Multimodal Contrastive Transformer for Explainable Recommendation

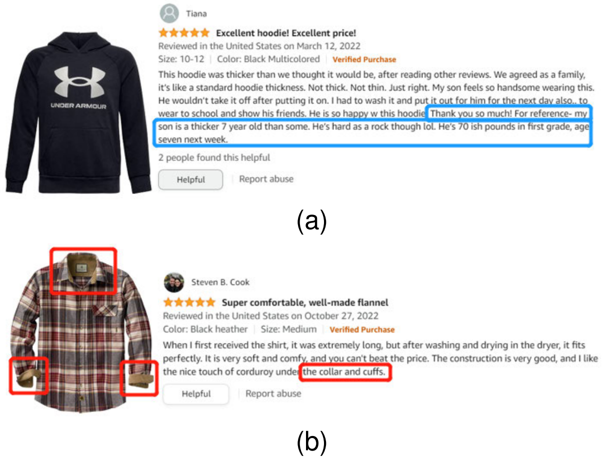
Zhuang Liu, Yunpu Ma, Matthias Schubert, Yuanxin Ouyang, Wenge Rong, and Zhang Xiong

***Abstract*— Explanations play an essential role in helping users evaluate results from recommender systems. Various natural language generation methods have been proposed to generate explanations for the recommendation. However, they usually suffer from two problems. First, since user-provided review text contains noisy data, the generated explanations may be irrelevant to the recommended items. Second, as lacking some supervision signals, most of the generated sentences are similar, which cannot meet the diversity and personalized needs of users. To tackle these problems, we propose a multimodal contrastive transformer (MMCT) model for an explainable recommendation, which incorporates multimodal information into the learning process, including sentiment features, item features, item images, and refined user reviews. Meanwhile, we propose a dynamic fusion mechanism during the decoding stage, which generates supervision signals to guide the explanation generation. Additionally, we develop a contrastive objective to generate diverse explainable texts. Comprehensive experiments on two real-world datasets show that the proposed model outperforms comparable explainable recommendation baselines in terms of explanation performance and recommendation performance. Efficiency analysis and robustness analysis verify the advantages of the proposed model. While ablation analysis establishes the relative contributions of the respective components and various modalities, the case study shows the working of our model from an intuitive sense.**

***Index Terms*— Contrastive learning, explainable recommendation, multimodal information, natural language generation, transformer.**

I. Introduction

RECOMMENDER systems are the most potential technology to tackle the problem of information overload and have been widely used in many fields. Users can rely on recommender systems to filter out the information they are interested in. However, most recommender systems only focus on the accuracy of recommended results while ignoring the explainability. Recently, many explainable recommendation methods have been proposed [1], [2], [3], [4], [5], which can not only provide users with recommendation results, but also generate corresponding explanations.


Tiana

★★★★★ Excellent hoodie! Excellent price!

Reviewed in the United States on March 12, 2022

Size: 10-12 | Color: Black Multicolored | Verified Purchase

This hoodie was thicker than we thought it would be, after reading other reviews. We agreed as a family, it's like a standard hoodie thickness. Not thick. Not thin. Just right. My son feels so handsome wearing this. He wouldn't take it off after putting it on. I had to wash it and put it out for him for the next day also.. to wear to school and show his friends. He is so happy w this hoodie Thank you so much! For reference- my son is a thicker 7 year old than some. He's hard as a rock though lol. He's 70 ish pounds in first grade, age seven next week.

2 people found this helpful

Helpful | Report abuse

(a)

Steven B. Cook

★★★★★ Super comfortable, well-made flannel

Reviewed in the United States on October 27, 2022

Color: Black heather | Size: Medium | Verified Purchase

When I first received the shirt, it was extremely long, but after washing and drying in the dryer, it fits perfectly. It is very soft and comfy, and you can't beat the price. The construction is very good, and I like the nice touch of corduroy under the collar and cuffs.

Helpful | Report abuse

(b)

Fig. 1. Two examples of user reviews on CSJ dataset. (a) Example 1. (b) Example 2.

Compared with other explanation styles (e.g., item neighbors [6], [7] and knowledge graph paths [8], [9]), text explanation is the most intuitive one since users generally provide a large number of reviews for these items they have interacted. The first example in Fig. 1(a) illustrates a user review on the Clothing_Shoes_and_Jewelry (short for CSJ) dataset from Amazon.com. The user *Tiana* rated this hoodie five stars with relevant text explanations such as *"Excellent hoodie! Excellent price!"*. This example indicates that the review text can well reflect the user's preference and make the rating more convincing.

Generally speaking, text explanation generation methods mainly consist of two categories: template-based methods and natural language generation. In particular, template-based methods first predefine a sentence template and then predict the vacant words in the sentence [3], [10]. However, these methods are challenging to meet the diversity and flexibility of the generated explanations. To address this problem, natural language generation approaches have been proposed recently [5], [11]. These methods can automatically generate flexible free-text explanations based on user reviews.

