improving the ratings prediction performance. Zhao et al. [50] collected a new large-scale real-world dataset for generating conversational reasons in the song recommendation domain. They proposed an encoder-decoder model with an attention mechanism to generate reasons for recommendation. They also integrated the tags of users to enhance the personality of the generated reasons. Li et al. [26] combined the long-term and short-term preferences of users to make recommendations. Then, they designed a back-routing scheme to generate explanations for users, such as "Item A is similar to Item B, which you watched for a long time" or "Item A is similar to Item B, which you recently watched". Sun et al. [36] proposed a dual learning-based model by jointly predicting user preference and generating reviews.

Our work differs from existing models because they are not designed for recommendation reason generation. Existing models focus on generating tips or reviews without considering user preferences and item characteristics [8], [25], [36]. Template-based approaches lack vitality, personality, and diversity [26]. While some models [50] consider word-level attention, our work incorporates high-level attribute representations and captures both word-level and attribute-level user preferences. Additionally, we introduce a new dataset that enhances the personality and diversity of generated reasons by considering different topics and emotional intensities.

B. Personalized Tour Recommendation

There are emerging studies about personalized tour recommendations to explore users' preferences and suggest itineraries for users.

The work in [11] focused on personalized 3D navigation and understanding of geo-referenced scenes. They proposed a best view algorithm considering both semantic and geometric features of the scene. They also estimate the camera speed of the trajectory based on the projected complexity of the texture of the image. A hierarchical framework is adopted for route planning estimation. Then, Yiakoumettis et al. [43] developed a personalized 3D route planning algorithm to explore users' preferences by active learning. Specifically, they designed an online learning strategy with a relevance feedback method to automatically exploit and adjust users' personalized weights on scene metadata. Aksenov et al. [1] introduced a three-level approach for personalized tour recommendation by integrating dynamic user profiles. They presented and discussed the characteristics and challenges of the three-level approach: tour programming, tour scheduling and travel route determination. The authors in [30] proposed two approaches to recommend an itinerary for users. The first one used visit frequency to represent the popularity of each POI and then recommended the itinerary based on the most popular POIs. The second one leveraged sentiment analysis of the textual opinions about a visited POI of users to explore their interests. Zhao et al. [53] focused on a visual feature enhanced tour recommender system. An end-to-end visual-enhanced probabilistic matrix factorization model (VPMF) was proposed to learn users' preferences by integrating visual features into the collaborative filtering model.

C. Natural Language Generation

Due to the recent success of deep learning techniques in natural language processing (NLP), models based on neural networks have obtained impressive improvements in various tasks [4]. For natural language generation (NLG), many methods have been proposed, such as seq2seq-based methods, reinforcement learning (RL)-based methods, attention-based methods, and variational autoencoder (VAE)-based methods.

Generally, seq2seq models contain encoder and decoder modules to encode the input sequence into a latent representation and then decode it into the desired sequence. To encode the input sequence, the neural network language model (NNLM) [6] was first proposed to exploit the advantages of neural networks for text generation tasks. To tackle the long-term dependency problem, Mikolov *et al.* [28] recurrent neural network-based language model (RNNLM) by leveraging the RNN structure. Subsequently, some variants of RNNs, e.g., long short-term memory (LSTM) [19] and gated recurrent units (GRUs) [10] have been proposed. In practice, RNNs are generally trained by maximizing the likelihood of each target token given the current state of the model and the previous target token. However, as argued in [5], the performance of RNNs suffers from *exposure bias* [5]. Therefore, researchers have proposed reinforcement learning (RL)-based models, such as generative adversarial nets (GANs) [14] and SeqGAN [46]), that use a discriminative model to guide the training of the generative model [27], [44].

In addition, attention-based methods [3], [12], [29], [37], [54] were also successfully applied in the NLG task. Recently, Vaswani *et al* [37] proposed Transformer based solely on attention mechanisms. This model can better capture the dependency among words in sentences by the self-attention mechanism. Then Devlin et al. proposed a word vector pre-training model named BERT [23], which utilize bidirectional Transformer to encode both left and right context to representations and achieved great performance in many natural language processing tasks [9], [48].

VAE-based methods offer a different approach to generative modeling by integrating stochastic latent variables into the conventional autoencoder architecture [7], [15]. For example, Bowman et al. [7] proposed a VAE model with LSTM as the encoder and decoder model. Semeniuta et al. [33] proposed a hybrid architecture that blends fully feedforward convolutional and deconvolutional components with a recurrent language model. Hu et al. [20] combined VAE and discriminators to generate sentences with explicit constraints.

The differences between our method and related works lie in the following: (1) The difference with the current personalized tour recommendation is that the existing tour recommendation methods only focus on the location recommendation without considering the explanations. Our method focuses on the revenue generation of tour recommendations. (2) The difference with the existing natural language generation methods is that they focus on modeling the sequential semantic relationship among the input words. The high-level attribute information is also important in our task. Therefore, we add an attribute-level attention module to better capture users' preferences on different attributes.

III. Problem Definition

We formulate the problem of reason generation for POI recommendation as follows.

Definition 1: POI information. Each POI is represented by (C, P, K), where C denotes the city name; P is the POI name; and $K = \{k_1, k_2, \cdots, k_m\}$ is the tag of the POI, which contains *m* words representing the inherent attributes.

Definition 2: The preferred topics of the user. We assume that a user may be interested in *n* topics $T = \{t_1, t_2, \cdots, t_n\}$ of the POI when he or she decides where to go.

Definition 3: The emotion score *E*. It represents the extent of positivity to enhance the diversity of the recommendation reasons.

Formally, given the combination set of the information of POI (C, P, K), the d topics T and the emotion score E, our target is to generate a sentence with N words $Y = \{y_1, y_2, \cdots, y_N\}$ as the recommendation reason.

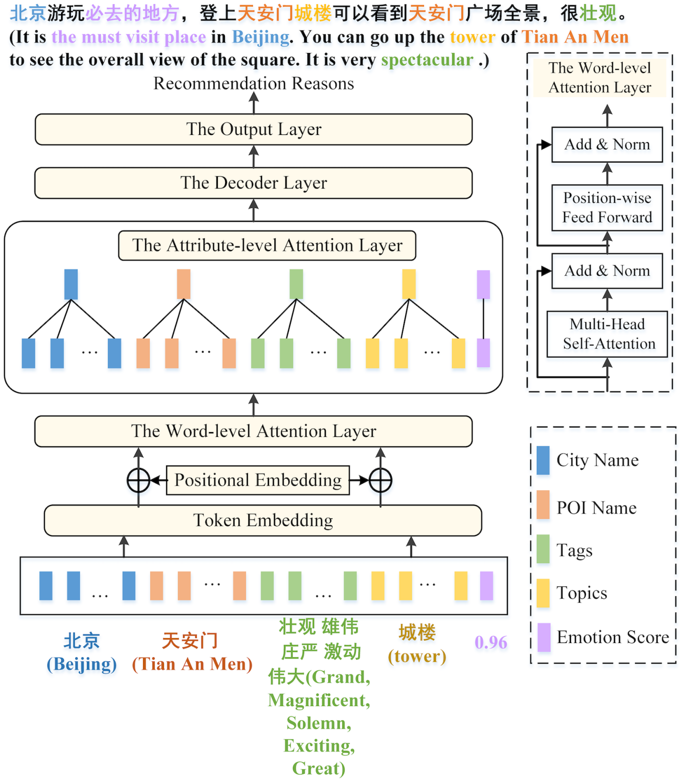

北京游玩必去的地方，登上天安门城楼可以看到天安门广场全景，很壮观。
(It is the must visit place in Beijing. You can go up the tower of Tian An Men to see the overall view of the square. It is very spectacular .)
Recommendation Reasons
The Output Layer
The Decoder Layer
The Attribute-level Attention Layer
The Word-level Attention Layer
Positional Embedding
Token Embedding
北京 (Beijing)
天安门 (Tian An Men)
壮观 雄伟 庄严 激动 伟大(Grand, Magnificent, Solemn, Exciting, Great)
城楼 (tower)
0.96
The Word-level Attention Layer
Add & Norm
Position-wise Feed Forward
Add & Norm
Multi-Head Self-Attention
City Name
POI Name
Tags
Topics
Emotion Score

Fig. 2. The overall architecture of our hierarchical attention transformer model. Each color of the input words indicates different attributes. (Note that the input sequence in our collected dataset contains a series of Chinese words. Here, we show the corresponding English words to facilitate understanding.)