However, there exist three key challenges in current natural language generation approaches. *C1:* user-provided review

Manuscript received 18 December 2022; revised 2 March 2023; accepted 9 May 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61977002, in part by the German Federal Ministry of Education and Research (BMBF) under Grant 01IS18036A, and in part by the State Key Laboratory of Software Development Environment of China under Grant SKLSDE-2022ZX-14. *(Corresponding author: Yuanxin Ouyang.)*

Zhuang Liu and Yuanxin Ouyang are with the State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China (e-mail: liuzhuang@buaa.edu.cn; oyyx@buaa.edu.cn).

Yunpu Ma and Matthias Schubert are with the Lehrstuhl fur Datenbanksysteme und Data Mining, Ludwig-Maximilians-Universitat Munchen, 80538 München, Germany (e-mail: cognitive.yunpu@gmail.com; schubert@dbs.ifi.lmu.de).

Wenge Rong and Zhang Xiong are with the Engineering Research Center of Advanced Computer Application Technology, Ministry of Education, Beihang University, Beijing 100191, China (e-mail: w.rong@buaa.edu.cn; xiongz@buaa.edu.cn).

Digital Object Identifier 10.1109/TCSS.2023.3276273

texts contain noisy data, resulting in the generated explanations that may be irrelevant to the target item. Take Fig. 1 as an example, the sentences in the blue rectangle are regarded as noises, which are irrelevant to the item, and cannot guarantee the quality of the generated explanations. *C2:* due to the lack of a variety of supervision signals, the diversity of the generated explanation is poor, which cannot meet the personalized needs of users. For the second example in Fig. 1(b), according to the review text, the focus of user *Steven B. Cook* lies on the collar and cuffs areas. Therefore, the image of this shirt can provide a supervision signal to guide the explanation generation. In addition, the sentiment information conveyed from the rating can also be viewed as another supervision signal. *C3:* most studies adopt the negative log-likelihood (NLL) as the loss function to train the generation model. However, it often learns an anisotropic distribution for word representations and further leads to the generated texts containing undesirable repetitions [12].

Motivated by these challenges, we propose a multimodal contrastive transformer (MMCT) model for an explainable recommendation. In particular, our model consists of two modules: rating prediction and explainable text generation. The rating prediction module is to provide predicted ratings. Here, we stack two multilayer perceptron (MLP) networks. The first MLP encodes the user–item pair to a latent sentiment vector, which is viewed as a supervision signal in explainable text generation. The second MLP maps the sentiment vector into the numerical rating. The explainable text-generation module aims to generate a recommendation reason consisting of a sequence of words. Motivated by the strong language modeling ability, we appeal to a Transformer model for explainable text generation. (*C1*) To ensure the quality of the ground-truth explanations, we resort to the Sentires toolkit [13] to filter out noisy information in the user reviews. To make the generated explanations related to the user and item, we first encode the user ID, item ID, and item features (e.g., *shoes*, *price*, which can be extracted from user reviews) into latent representation by a personalized Transformer encoder. The user ID and item ID are used to satisfy personalization and the item features aim to guide the model to talk about specific topics. Following [5], we employ a context prediction task to predict the words in the explanation text based on the user ID and item ID. (*C2*) In the stage of generating explanations, we propose a dynamic fusion mechanism, which incorporates multimodal information, including sentiment features (obtained from the recommendation module), item images, and refined user reviews. We obtain the explainable text through the contrastive Transformer decoder and view the multimodal information as supervision signals to guide the explanation generation. (*C3*) In addition, to learn discriminative and diverse explainable texts, we develop an auxiliary contrastive objective for the training of the generation.

The contributions of this article are threefold.

1) We propose the MMCT model for personalized recommendation and natural language explanation generation, which incorporates multimodal information into the learning process, including sentiment features, item features, item images, and refined user reviews.
2) We propose a dynamic fusion mechanism during the decoding process, which induces multimodal information as supervision signals to guide the explanation generation. And we develop a contrastive objective to generate diverse explainable texts.
3) We conduct extensive experiments in terms of explanation performance and recommendation performance on two real-world datasets. Experimental results show the rationality and effectiveness of our model. In addition, we design a comprehensive ablation study to investigate the contributions of the various modalities and components of our model, while the case studies provide a clearer understanding of recommendations.

In the following, we first review some related works in Section II and then formulate the problem of explainable recommendation in Section III. In Section IV, we elaborate on our MMCT framework and introduce each component in detail. Subsequently, we present the experimental results and related analyses in Section V. We conclude this work and discuss some future directions in Section VI.

II. Related Work

In this section, we briefly review the recent progress related to our work: explainable recommendation and contrastive learning-based recommendation.

A. Explainable Recommendation

Explainability of recommendations can have several forms, and we will mainly focus on generation-based approaches here. There are currently two mainstream solutions to generate explainable texts: template-based methods and natural language generation-based methods. Template-based methods adopt some fixed predefined templates to generate explanations, and they predict different words in the templates to personalize them [14], [15], [16]. For example, Extract–Expect–Explain (EX^3) [17] learns important attributes directly from users' historical behaviors to explain the recommendation results, which is a behavior-oriented method. However, it relies on predefined templates and cannot control sentence quality well. To tackle that, NETE [10] designs a "neural template" to guide the model to generate template-controlled sentences that is adaptive to specific features. In summary, template-based methods need extensive human efforts to define various templates in different scenarios, which may hinder the diversity and flexibility of explanations. To address the problem, natural language generation methods have been proposed in recent years [11], [18], [19], [20]. As an earlier work, NRT [21] aims to generate an abstractive tip based on user and item IDs through gated recurrent neural networks. Compared to NRT, MRG [22] incorporates multimodal information to generate explanations, including sentiment features, visual features, and review texts. It further improves the recommendation performance and text quality. To strengthen the connection between the explanation task and recommendation task, SAER [11] proposes a sentiment alignment task to force the recommendation module to influence explanations' learning directly. To enhance the personalization of the generated explanations, PETER [5] is proposed, which is a personalized Transformer model. It designs a context prediction task to predict words in the explanations based on the user ID and item ID.

B. Contrastive Learning-Based Recommendation

Aims to learn high-quality discriminative user and item representations in a self-supervised manner [23], [24], [25], [26], [27]. For example, CLRec [28] proposes a contrastive learning framework for debiased deep candidate generation in recommender systems. It can efficiently reduce selection bias and improve recommendation performance. SGL [29] proposes a self-supervised graph learning framework, which performs dropout operations over the user–item graph. Additionally, CML [30] designs a multibehavior contrastive learning paradigm to capture the transferable user–item relationships from multityped user behavior data. To verify the effectiveness of data augmentation in recommender systems, GACL [31] proposes a graph augmentation-free contrastive learning method. It simplifies the uniformity by adding random noise to users' and items' embeddings and enhances the recommendation performance from a geometric view. In addition, KGCL [26] designed a general knowledge graph contrastive learning framework to alleviate the information noise for knowledge graph-enhanced recommender systems.

III. Problem Formulation

The universal sets of users and items are denoted $\mathcal{U}$ and $\mathcal{I}$, respectively. Their latent embedding matrices are defined as $\mathbf{P} \in \mathbb{R}^{|\mathcal{U}|\times d}$ and $\mathbf{Q} \in \mathbb{R}^{|\mathcal{I}|\times d}$, respectively, and d is the dimension of the embedding vectors. The rating set between the users and items is defined as $\mathcal{R} = \{(u,i)|$, where user u has rated item i with rating score $r_{u,i}\}$. Each item $i \in \mathcal{I}$ is attached with an image, which can be viewed as a visual feature of this item. We also define a vocabulary set $\mathcal{V} = \{w_1, w_2, \ldots, w_{|\mathcal{V}|}\}$ for explanation generation, $\mathcal{F}$ is the feature set, which is a subset of vocabulary $\mathcal{F} \subset \mathcal{V}$. Let $E_{u,i} = \{w_{u,i}^1, w_{u,i}^2, \ldots, w_{u,i}^{|E_{u,i}|}\}(u \in \mathcal{U}, i \in \mathcal{I})$ be the refined review text of user u on item i, where $w_{u,i}^t$ is the tth word, and $|E_{u,i}|$ is the length of the review. We define the set of all user refined reviews as $\mathcal{W} = \{E_{u,i}|(u,i) \in \mathcal{R}\}$.

Formally, given a multimodal recommendation dataset $\{\mathcal{U}, \mathcal{I}, \mathcal{R}, \mathcal{F}, \mathcal{W}\}$, our task is to predict a rating $\hat{r}_{u,i}$ and generate a natural language sentence $\hat{E}_{u,i} = \{\hat{w}_{u,i}^1, \hat{w}_{u,i}^2, \ldots, \hat{w}_{u,i}^{|\hat{E}_{u,i}|}\}$ as an explanation for each user $u \in \mathcal{U}$ and item $i \in \mathcal{I}$ to justify why i is recommended to u, where $|\hat{E}_{u,i}|$ is the length of generated explanation. When generating the explainable texts, we consider multimodal information, including user ID p_u, item ID q_i, item features $F_{u,i} = \{f_{u,i}^1, f_{u,i}^2, \ldots, f_{u,i}^{l_{u,i}}\}$ (it is a subset of feature set $F_{u,i} \subset \mathcal{F}$, $l_{u,i}$ is the length of features for user–item pair (u,i)), item image, sentiment feature $\mathbf{s}_{u,i}$, and refined user review $E_{u,i}$ for item i.

IV. Methodology

In this section, we introduce our MMCT model principle. Specifically, we first give an overview of the proposed model and then detail each model component. At last, we present the overall optimization objective and discuss how to train the model parameters.