IV. Method

In this section, we introduce the architecture of our hierarchical attention-based Transformer (HAT) model shown in Fig. 2. First, the input words are fed into the embedding layer to learn hidden features. Then, word-level and attribute-level attention layers are designed to learn the high-level features of each attribute. Finally, the recommendation reason is generated by the decoder and output layer.

A. The Embedding Layer

The embedding layer aims to learn the latent feature of the input textual sequence $X = \{x_1, x_2, \cdots, x_M\}$. We regard each word as a token by splitting the input text into a single word based on the spaces in the text [21], [39].

In addition, the order of tokens in the input text is important for understanding grammar and semantics. Researchers have shown their significance for language tasks [13], [18], [34]. The Transformer architecture replaces RNN with self-attention, which is order-independent and captures longer dependencies. Therfore, a positional embedding layer is added. In the next part, we discuss token embedding and positional embedding in detail.

1) Token Embedding: For each word $x_k \in X$, we first represent it as a $|D|$-dimensional one-hot vector r_k, where the nonzero entry denotes the index for the corresponding location in $D = \{D_1, D_2, \cdots, D_{|D|}\}$ and $|D|$ is the total number of the words in the vocabulary. Then we learn an embedding matrix $\boldsymbol{W}_{\rm D} \in \mathbb{R}^{|D|\times d}$, where d denotes the dimension size of the embedding. With the matrix $\boldsymbol{W}_{\rm D}$, we can transform the one-hot vector r_k into a d-dimensional embedding vector e_{token} based on the following equation:

$$e^k_{token} = \boldsymbol{W}_{\rm D} \cdot r_k. \tag{1}$$

To better learn the latent vectors, we apply the pretraining model BERT [23] to initialize the embedding matrix $\boldsymbol{W}_{\rm D}$. We set the size d to be 768 which is the same as in the pretrained BERT model.

2) Positional Embedding: To model the positional relationship, researchers propose to leveraging the character of the sin/cos function [37]. Considering the periodicity of sin/cos function, the representations will be the same for two different positions if *sin(pos)* is directly used to represent the feature of position *pos*. Therefore, different wavelengths are used for different dimension indices of the representation vector. The detailed calculation is as follows:

$$PE(pos, 2i) = sin(pos/10000^{2i/d}), \tag{2}$$

$$PE(pos, 2i+1) = cos(pos/10000^{2i/d}), \tag{3}$$

where pos is the position of the word x_k in the input sequence (pos = 0,1,...,M-1). i is the dimension index of the d-dimensional embedding vector e_{pos}. Suppose that the embedding dimension is d, then i = 0,1,...,d/2-1.

This function was chosen because it would allow the model to easily learn to attend to relative positions. The embedding vector of position *pos* is:

$$\begin{aligned} e_{pos} = [&sin(pos), cos(pos), \\ &sin(pos/10000^{2/d}), cos(pos/10000^{2/d}), \cdots, \\ &sin(pos/10000^{d-2/d}), cos(pos/10000^{d-2/d}] \end{aligned} \tag{4}$$

For any fixed offset k, the trig function allows PE(pos+k) to be represented as a linear function of PE(pos):

$$sin(\alpha+\beta) = sin\alpha cos\beta + cos\alpha sin\beta, \tag{5}$$

$$cos(\alpha+\beta) = cos\alpha cos\beta - sin\alpha sin\beta, \tag{6}$$

Then the PE(pos+k) can be represented as:

$$\begin{aligned} PE(pos+k, 2i) &= PE(\mathrm{pos, 2i}) \times PE(\mathrm{k, 2i+1}) \\ &\quad + PE(\mathrm{pos, 2i+1}) \times PE(\mathrm{k, 2i}), \\ PE(pos+k, 2i+1) &= PE(\mathrm{pos, 2i+1}) \times PE(\mathrm{k, 2i+1}) \\ &\quad - PE(\mathrm{pos, 2i}) \times PE(\mathrm{k, 2i}), \end{aligned} \tag{7}$$

The final embedding vector of x_k is defined as:

$$e_k = e_{token} + e_{pos}. \tag{8}$$

B. The Word-level Attention Layer

The word-level attention layer is composed of a stack of L identical layers. Each layer contains two sublayers, multihead self-attention and position wise feed-forward networks. Each layer extracts essential and useful information and then sends it into the next layer. In this way, the semantic information is gradually extracted. Both the word-level attention layer and the decoder layer in our model are stacked layers with L identical layers.

1) Multi-Head Self-Attention: The attention mechanism has been widely used to measure the dependencies of two items [16], [55]. In our model, the attention module can be treated as a retrieval process. For the retrieval scenario, when users type a query to search for items on the engine, the system will map the query against a set of keys associated with candidate items in the database. Then the system will present the best matched items (values) to the user. Similarly, for the self-attention mechanism, the target token is associated with a query vector and all the other tokens in the input are candidates associated with key and value vectors. The attention weights for the candidate tokens are computed by a compatibility function of the query with the corresponding keys of the candidates.

Here, we apply multihead self-attention to learn the hidden representations of all the words simultaneously. We stack each e_k together into matrix $\boldsymbol{R} \in \mathbb{R}^{L \times d}$. Compared with single attention, multihead attention projects $\boldsymbol{R}$ to h subspaces to capture different representations of the input sequence from different viewpoints.

We apply self-attention using scaled dot-product attention as follows:

$$MultiHead(\boldsymbol{R}) = Concat(head_1, \cdots, head_h)\boldsymbol{W}^O, \quad (9)$$

$$head_i = Attention(\boldsymbol{R}\boldsymbol{W}_i^Q, \boldsymbol{R}\boldsymbol{W}_i^K, \mathbf{R}\mathbf{W}_i^V), \quad (10)$$

$$Attention(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = softmax(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d/h}})\boldsymbol{V}, \quad (11)$$

where $\boldsymbol{Q} = \boldsymbol{R}\boldsymbol{W}_i^Q$, $\boldsymbol{K} = \boldsymbol{R}\boldsymbol{W}_i^K$, and $\boldsymbol{V} = \boldsymbol{R}\boldsymbol{W}_i^V$ represent the queries, keys and values, respectively. They are projected linearly from the same hidden representation matrix $\boldsymbol{R}$ with different projection matrices. $\boldsymbol{W}_i^Q \in \mathbb{R}^{d \times d/h}$, $\boldsymbol{W}_i^K \in \mathbb{R}^{d \times d/h}$, $\mathbf{W}_i^V \in \mathbb{R}^{d \times d/h}$ and $\boldsymbol{W}^O \in \mathbb{R}^{d \times d}$ are learnable parameters matrices. The output of the attention operation is a weighted sum of the values *V*. *Softmax* is used for normalization of the attention weights which is widely used in attention mechanisms [3], [31], [32], [41], [45].