A. An Overview of MMCT Model

Our proposed MMCT model consists of two major modules: a rating prediction module and an explainable text-generation module. An overview of the model is given in Fig. 2. The rating prediction module takes a user and item pair (u,i) as input to predict a recommendation score $\hat{r}_{u,i}$ through stacking two MLP networks. The first MLP encodes the (u,i) pair into a latent sentiment vector $\mathbf{s}_{u,i}$, which is defined as a sentiment encoder. The second one maps the sentiment vector $\mathbf{s}_{u,i}$ into the recommendation score $\hat{r}_{u,i}$, which is defined as a rating regressor. The sentiment vector $\mathbf{s}_{u,i}$ can be viewed as a supervision signal to guide the explanation generation. The right part of Fig. 2 depicts the process of generating explainable text. It contains two major components, the encoder for extracting user and item information and the decoder for explainable text generation. Based on Transformer, the encoder extracts features from user u, item i, and item features $F_{u,i}$. The extracted features are fed into the Transformer decoder so that it can generate personalized and fitting topic explanations for the target item. Following [5], we employ a context prediction task to predict the words in the explanation text based on the user ID and item ID. In the decoding process, we propose a dynamic fusion mechanism to guide explanation generation, which considers multimodal information, including an explanation $E_{u,i}$ extracted from user reviews, item visual feature $\mathbf{e}_v$, and sentiment vector $\mathbf{s}_{u,i}$. In this way, we can adaptively incorporate this multimodal information for each user and generate high-quality explanations. In addition, we employ a contrastive Transformer decoder to translate this multimodal information into a sequence of words dynamically as an explanation. The contrastive objective aims to learn discriminative and diverse sentences. We describe each component and corresponding technique in the following subsections.

B. Rating Prediction

In this module, we aim to get the recommendation score $\hat{r}_{u,i}$ and the sentiment vector $\mathbf{s}_{u,i}$ for each user and item pair (u,i) in $\mathcal{R}$. More formally, we define the embedding matrices of users and items as $\mathbf{P} \in \mathbb{R}^{|\mathcal{U}|\times d}$ and $\mathbf{Q} \in \mathbb{R}^{|\mathcal{I}|\times d}$, respectively. Given a user–item pair (u,i), we first encode them into one-hot vector $T_u \in \mathbb{R}^{1\times|\mathcal{U}|}$ and $T_i \in \mathbb{R}^{1\times|\mathcal{I}|}$ and then obtain the user embedding vector $\mathbf{p}_u \in \mathbb{R}^{1\times d} = T_u \cdot \mathbf{P}$ and item embedding vector $\mathbf{q}_i \in \mathbb{R}^{1\times d} = T_i \cdot \mathbf{Q}$ according to the embedding matrix. Next, we concatenate these two vectors together

$$\mathbf{z}_0 = [\mathbf{p}_u; \mathbf{q}_i]^\top \in \mathbb{R}^{2d\times 1}. \tag{1}$$

The sentiment encoder takes $\mathbf{z}_0$ as its input and passes it through an l-layer MLP network to get the sentiment vector $\mathbf{s}_{u,i}$. We can describe this process as follows:

$$\mathbf{z}_1 = \phi(\mathbf{W}_1\mathbf{z}_0 + \mathbf{b}_1), \ldots, \mathbf{z}_l = \phi(\mathbf{W}_l\mathbf{z}_{l-1} + \mathbf{b}_l) \tag{2}$$

where ϕ is the activate function (e.g., sigmoid function, $\phi(x) = 1/(1+e^{-x})$), $\mathbf{W}_1 \in \mathbb{R}^{d\times 2d}, \mathbf{W}_l \in \mathbb{R}^{d\times d}$, $\mathbf{b}_1 \in \mathbb{R}^{d\times 1}, \mathbf{b}_l \in \mathbb{R}^{d\times 1}$ are projection matrix and bias. Here, we view the lth layer's output $\mathbf{z}_l$ as sentiment vector $\mathbf{s}_{u,i}$, $\mathbf{s}_{u,i} = \mathbf{z}_l^\top \in \mathbb{R}^{1\times d}$. Let us note that, on the one hand, the sentiment vector $\mathbf{s}_{u,i}$ is viewed as a supervision signal in the

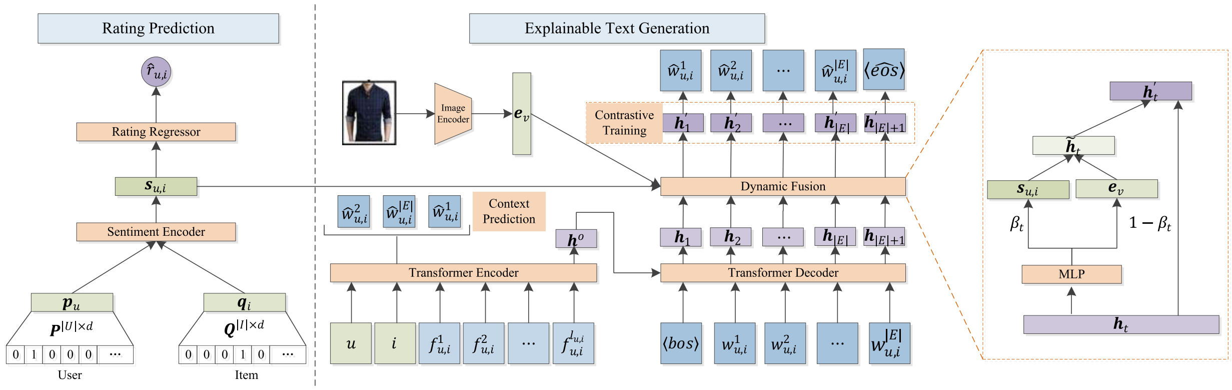


Fig. 2. Architecture of our proposed MMCT model. There are two major modules: a rating prediction module and an explainable text generation module.

process of explainable text generation; on the other hand, it is used by the rating regressor through another MLP to get the final recommendation score $\hat{r}_{u,i}$

$$\mathbf{z}_{l+1} = \phi(\mathbf{W}_{l+1}\mathbf{z}_l + \mathbf{b}_{l+1}), \ldots, \mathbf{z}_L = \phi(\mathbf{W}_L\mathbf{z}_{L-1} + \mathbf{b}_L) \quad (3)$$

$$\hat{r}_{u,i} = \mathbf{W}_r\mathbf{z}_L^\top + b_r \quad (4)$$

where $\mathbf{W}_{l+1} \in \mathbb{R}^{d\times d}, \mathbf{W}_L \in \mathbb{R}^{d\times d}, \mathbf{W}_r \in \mathbb{R}^{1\times d}$, $\mathbf{b}_{l+1} \in \mathbb{R}^{d\times 1}$, $\mathbf{b}_L \in \mathbb{R}^{d\times 1}$, b_r are trainable parameters and L is the number of layers of the MLP network. For this module, we use mean square error (MSE) as the loss function

$$\mathcal{L}_r = \frac{1}{|\mathcal{D}|}\sum_{(u,i)\in\mathcal{D}}\left(r_{u,i} - \hat{r}_{u,i}\right)^2 \quad (5)$$

where $\mathcal{D}$ is the set of user–item pairs in the training set, and $r_{u,i}$ is the ground-truth rating. After training, the rating information is incorporated into $\mathbf{s}_{u,i}$, which can be viewed as a sentiment feature of the user u toward the item i. And it is helpful to guide the explanation generation.

C. Explainable Text Generation

In the following, we will introduce each component of our model in detail to generate explanations for the recommendation results.

1) Personalized Transformer Encoder: To generate personalized explanations, this module aims to encode user and item information, which is then fed into the contrastive Transformer decoder. Specifically, given a user–item pair (u, i), we first form the user ID, item ID, and item features $F_{u,i} = \{f^1_{u,i}, f^2_{u,i}, \ldots, f^{l_{u,i}}_{u,i}\}$ into a sequence

$$X = \left\{u, i, f^1_{u,i}, f^2_{u,i}, \ldots, f^{l_{u,i}}_{u,i}\right\}. \quad (6)$$

The length of X is $l_{u,i} + 2$, where $l_{u,i}$ is the length of item features. Here, the user ID and item ID are used to meet the personalized needs of users, the item features aim to guide the decode module to focus on specific topics. It is worth noting that item features are extracted from user review text, which not only satisfies user preferences but also provides guidance for the explanation generation.

Then, we feed the sequence X into the Transformer encoder to obtain a latent semantic representation $H_K = [\mathbf{h}_{K,1}, \ldots, \mathbf{h}_{K,l_{u,i}+2}]$, where K is the number of layers. Each layer contains two sublayers: a multihead self-attention and a position-wise feed-forward network. To make up the gap between ID distribution and word distribution, we follow [5] and employ a context prediction task to predict the words in the explanation text based on the user ID and item ID. More formally, it utilizes $\mathbf{h}_{K,2}$ to predict the word in the explanation $E_{u,i}$, that leads to

$$\mathbf{c}_t = \text{softmax}\left(\mathbf{W}_v\mathbf{h}_{K,2} + \mathbf{b}_v\right) \quad (7)$$

where $\mathbf{c}_t \in \mathbb{R}^{|\mathcal{V}|}$ is a $|\mathcal{V}|$-sized vector, which represents the probability distribution over the vocabulary $\mathcal{V}$, $\mathbf{W}_v \in \mathbb{R}^{|\mathcal{V}|\times d}$, and $\mathbf{b}_v \in \mathbb{R}^{|\mathcal{V}|}$ are weight parameters. We can sample a word w from $\mathbf{c}_t$ with probability c^w_t. Finally, we adopt NLL as the loss function to implement this task

$$\mathcal{L}_{cp} = \frac{1}{|\mathcal{D}|}\sum_{(u,i)\in\mathcal{D}}\frac{1}{|E_{u,i}|}\sum_{t=1}^{|E_{u,i}|} - \log c_2^{w_t} \quad (8)$$

where $E_{u,i}$ is a ground-truth explanation, which is extracted from the review text of user u toward item i. w_t denotes the predicted word from the second position in the input sequence, t is the time step, $c_2^{w_t}$ is the sample probability for w_t. It is worth noting that the words in context are from the explanation $E_{u,i}$, but they are not sequentially ordered (see Fig. 2).