2) Point wise Feed-Forward Networks: To enhance the nonlinearity of our model, we add a point wise feed-forward network (FFN) layer following the self-attention layer [37]. It is composed by a fully connected feed-forward network including two linear transformations and an activation function ReLU.

$$\boldsymbol{FFN} = max(0, MultiHead(\boldsymbol{R})W_1 + b_1)W_2 + b_2. \quad (12)$$

3) Stacking Layer: The stacking layer is used to understand the high-level representations of the input. Each layer extracts essential and useful information and then sends it into the next layer. In this way, the semantic information is gradually extracted. Both the word-level attention layer and the decoder layer in our model are stacked layers with *L* identical layers. Similar to the Transformer model [37], we employ a residual connection [17] around each of the two sublayers, followed by layer normalization [2]. Then, the output of each sublayer is LayerNorm(x + Sublayer(x)), where Sublayer(x) is the function implemented by the sublayer itself.

Finally, the output of the word-level attention for the l^{th} layer is as follows:

$$\boldsymbol{R}^l = SL(\boldsymbol{R}^{l-1}), \quad (13)$$

$$SL(\boldsymbol{R}^{l-1}) = LN(SAttn^{l-1} + PFFN(SAttn^{l-1})), \quad (14)$$

$$SAttn^{l-1} = LN(\boldsymbol{R}^{l-1} + MultiHead(\boldsymbol{R}^{l-1})), \quad (15)$$

where *LN* represents the layer normalization. *SAttn* and *SL* are the self-attention layer and the stacking layer, respectively.

C. The Attribute-level Attention Layer

With the word-level attention layer, we have obtained the high level representations of our input sequence $X = \{C, P, K, T, E\}$. However, different from general text generation, whose input is a fluency sentence or paragraph, the input sequence of our task consists of several attribute terms. As described above, there are different attributes in *X* containing various aspects of the POI, the preferred topics of users and the emotion score. Therefore, it is necessary to learn the latent representation of every attribute.

However, not all the words in one attribute have the same importance. To this end, we propose an attribute-level attention mechanism to obtain a comprehensive representation of attributes. The hierarchical attention mechanism with word-level attention and attribute-level attention modules is the key innovation of the proposed HAT model. Compared with the original transformer model, we add the attribute-level attention module to further capture users' preferences for different attributes.

In our scenario, we obtain five attributes for each input sequence. To unify the symbols, we set $X = \{A^1, A^2, \cdots, A^5\}$. Each element A^j represents one of the sequences C, P, K, T, and E, respectively.

Formally, we denote the word sequence of A^j as $A^j = \{w_1^j, w_2^j, \cdots, w_{N_j}^j\}$. The hidden representation of the word w_i^j is h_i^j, which is obtained from the word-level attention layer. Finally, the attribute-level attention for the attribute j is defined as follows:

$$u_i = tanh(\boldsymbol{W}_a h_i^j + b_a), \quad (16)$$

$$\alpha_i = \frac{exp(u_i^T u_a^j)}{\sum_i exp(u_i^T u_a^j)}, \quad (17)$$

$$v_i = \sum_i \alpha_i h_i^j, \quad (18)$$

where u_a is the learned high level representation vector of the attribute A^j and can be seen as a query vector over the words belonging to the attribute A^j. α_i is the normalized importance weight of each word through a softmax function. v_i is the comprehensive vector that fuses all the information of words in the attribute A^j.

D. The Decoder Layer

Similar to the original Transformer model [37], the structure of the decoder layer is the similar to the word-level attention layer, which is also composed of a stack of *L* identical layers. Except for the multihead self-attention and position wise feed-forward networks, there is another sublayer in the decoder layer, which performs multi-head attention over the output of the attribute-level attention layer. For each time step *i*, the attention module is computed as follows:

$$Attention(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = softmax(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d/h}})\boldsymbol{V}, \tag{19}$$

where ***K*** and ***V*** represent the output latent vector of the attribute-level attention layer. ***Q*** is the output of the decoder layer at time step *i-1* .

Similar to the encoder, we employ residual connections around each of the sublayers, followed by layer normalization. Specifically, the self-attention in the decoder layer is masked multi-head attention to prevent future words from being part of the attention. Then, each sublayer is stacked together and fed into the output layer.

E. The Output Layer

The output layer contains a fully-connected linear layer and a softmax layer to transform the output of the decoder into the V-dimensional matrix O_i. Each element in O_i indicates the probability of words in the vocabulary. The output probability is calculated as follows:

$$p_i = \boldsymbol{W}^T x_i + b, \tag{20}$$

$$O_i = \frac{e^{p_i}}{\sum_{j=1}^{V} e^{p_j}}, \tag{21}$$

where x_i represents the hidden vector of the i^{th} time step. ***W*** and *b* are the weight and bias to be learned. p_i is the probability of the output words in the vocabulary.

F. Model Optimization

For model optimization, we use cross-entropy loss to train our model. Given a training set with $\mathcal{R}$ samples, and the generated reasons with *N* words, the loss function is defined as:

$$J = \frac{1}{\mathcal{R}} \sum_{i=1}^{\mathcal{R}} \sum_{j=1}^{N} y_j \log(O_j) + \lambda \|\Theta\|_2, \tag{22}$$

where *J* is the cross-entropy loss between the generated recommendation reason and the ground truth. $\mathcal{R}$ represents the number of the samples. *N* is the length of the generated reasons. y_j is the one-hot vector of the i^{th} word in the ground truth sentence. O_j is the output probability of our model. $\|\Theta\|_2$ is the regularization term to avoid overfitting. λ controls the importance of the regularization term. To minimize the objective function, we use the Adam optimizer to learn the parameters. The detailed learning process is described as follows. We initialize the embedding matrix for token embedding by the pretraining model BERT. Combined with the positional embedding, we obtain the final embedding vectors of the input words. Then, we feed them into the hierarchical attention layer with stacked word-level and attribute-level attention layers. In this way, we obtain the high level latent representation of the input words. Finally, the recommendation reasons are generated by the decoder layer with the loss function *J*.

V. Data Acquisition

A. Dataset Collection

We collect the reviews of 876 POIs in China from several popular travel service websites including Ctrip[1], Baidu Travel[2] and Qunar[3]. The statistical information of part of the original dataset named as Travel is shown in Table. I. The data prerocessing consists of reason-like data selection and data annotations, which will be introduced in the following parts.

TABLE I
The statistical information of part of the Travel dataset

City	#POIs	#Reviews	#Reviews/POI
Beijing	66	167296	2535
Shanghai	46	91494	1989
Hangzhou	36	48486	1347
Nanjing	40	45135	1128
Chengdu	40	42173	1054

In addition, to improve the persuasiveness of the experimental results, we use a public large-scale dataset from Yelp Challenge 2019[4] for the restaurant domain, which requires only minimal manual preprocessing. Each record includes a user ID, an item ID, a rating score and an explanation written by one user in English.The explanation contains at least one item feature that can be used as the user-preferred topic. We can also extract the POI information from the raw data of the items, including the POI name, the city and the tags (the category terms in the original data). We use the public pretrained BERT model built by Hugging Face[5] to extract the emotion score and only choose the reviews with positive emotion scores higher than 0.5.

B. Reason-like Data Selection

The proper reasons should be positive and have a high correlation with the target POI.