2) Contrastive Transformer Decoder: This module aims to generate personalized and diverse explainable text for recommendation results based on multimodal information. Here, we take the last step output $\mathbf{h}_{K,l_{u,i}+2}$ as the initial state for the decoder and redefine it as $\mathbf{h}^o$. First of all, we need to extract ground-truth explanations from user reviews. As suggested in [19], some sentences in the user review are irrelevant to the target item, and these noisy data will mislead the model to generate reliable explanations. Take Fig. 1(a) as an example, the sentence in blue rectangle "*Thank you so much! For reference—my son is a thicker 7 year old than some.*" does not provide information about the reason why the user liked this item. Therefore, sentences like this should be filtered out to ensure the reliability of ground-truth explanations. An effective explanation should describe the user's preferences for the item, like "*super comfortable, well-made flannel.*"

To filter out some noises in the user reviews mentioned above, we resort to the Sentires toolkit [13]. In particular, we first adopt this toolkit to extract (*feature*, *opinion*, *sentiment*) triplets from the user reviews to construct a contextual sentiment lexicon (see [13] for detail). Then, we can obtain features (e.g., *shoes, price*) to describe the attributes of the item. Next, we filter out the sentences that do not contain the words with these features, so that each of the remaining sentences contains at least one item feature. Finally, we manually filter out some irrelevant sentences based on domain knowledge to form ground-truth explanations. In this way, we can filter out some noisy information to a certain extent to ensure high-quality explanations. We define the set of all obtained ground-truth explanations as $\mathcal{W} = \{E_{u,i} | (u, i) \in \mathcal{R}\}$.

For a ground-truth explanation $E_{u,i} = \{w_{u,i}^1, w_{u,i}^2, \ldots, w_{u,i}^{|E_{u,i}|}\} \in \mathcal{W}$ of user u toward item i, we produce a special begin-of sequence token $\langle \text{bos} \rangle$ and a special end-of-sequence token $\langle \text{eos} \rangle$ to mark its start and end positions, which leads to $\{\langle \text{bos} \rangle, w_{u,i}^1, w_{u,i}^2, \ldots, w_{u,i}^{|E_{u,i}|}, \langle \text{eos} \rangle\}$. We define each conditional probability in the decoder as

$$p\left(w_{u,i}^t \middle| \langle \text{bos} \rangle, w_{u,i}^1, \ldots, w_{u,i}^{t-1}\right) = p\left(w_{u,i}^t \middle| w_{u,i}^{t-1}, \mathbf{h}_t\right) \tag{9}$$

where $\mathbf{h}_t$ is the context vector for time step t, which can be calculated through the multihead self-attention sublayer

$$\mathbf{h}_t = \text{concat}\left(\mathbf{A}_{L,1}, \ldots, \mathbf{A}_{L,H}\right) \tag{10}$$

where L is the number of layers, H is the number of heads, and $\mathbf{A}_{L,H}$ is the Hth head of the Lth layer's output. For each $\mathbf{A}_{l,h}$, it can be computed as follows:

$$\mathbf{A}_{l,h} = \text{softmax}\left(\frac{\mathbf{Q}_{l,h}\mathbf{K}_{l,h}^{\top}}{\sqrt{d}} + \mathbf{M}\right)\mathbf{V}_{l,h} \tag{11}$$

$$\mathbf{Q}_{l,h} = \mathbf{S}_{l-1}\mathbf{W}_{l,h}^{Q}, \quad \mathbf{K}_{l,h} = \mathbf{S}_{l-1}\mathbf{W}_{l,h}^{K}, \quad \mathbf{V}_{l,h} = \mathbf{S}_{l-1}\mathbf{W}_{l,h}^{V} \tag{12}$$

$$\mathbf{M}_{i,j} = \begin{cases} 0, & i \geq j \\ -\infty, & \text{otherwise} \end{cases} \tag{13}$$

where $\mathbf{S}_{l-1} \in \mathbb{R}^{|E_{u,i}| \times d}$ is the $(l-1)$th layer's output, $\mathbf{W}_{l,h}^{Q} \in \mathbb{R}^{d \times (d/H)}$, $\mathbf{W}_{l,h}^{K} \in \mathbb{R}^{d \times (d/H)}$, $\mathbf{W}_{l,h}^{V} \in \mathbb{R}^{d \times (d/H)}$ are weight matrices, which project $\mathbf{S}_{l-1}$ into query, key, and value, respectively, and $\mathbf{M} \in \mathbb{R}^{|E_{u,i}| \times |E_{u,i}|}$ is the attention masking matrix.