1) Correlation analysis: We train a doc2vec model to learn the feature of each review. The correlation for a given sentence *i* is determined by calculating the cosine similarity between this sentence and other reviews belonging to the same POI.

$$corr_i = sum_{j,j\neq i}^{m} sim_{ij}, \tag{23}$$

where sim_{ij} represents the similarity of the review sentence *i* and *j*. *m* is the number of reviews in one POI. The larger the value of $corr_i$, the more relevant the sentence *i* is to the POI.

2) Emotional analysis: We employ the interface of the Baidu Open Platform for AI[6] to assess the sentiment information within reviews. The advantage of this algorithm is that it can accurately judge emotions and calculate the confidence level for reference and personal users can freely access this interface. By making API calls to the platform, we receive an emotional result for each requested review. This result includes sentiment, confidence, and positive/negative probabilities. The sentiment is the emotional category of the

[1] https://www.ctrip.com/

[2] http://lvyou.baidu.com/

[3] https://www.qunar.com/

[4] https://www.yelp.com/dataset

[5] https://huggingface.co/blog/sentiment-analysis-python

[6] https://ai.baidu.com/tech/nlp_apply/sentiment_classify

review (0: negative, 1: neutral, 2: positive); the confidence indicates the accuracy of the emotional score of the review (value range [0,1]); the positive/negative probabilities indicate the emotional degree of the review. The greater the positive score, the stronger the emotion. For example, when the input text is "The lotus flowers in West Lake are very beautiful, and boating on the lake is also a great choice.", the result is: {"sentiment": 2, "confidence": 0.97, "positive_prob":0.99, "negative_prob":0.01}. We need to select the reviews with high confidence and positive scores.

3) Data selection: We select the reason-like reviews with a threshold by calculating the comprehensive score as follows:

$$score_i = (corr_i + emo_i + con_i)/3 \tag{24}$$

where $corr_i$, emo_i and con_i are the normalized correlation score, emotional score and confidence score of review *i*, respectively. The larger the $score_i$, the more important the review is.

C. Data Annotation

The data annotation consists of tag extraction, topic extraction and emotion score annotation. The emotion score is obtained in the same way as emotional analysis. To facilitate learning the embedding of the emotion, we discretize the emotion score to 1-5.

Now, we introduce the tag and topic extraction process in the following parts.

1) Tag extraction.: For each POI, we first extract the candidate tags from the reviews by word frequency statistics and TF-IDF (Term Frequency–Inverse Document Frequency) and then mannually select proper words as the final tags based on rules of relevance and diversity.

2) Topic extraction.: For the reviews of each POI, we first extract the candidate topic words based on manual selection followed by frequency statistics and TF-IDF. Then, we use the word2vec model to search similar words as an extension. Finally, we manually select proper topics and filter the reviews containing the topic words as ground truth recommendation reasons.

The final statistics of the dataset are shown in Table. II

TABLE II
THE FINAL STATISTICAL INFORMATION OF THE DATASETS

Dataset	#Train	#Validation	#Test
Travel (ours)	103,166	12,986	12,899
Yelp (public)	474,060	74,760	74,126

VI. Experimentation

In this section, we first introduce the experimental setup. Then, we perform extensive comparisons against several state-of-the-art methods and give some discussions about the proposed framework.

A. Experimental Setup

1) Compared Methods: There is no existing special method for the reason generation task for POI recommendation. For comparison, we select the widely used text generation methods and the variant models of our proposed model. Besides, we also discuss the variant models to demonstrate the importance of each part of our model: Transformer and Bert_Transformer.

Seq2Seq + attn [3]: the Seq2Seq model with an attention mechanism. Here, we use LSTM with an attention mechanism as the encoder and decoder model to generate recommendation reasons.

VAE [7]: a variational autoencoder generative model that incorporates distributed latent representations of sentences with LSTM as the encoder and decoder model.

Transformer [37]: The original Transformer model without pretraining in the embedding layer. It contains the word-level attention mechanism in the encoder and decoder module.

Bert_Transformer: The Transformer model with BERT [23] as the pretraining model in the embedding layer.

HAT: Our proposed method based on the original Transformer model. We apply the BERT as the pretraining model in the embedding layer. In addition, we also design a hierarchical attention mechanism containing word-level attention and attribute-level attention layers to learn the word-level and attribute-level attention weights.

2) Evaluation Metrics: The detailed descriptions of the first three metrics are as follows:

(1) Topic relevance (Top.). We train a multilabel classification model on training data to obtain the topics of the generated reasons. The topic relevance is defined as follows:

$$acc\_t_j{}^i = \begin{cases} 1, & gt_j^i \text{ in } output^i \\ 0, & gt_j^i \text{ not in } output^i \end{cases} \tag{25}$$

where $output^i$ and $GT^i = \{gt_1^i, gt_2^i, \cdots, gt_N^i\}$ is the predicted and ground truth topics, respectively.

(2) Emotional accuracy (Emo.). We compute the emotional score by the Baidu Open Platform for AI. Similarly, we discretize the sentiment score to 1-5.

Given the test set with $\mathcal{R}$ examples, the emotional accuracy is defined as follows:

$$acc\_emo^i = \begin{cases} 1, & output\_emo^i \equiv gt\_emo \\ 0, & output\_emo^i \neq gt\_emo \end{cases} \tag{26}$$

where $output\_emo^i$ and gt_emo^i are the predicted and ground truth emotional scores, respectively.

(3) Perplexity (Per.). Perplexity is a method for evaluating the performance of probabilistic generative models. The lower the perplexity score is, the better the performance of the model. For a sample $Y = \{y_1, y_2, \cdots, y_N\}$, the perplexity is defined as:

$$S_{PPL} = P(Y|Model) = exp^{-\sum_{j=1}^{N} \log(p(y_j|y_1,y_2,\cdots,y_{j-1}))/N}, \tag{27}$$

where *Model* is the probabilistic generative model and x_i is the word to be predicted.

For human evaluations, we measure the performance via multiple aspects of quality. We ask six assessors to independently judge the quality of the generated reasons in the test set in terms of four qualities:

(1) Informativeness (Info.): whether the generated reasons contain sufficient information with less meaningless and redundant information;

TABLE III
PERFORMANCE OF ALL THE METHODS ON TRAVEL DATASET

Methods	Top.	Emo.	Per.	BLE.	ROU.	Inf.	Flu.	Con.	Fai.
Seq2Seq + attn	0.43	0.85	3.07	0.11	0.07	1.53	2.73	2.45	2.11
VAE	0.04	0.85	2.59	0.06	0.05	1.52	1.37	1.77	2.04
Transformer	0.18	0.84	1.17	0.15	0.09	2.66	2.80	2.12	2.65
Bert_Transformer	0.53	0.85	1.13	0.15	0.10	2.69	2.82	2.90	2.88
HAT(Ours)	**0.55**	**0.86**	**1.11**	**0.17**	**0.12**	**2.71**	**2.82**	**2.94**	**2.93**

TABLE IV
PERFORMANCE OF ALL THE METHODS ON YELP DATASET

Methods	Top.	Emo.	Per.	BLE.	ROU.	Inf.	Flu.	Con.	Fai.
Seq2Seq + attn	0.79	0.92	0.68	0.23	0.12	2.92	2.94	2.85	2.90
VAE	0.43	0.91	0.70	0.21	0.10	2.80	2.72	2.74	2.79
Transformer	0.62	0.92	0.67	0.32	0.13	2.83	2.95	2.85	2.92
Bert_Transformer	0.81	0.93	0.68	0.35	0.14	2.89	2.95	2.90	2.93
HAT(Ours)	**0.89**	**0.93**	**0.65**	**0.38**	**0.16**	**2.94**	**2.96**	**2.94**	**2.95**

(2) Fluency (Flu.): whether the generated text is understandable, fluent, grammatical, and coherent;

(3) Controllability (Cont.): whether the generated text can satisfy the pregiven constraints;

(4) Faithfulness (Fait.): whether the generated content is consistent with the input information.

Each aspect is rated on a three-point scale, where 1, 2 and 3 indicate bad, acceptable and excellent performance respectively. Considering the large amount of data, we only randomly selected 5,000 samples for human evaluation.

B. Results Analysis and Ablation Study

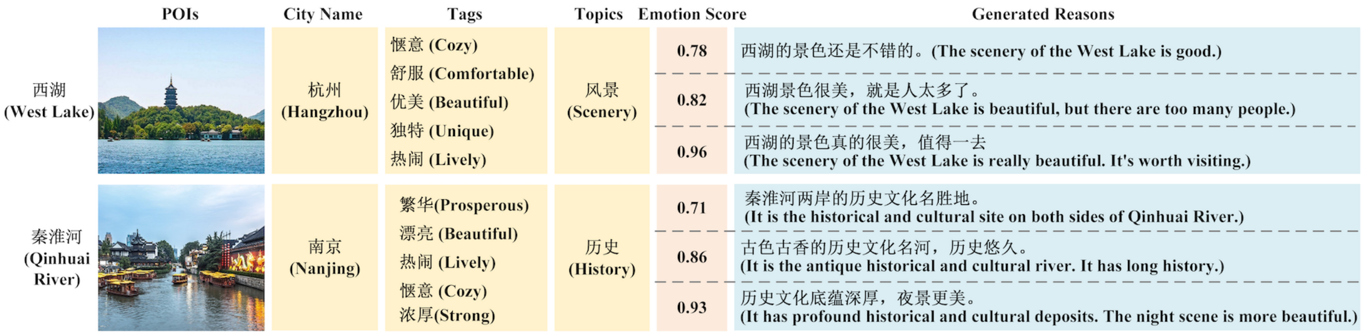
We conduct experiments on one GPU RTX 2080Ti. By running inference for the test dataset, the time cost for each response is approximately 70 seconds. The performance of all methods on the two datasets is illustrated in Table. III and Table. IV. We can observe the following:

(1) The topic relevance and emotional accuracy of Seq2Seq+attn are better than those of VAE. This is because Seq2Seq+attn models the attention weight of different input words when generating the recommendation reasons. The attention mechanism contributes to the important words about the target topics and emotions. The topic relevance of VAE is very low. This is because the VAE focuses on learning the distribution of the inputs and then sampling from the distribution to obtain the output. However, the distributions of the inputs and outputs are different in our scenario. Therefore, it is difficult for VAE to generate reasons with specific topics. The perplexity of Seq2Seq+attn and VAE are higher than others, indicating that the sentences are not fluent enough. This is because they focus on modeling the sequential relationship of the sentence, especially the adjacent words. They fail to model the global influence of each word in the whole sentence.

(2) The Transformer-based methods show better performance than other baselines, such as perplexity, BLEU, ROUGE, and controllability. This indicates that the generated reasons are more fluent and reasonable. This is because the self-attention mechanism in Transformer can model both the local influence of the adjacent words and the global influence of each word. The positional embedding retains the ability to model the sequential relationship of the words. (3) The topic relevance of Transformer is lower than that of other Transformer-based methods. This is because the training set of our dataset is not very large and the number of samples of different topics is unbalanced. This makes it difficult for the model to perform well on all topics.

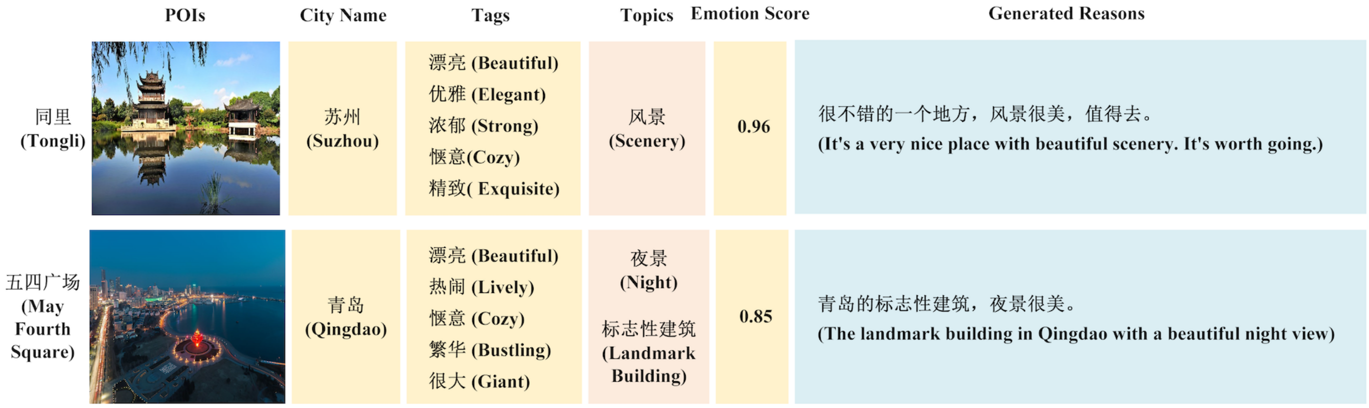
(4) The Bert_Transformer shows better performance than the original Transformer method under many metrics. This demonstrates the effectiveness of the pretraining BERT model. This is because BERT learned the relationship of the words by pretraining on the large-scale corpus. With the pretraining BERT as the initial latent vector in Transfomer, we can better learn the vector of the words.

(5) Generally, the results of the Yelp dataset are better than those of our proposed dataset. This is because Yelp contains many more training samples, which facilitates the training of the models. We can observe that our method shows the best performance compared with the other methods under nearly all the metrics. This is because we apply the pretraining BERT to initialize the latent vector of words. In addition, we design a hierarchical attention module to learn the attention weights when generating reasons for specific topics and emotion scores. With the word-level attention mechanism, we learn the attention weights of the input words. With the attribute-level attention mechanism, we can capture the attribute-based attention weights. Therefore, hierarchical attention can better learn users' preferences and generate reasons with emotional diversity. For BLEU and ROUGE, the results show that our model can generate high-quality reasons much closer than the baselines. This demonstrates the effectiveness of the proposed hierarchical attention model. For human evaluation, the differences in informativeness for the transformer-based model are relatively small. Our model improves the baselines by a large