After obtaining the context vector $\mathbf{h}_t$ at time step t, we dynamically incorporate multimodal information as supervision signals to guide the explainable text generation. Then, we can obtain a more descriptive context vector $\mathbf{h}_t'$ from $\mathbf{h}_t$. We will describe the dynamic fusion mechanism in detail in the following subsection. We also adopt NLL as the explanation task's loss function

$$\mathcal{L}_e = \frac{1}{|\mathcal{D}|} \sum_{(u,i) \in \mathcal{D}} \frac{1}{|E_{u,i}|} \sum_{t=1}^{|E_{u,i}|} -\log c_t^{w_t}. \tag{14}$$

However, recent studies [32], [33] show that only using the above objective function to train the language models often leads to the problem of degeneration, especially for Transformer-based models. It yields an anisotropic distribution of word representations in the hidden space and results in high similarity among different words. And it further makes the generated texts contain some undesirable repetitions [12]. To alleviate this problem, we develop a contrastive training objective, which is defined as

$$\mathcal{L}_{cl} = \frac{1}{|E_{u,i}| \times (|E_{u,i}| - 1)} \sum_{i=1}^{|E_{u,i}|} \sum_{j=1, j \neq i}^{|E_{u,i}|} \max\left\{0, \rho - s\left(\mathbf{e}_{w_i}, \mathbf{e}_{w_i}\right) + s\left(\mathbf{e}_{w_i}, \mathbf{e}_{w_j}\right)\right\} \tag{15}$$

where $\rho \in [-1, 1]$ is a predefined margin, $s(\mathbf{e}_{w_i}, \mathbf{e}_{w_j})$ is used to calculate the similarity between the word representations $\mathbf{e}_{w_i}$ and $\mathbf{e}_{w_j}$, and $s(\mathbf{e}_{w_i}, \mathbf{e}_{w_j}) = \mathbf{e}_{w_i}^{\top}\mathbf{e}_{w_j}/||\mathbf{e}_{w_i}||||\mathbf{e}_{w_j}||$. It is worth noting that when the margin $\rho = 0$, the $\mathcal{L}_{cl}$ degenerates to the NLL loss $\mathcal{L}_e$. We fine-tune our model by varying ρ from -1.0 to 1.0 and ensure the best experimental results. In this way, we aim to pull away the distance between representations of distinct words and close the same word representation. This can alleviate the degradation problem to a certain extent and obtain an isotropic distribution of word representations and further generate diverse explainable texts.

At the testing stage, we first feed the $\langle \text{bos} \rangle$ token and the last hidden state $\mathbf{h}^o$ from the Transformer encoder into the Transformer decoder. And then, the decoder can produce a probability distribution $\mathbf{c}_{\langle \text{bos} \rangle}$, and we select greedy decoding to generate a word with the largest probability. We do this repeatedly until it produces $\langle \text{eos} \rangle$ token or reaches a predefined length. This process can be formulated as follows:

$$\hat{w}_{u,i}^t = \underset{w_t \in \mathcal{V}}{\text{argmax}}\, c_t^{w_t} \tag{16}$$

$$\hat{E}_{u,i} = \left\{\hat{w}_{u,i}^1, \hat{w}_{u,i}^2, \ldots, \hat{w}_{u,i}^{|\hat{E}_{u,i}|}\right\} \tag{17}$$

where $\hat{w}_{u,i}^t$ is the generated word in time step t, $\hat{E}_{u,i}$ is the word sequence of generated explanation, and $|\hat{E}_{u,i}|$ is the length of generated explanation.

3) Dynamic Fusion Mechanism: In the process of decoding, we propose a dynamic fusion mechanism, which dynamically incorporates multimodal information, including refined user reviews, item images, sentiment features, and so on. We view this information as a supervision signal to guide the explainable text generation, which can purify the pregenerated topic from the Transformer encoder.

To extract the features of the item image, we consider CLIP [34] as the image encoder, whose parameters are pretrained based on the dataset of 400 million (image, text) pairs collected by [34]. Therefore, the visual feature can be defined as

$$\mathbf{e}_v = \text{CLIP}(\text{img}) \tag{18}$$

where img denotes item i's image information. The sentiment vector $\mathbf{s}_{u,i}$ and the context representation $\mathbf{h}_t$ at time step t have been produced in the previous subsection.