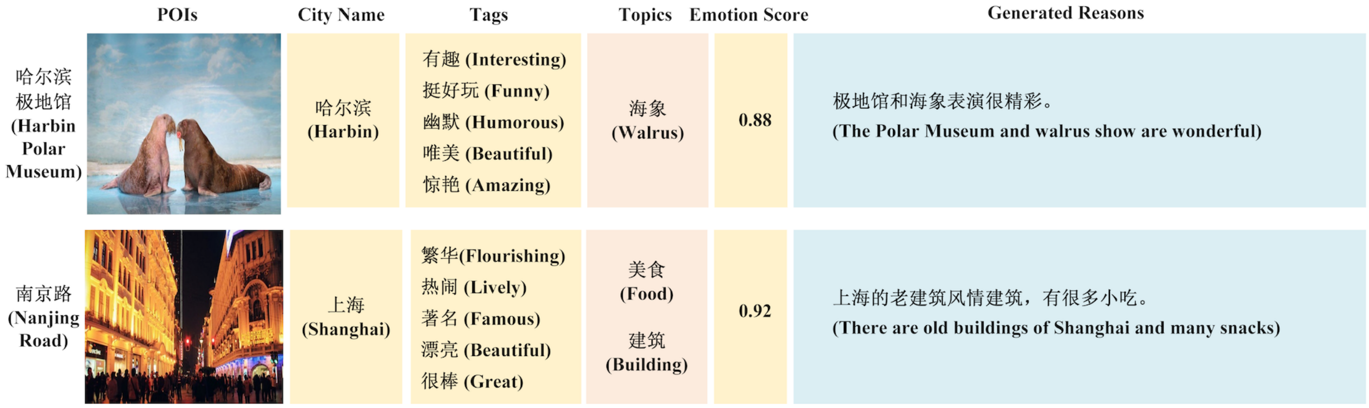
POIs		City Name	Tags	Topics	Emotion Score	Generated Reasons
西湖 (West Lake)		杭州 (Hangzhou)	惬意 (Cozy) 舒服 (Comfortable) 优美 (Beautiful) 独特 (Unique) 热闹 (Lively)	风景 (Scenery)	0.78	西湖的景色还是不错的。(The scenery of the West Lake is good.)
					0.82	西湖景色很美，就是人太多了。 (The scenery of the West Lake is beautiful, but there are too many people.)
					0.96	西湖的景色真的很美，值得一去 (The scenery of the West Lake is really beautiful. It's worth visiting.)
秦淮河 (Qinhuai River)		南京 (Nanjing)	繁华(Prosperous) 漂亮 (Beautiful) 热闹 (Lively) 惬意 (Cozy) 浓厚(Strong)	历史 (History)	0.71	秦淮河两岸的历史文化名胜地。 (It is the historical and cultural site on both sides of Qinhuai River.)
					0.86	古色古香的历史文化名河，历史悠久。 (It is the antique historical and cultural river. It has long history.)
					0.93	历史文化底蕴深厚，夜景更美。 (It has profound historical and cultural deposits. The night scene is more beautiful.)

Fig. 3. Examples of the emotional diversity of the recommendation reasons generated by our method.



POIs		City Name	Tags	Topics	Emotion Score	Generated Reasons
同里 (Tongli)		苏州 (Suzhou)	漂亮 (Beautiful) 优雅 (Elegant) 浓郁 (Strong) 惬意(Cozy) 精致(Exquisite)	风景 (Scenery)	0.96	很不错的一个地方，风景很美，值得去。 (It's a very nice place with beautiful scenery. It's worth going.)
五四广场 (May Fourth Square)		青岛 (Qingdao)	漂亮 (Beautiful) 热闹 (Lively) 惬意 (Cozy) 繁华 (Bustling) 很大 (Giant)	夜景 (Night) 标志性建筑 (Landmark Building)	0.85	青岛的标志性建筑，夜景很美。 (The landmark building in Qingdao with a beautiful night view)

Fig. 4. Examples of reasons with known topics generated by our method.



POIs		City Name	Tags	Topics	Emotion Score	Generated Reasons
哈尔滨极地馆 (Harbin Polar Museum)		哈尔滨 (Harbin)	有趣 (Interesting) 挺好玩 (Funny) 幽默 (Humorous) 唯美 (Beautiful) 惊艳 (Amazing)	海象 (Walrus)	0.88	极地馆和海象表演很精彩。 (The Polar Museum and walrus show are wonderful)
南京路 (Nanjing Road)		上海 (Shanghai)	繁华(Flourishing) 热闹 (Lively) 著名 (Famous) 漂亮 (Beautiful) 很棒 (Great)	美食 (Food) 建筑 (Building)	0.92	上海的老建筑风情建筑，有很多小吃。 (There are old buildings of Shanghai and many snacks)

Fig. 5. Examples of reasons for new topics generated by our method.

margin under controllability and faithfulness, indicating that the generated reasons can better reflect user-preferred topics.

C. Examples of Reasons Generated by Different Methods

We show some examples of the generated reasons in Table. V. We can observe that the reason generated by the VAE is not fluent. Many methods only pay attention to the number of animals and ignore the topic "Torch". Our hierarchical attention based method can better learn the preference of the topic and generate reasons fitting the input topic.

D. Discussions

In this subsection, we give some discussions on our Travel dataset: (1) examples of the emotional diversity of the recommendation reasons; (1) the effectiveness on known topics and new topics; (3) the effectiveness on a single topic and multiple topics; and (4) the influences of different attributes. (5) The effectiveness of the order of different attributes; (6)

TABLE V
EXAMPLES OF THE GENERATED REASONS

Input:
City Name: 北京(Beijing) **POI Name:** 八达岭野生动物世界 (Badaling wildlife world)
Tags: 亲密 好玩 便宜 珍惜 有趣 (intimate, fun, cheap, precious, interesting)
Topic: 接触 (torch) **Emotion Score:** 0.98

Methods	Generated Reasons
Retrieval	很不错，近距离感受动物 (Very Good. We can feel the animal at close range)
Seq2Seq +attn	野生动物园很好玩，动物很多 (The animal park is very interesting and has lots of animals)
VAE	每个，表演很不错，值得一看。 (Each. The performance is very good, it is worth watching.)
Transformer	动物园很大，动物很多 (The zoo is very big and has many animals)
Bert_ Transformer	动物园里面的动物很多，很喜欢。 (There are many animals in the zoo. I like it very much.)
HAT (Ours)	动物种类很多，可以近距离接触。 (There are many kinds of animals. We can touch them at close range.)

visualization of the attention weights; and (7) the impact of the dimensions of the attribute attention.

1) Examples of the emotional diversity of the recommendation reasons: To enhance the diversity of the recommendation reason, we conduct emotional diverse reason generation. Here, we show some examples for the same topic on different emotion scores in Fig. 3. We can observe that our model can generate reasons with different emotion scores, indicating the emotional diversity of the reasons. This will help us generate diverse recommendation reasons for different users and help users perceive the characteristics of the POIs from different views.

2) The effectiveness on known topics and new topics: Our method can generate new and personalized recommendation reasons for both known and new topics. As shown in Fig. 4 and Fig. 5, our model can generate reasons consistent with given topics and emotions.

To demonstrate the effectiveness of our method for known topics and new topics, we also compare our method with other models. The result is shown in Fig. 6. It can be observed that HAT performs best on the old topic, but its performance on the new topic is lower than that of Bert_Transformer. This is because HAT incorporates attribute-level preferences, learning the user's preference weights within the topic terms through training data. The introduction of new topic terms makes it difficult for the model to capture the user's preferences at the new topic level.

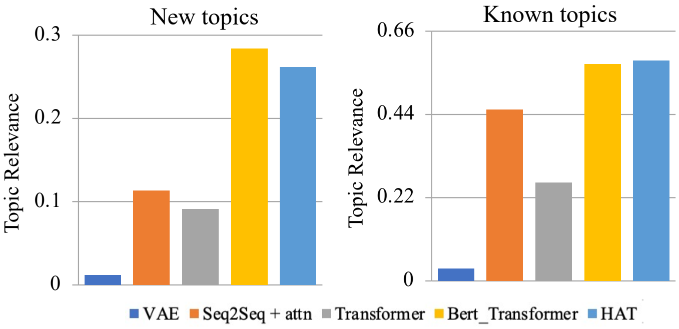


Fig. 6. The discussions of different methods on known topics and new topics.

3) The effectiveness on a single topic and multiple topics: From Fig. 4 and Fig. 5, we can observe that our method can generate proper and interesting reasons for both single and multiple topics. It should be noted that even for new multiple topics that never appear in our training dataset, our method can also generate reasons containing each topic information. In addition, we also evaluate the performance for single and multiple topics on different methods. As shown in Fig. 7, our method improves the performance compared with other methods on both single and multiple topics.

4) The influences of different attributes: We show the influences of different attributes by removing one attribute for the inputs. Considering that the topic and emotion score are the essential parts of our model, we only remove the city name, POI name and tags. The name of the variant models is the combination of the attribute names.

As shown in Fig. 8, the performance of **CPTE** is similar to that of **CPKTE**. This indicates that the tags of POIs show little impact on reason generation with specific topics. The preformance of **PKTE** is the worst, which implies that the city name is an important attribute for generating recommendation reasons.

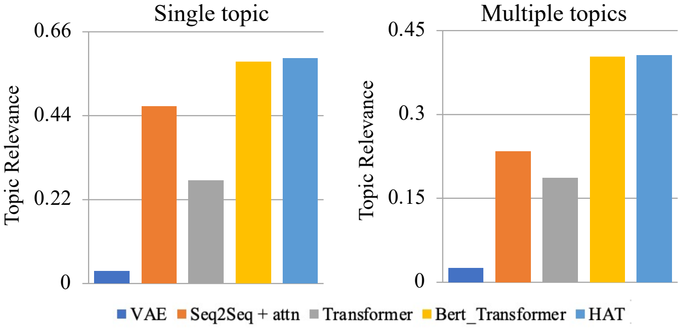


Fig. 7. The discussions of different methods on a single topic and multiple topics.

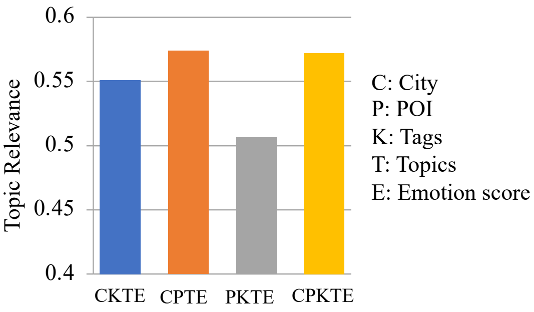


Fig. 8. The impact of different attributes.

5) The effectiveness of the order of different attributes: To show the effectiveness of the order of different attributes, we randomly disturb the order of the attributes. We choose five variant models, including **PCKTE**, **PCTKE**, **ETKCP**, **TKECP**, and **TEKPC**. As shown in Fig. 9, the order does not have a significant impact on the results.

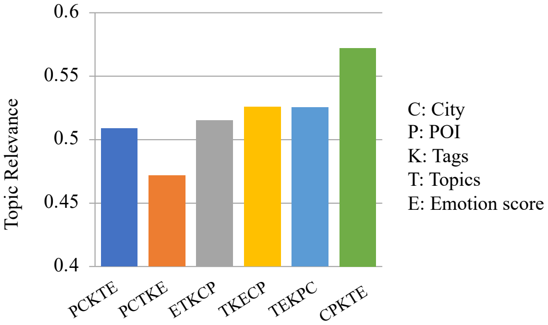


Fig. 9. The effectiveness of the order of different attributes

6) Visualization of the Attribute Level Attention Weights: Fig. 10 shows the attribute-level attention weights of one generated reason. The color of each cell represents the attention weight of the input attribute and the output word. We can observe that when the model generates the word *flourishing*, it pays more attention to the tags that contain *flourishing*. When the model generates the word *street*, it pays more attention to the POI name *Guanqian Street* and the topic *Commericial Street*.

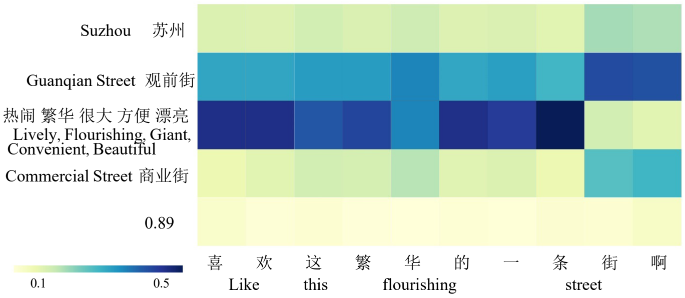


Fig. 10. Visualization of the attribute-level attention weights.

7) The impact of the dimensions of the attribute attention: We set the dimensions of attributes to be 16, 32, 64, 128, 256, 512, 768 with other parameters fixed. As shown in Fig. 11, when the dimension is too large or too small, the performance is not as good. The model performs best when the dimension is 256.

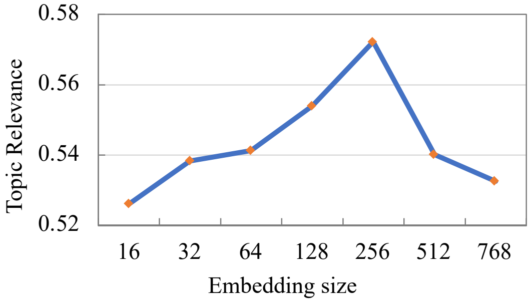


Fig. 11. The impact of the dimensions of the attribute attention.

VII. Conclusion and Future Work

We formulate a new problem to generate personalized recommendation reasons for POIs with specific topics and emotional diversity. To tackle these challenges, we collected a new dataset including POI information (city name, POI name, tags), topics and emotion score as inputs and the corresponding comments as outputs. We also propose a hierarchical attention model based on Transformer to generate recommendation reasons. The experiments demonstrate that our method can better learn users' preferences and improve the emotional diversity of recommendation reasons. Compared with the rigid descriptions of POIs on travel websites, our model can generate personalized and vivid recommendation reasons to attract users' interest. In addition, the integration of emotional information can enhance the diversity of the generated reasons. Our model can be extended to other recommendation reason generation tasks. For other tasks, we just need to change the input information. It can be the attributes of the target item and the preferences of users.

In future work, we will explore the integration of our model with user preference learning models to automatically capture user-preferred topics based on their travel history. We also plan to add the constraints for topic and emotion score into our model to improve the performance.

References

[1] P. Aksenov, A. Kemperman, and T. Arentze, "Toward personalised and dynamic cultural routing: a three-level approach," *Procedia Environmental Sciences*, vol. 22, pp. 257–269, 2014.

[2] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[3] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations*, 2015.

[4] Y. Belinkov and J. R. Glass, "Analysis methods in neural language processing: A survey," *TACL*, vol. 7, pp. 49–72, 2019.

[5] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," in *NIPS*, 2015, pp. 1171–1179.

[6] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *J. Mach. Learn. Res.*, vol. 3, pp. 1137–1155, 2003.

[7] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," in *20th SIGNLL Conference on Computational Natural Language Learning*, 2016, pp. 10–21.

[8] C. Chen, M. Zhang, Y. Liu, and S. Ma, "Neural attentional rating regression with review-level explanations," in *WWW*, 2018, pp. 1583–1592.

[9] Y.-C. Chen, Z. Gan, Y. Cheng, J. Liu, and J. Liu, "Distilling knowledge learned in bert for text generation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7893–7905.

[10] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *EMNLP*, 2014, pp. 1724–1734.

[11] N. Doulamis, C. Yiakoumettis, and G. Miaoulis, "Personalised 3d navigation and understanding of geo-referenced scenes," in *2013 IEEE 14th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*. IEEE, 2013, pp. 1–6.

[12] X. Gao, F. Feng, X. He, H. Huang, X. Guan, C. Feng, Z. Ming, and T. Chua, "Hierarchical attention network for visually-aware food recommendation," *IEEE Trans. Multimedia*, vol. 22, no. 6, pp. 1647–1659, 2020.

[13] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1243–1252.

[14] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial nets," in *NIPS*, 2014, pp. 2672–2680.

[15] D. Guo, D. Tang, N. Duan, J. Yin, D. Jiang, and M. Zhou, "Evidence-aware inferential text generation with vector quantised variational autoencoder," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 6118–6129.

[16] J. Hao, Y. Dun, G. Zhao, Y. Wu, and X. Qian, "Annular-graph attention model for personalized sequential recommendation," *IEEE Transactions on Multimedia*, vol. 24, pp. 3381–3391, 2021.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[18] P. He, X. Liu, J. Gao, and W. Chen, "Deberta: Decoding-enhanced bert with disentangled attention," in *International Conference on Learning Representations*, 2020.

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[20] Z. Hu, Z. Yang, X. Liang, R. Salakhutdinov, and E. P. Xing, "Toward controlled generation of text," in *International Conference on Machine Learning*, 2017, pp. 1587–1596.

[21] C.-R. Huang, P. Šimon, S.-K. Hsieh, and L. Prévot, "Rethinking chinese word segmentation: tokenization, character classification, or wordbreak identification," in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, 2007, pp. 69–72.

[22] M. Islam, M. M. Mohammad, S. S. S. Das, M. E. Ali *et al.*, "A survey on deep learning based point-of-interest (poi) recommendations," *arXiv preprint arXiv:2011.10187*, 2020.

[23] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT*, 2019, pp. 4171–4186.

[24] X. Lei, X. Qian, and G. Zhao, "Rating prediction based on social sentiment from textual reviews," *IEEE Trans. Multimedia*, vol. 18, no. 9, pp. 1910–1921, 2016.

[25] P. Li, Z. Wang, Z. Ren, L. Bing, and W. Lam, "Neural rating regression with abstractive tips generation for recommendation," in *SIGIR*, 2017, pp. 345–354.

[26] X. Li, W. Jiang, W. Chen, J. Wu, G. Wang, and K. Li, "Directional and explainable serendipity recommendation," in *WWW*, 2020, pp. 122–132.

[27] Z. Liu, J. Wang, and Z. Liang, "Catgan: Category-aware generative adversarial networks with hierarchical evolutionary learning for category text generation," in *AAAI*, 2020, pp. 8425–8432.

[28] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, "Recurrent neural network based language model," in *INTERSPEECH*, 2010, pp. 1045–1048.

[29] R. Nallapati, F. Zhai, and B. Zhou, "Summarunner: A recurrent neural network based sequence model for extractive summarization of documents," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[30] P. Padia, K. H. Lim, J. Cha, and A. Harwood, "Sentiment-aware and personalized tour recommendation," in *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019, pp. 900–909.

[31] Y. Pan, T. Yao, Y. Li, and T. Mei, "X-linear attention networks for image captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 971–10 980.

[32] A. Parikh, O. Täckström, D. Das, and J. Uszkoreit, "A decomposable attention model for natural language inference," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 2249–2255.

[33] S. Semeniuta, A. Severyn, and E. Barth, "A hybrid convolutional variational autoencoder for text generation," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 627–637.

[34] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," in *NAACL-HLT (2)*, 2018.

[35] K. Sun, T. Qian, T. Chen, Y. Liang, Q. V. H. Nguyen, and H. Yin, "Where to go next: Modeling long-and short-term user preferences for point-of-interest recommendation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 214–221.

[36] P. Sun, L. Wu, K. Zhang, Y. Fu, R. Hong, and M. Wang, "Dual learning for explainable recommendation: Towards unifying user preference prediction and review generation," in *WWW*, 2020, pp. 837–847.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[38] X. Wang, Y. Zhao, L. Nie, Y. Gao, W. Nie, Z. Zha, and T. Chua, "Semantic-based location recommendation with multimodal venue semantics," *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 409–419, 2015.

[39] J. J. Webster and C. Kit, "Tokenization as the initial phase in nlp," in *The 14th International Conference on Computational Linguistics*, 1992.

[40] Y. Wu, K. Li, G. Zhao, and X. Qian, "Personalized long-and short-term preference learning for next poi recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 4, pp. 1944–1957, 2020.

[41] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*. PMLR, 2015, pp. 2048–2057.

[42] Z. Xu, L. Chen, Y. Dai, and G. Chen, "A dynamic topic model and matrix factorization-based travel recommendation method exploiting ubiquitous data," *IEEE Trans. Multimedia*, vol. 19, no. 8, pp. 1933–1945, 2017.

[43] C. Yiakoumettis, N. Doulamis, G. Miaoulis, and D. Ghazanfarpour, "Active learning of user's preferences estimation towards a personalized 3d navigation of geo-referenced scenes," *GeoInformatica*, vol. 18, no. 1, pp. 27–62, 2014.

[44] H. Yin, D. Li, X. Li, and P. Li, "Meta-cotgan: A meta cooperative training paradigm for improving adversarial text generation," in *AAAI*, 2020, pp. 9466–9473.

[45] A. W. Yu, D. Dohan, M.-T. Luong, R. Zhao, K. Chen, M. Norouzi, and Q. V. Le, "Qanet: Combining local convolution with global self-attention for reading comprehension," in *International Conference on Learning Representations*, 2018.

[46] L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, no. 1, 2017.

[47] J. Zhang, Y. Yang, L. Zhuo, Q. Tian, and X. Liang, "Personalized recommendation of social images by constructing a user interest tree with deep features and tag trees," *IEEE Trans. Multimedia*, vol. 21, no. 11, pp. 2762–2775, 2019.

[48] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with bert," in *International Conference on Learning Representations*, 2019.

[49] Y. Zhang and X. Chen, "Explainable recommendation: A survey and new perspectives," *Found. Trends Inf. Retr.*, vol. 14, no. 1, pp. 1–101, 2020.

[50] G. Zhao, H. Fu, R. Song, T. Sakai, Z. Chen, X. Xie, and X. Qian, "Personalized reason generation for explainable song recommendation," *ACM TIST*, vol. 10, no. 4, pp. 41:1–41:21, 2019.

[51] G. Zhao, X. Lei, X. Qian, and T. Mei, "Exploring users' internal influence from reviews for social recommendation," *IEEE Trans. Multimedia*, vol. 21, no. 3, pp. 771–781, 2019.

[52] K. Zhao, Y. Zhang, H. Yin, J. Wang, K. Zheng, X. Zhou, and C. Xing, "Discovering subsequence patterns for next poi recommendation." in *IJCAI*, 2020, pp. 3216–3222.

[53] P. Zhao, C. Xu, Y. Liu, V. S. Sheng, K. Zheng, H. Xiong, and X. Zhou, "Photo2trip: Exploiting visual contents in geo-tagged photos for personalized tour recommendation," *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[54] Y. Zhao, L. Chen, Z. Chen, R. Cao, S. Zhu, and K. Yu, "Line graph enhanced amr-to-text generation with mix-order graph attention networks," in *Proceedings of the 58th Annual meeting of the association for computational linguistics*, 2020, pp. 732–741.

[55] X. Zheng, G. Zhao, L. Zhu, J. Zhu, and X. Qian, "What you like, what i am: online dating recommendation via matching individual preferences with features," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[56] Y. Zhou, J. Wu, T. H. Chan, S. Ho, D. Chiu, and D. Wu, "Interpreting video recommendation mechanisms by mining view count traces," *IEEE Trans. Multimedia*, vol. 20, no. 8, pp. 2153–2165, 2018.

[57] J. Zhu, Y. He, G. Zhao, X. Bu, and X. Qian, "Joint reason generation and rating prediction for explainable recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 5, pp. 4940–4953, 2022.