To guide the word generation at time step t, we dynamically incorporate the visual feature $\mathbf{e}_v$ and sentiment vector $\mathbf{s}_{u,i}$. First, we project the current context representation $\mathbf{h}_t$ to a scalar β_t by a single-layer MLP network

$$\beta_t = \sigma\left(\mathbf{W}^x\mathbf{h}_t + b^x\right) \tag{19}$$

where σ is the sigmoid function and $\mathbf{W}^x \in \mathbb{R}^{1 \times d}, b^x$ are projection matrix and bias, respectively. And then, we merge

$\mathbf{s}_{u,i}$ and $\mathbf{e}_v$ dynamically by β_t

$$\tilde{\mathbf{h}}_t = \beta_t \mathbf{s}_{u,i} + (1 - \beta_t)\mathbf{e}_v. \tag{20}$$

Finally, more descriptive context representation $\mathbf{h}'_t$ can be obtained through concatenating $\tilde{\mathbf{h}}_t$ and $\mathbf{h}_t$, for example, $\mathbf{h}'_t = [\tilde{\mathbf{h}}_t; \mathbf{h}_t]$. It is worth noting that β_t is a time-varying gate function to model whether the current word is generated from the visual features or the sentiment features in a soft manner. The larger the β_t is, the more attention is paid to the sentiment features when decoding; otherwise, more attention is paid to the visual features. In this way, we can achieve personalization for each user by adaptively learning the weight parameter β_t. Meanwhile, it can generate a personalized and high-quality explanation for each recommendation result.

D. Model Optimization

To effectively learn the parameters in our proposed MMCT model, we form a linear combination of the above-mentioned objectives to obtain a joint loss function that leads to

$$\mathcal{L} = \lambda_r \mathcal{L}_r + \lambda_{cp}\mathcal{L}_{cp} + \lambda_e \mathcal{L}_e + \lambda_{cl}\mathcal{L}_{cl} \tag{21}$$

where $\lambda_r, \lambda_{cp}, \lambda_e, \lambda_{cl}$ are regularization weights that balance the learning of different components.

E. Model Analysis

The time complexity of our MMCT model mainly contains two parts: rating prediction and explainable text generation. Particularly, the complexity of rating prediction is $O(L(|\mathcal{U}| + |\mathcal{I}|)d^2)$, where L is the number of layers of the MLP network. Generally speaking, $|\mathcal{U}| + |\mathcal{I}| \gg d$, the essential time complexity of rating prediction is controlled by the number of users and items, $|\mathcal{U}|$ and $|\mathcal{I}|$. As for explainable text generation, the Transformer model is the main time cost, which is controlled by the number of ratings $|\mathcal{R}|$, the dimension of embedding vectors d, the number of heads H, and the length of input sequence $|E|$. Summarily, the complexity of the explainable text-generation task is $O(|\mathcal{R}|(|E|^2 dH + |E|d^2))$, which includes the complexity of multihead self-attention $O(|E|^2 dH)$ and the complexity of feed-forward network $O(|E|d^2)$. Since the multihead attention used in the proposed MMCT is parallelizable, the time cost is smaller than RNN-based network, for example, LSTM and GRU.

V. Experiments

A. Datasets

We conduct our experiments on two publicly available Amazon e-commerce explainable recommendation datasets[1]: Movies_and_TV and Clothing_Shoes_and_Jewelry (short for MTV and CSJ, respectively). The way of data preprocessing follows [5]. Each record is comprised of a user ID, an item ID, a rating on a scale of 1–5, an explanation, and a feature. Each item is accompanied by an image. We keep the top 20 000 words with the highest frequency to form our vocabulary $\mathcal{V}$ and mark the rest as "unk." Statistics of the two datasets after processing are shown in Table I. We randomly divide each dataset into three subsets: 80%, 10%, and 10%, for training, validation, and testing. And ensure that each user and item in the testing set is included in the training set. Each user and item in the training set contains at least one record.

TABLE I
Statistics of the Two Datasets

	MTV	CSJ
Number of users	7506	38764
Number of items	7360	22919
Number of images	7360	22919
Number of records	441783	179223
Number of features	5399	1162
average records per user	58.86	4.62
average records per item	60.02	7.82
average words per explanation	14.14	10.48

B. Baselines

We compare our model with the following explainable recommendation baselines.

1) *ACMLM* [19] is an aspect conditional model based on fine-tuned BERT [35] to generate personalized and diverse justifications as explanations.
2) *MRG* [22] is a multimodal review generation model which simultaneously predicts rating with an MLP network and generates explanations with LSTM.
3) *NETE* [10] is a tailored GRU that learns sentence templates from data and generates template-controlled sentences by incorporating a specific feature.
4) *SAER* [11] designs a sentiment alignment task between recommendation and explainable text generation, which force the recommendation module to influence the learning of explanations directly.
5) *PETER* [5] is a personalized Transformer model for explainable recommendation. For a fair comparison, we select feature-based PETER, namely, PETER+, which is the state-of-the-art explanation generation approach.

For recommendation performance, besides MRG, NETE, SAER, and PETER+, we compare our model with another three recommendation baselines.

1) *PMF* [36] is a probabilistic matrix factorization model with the assumptions of Gaussian distribution for rating values.
2) *SVD++* [37] is a singular value decomposition model incorporating neighborhood information and rating values.
3) *NRT* [21] can simultaneously generate short tips and make recommendations based on user and item IDs. However, it only considers ID information. For a fair comparison, we only compare the recommendation performance, not the explanation performance.

C. Evaluation Metrics

1) Recommendation Task: To measure the recommendation performance of different methods, we adopt four widely used evaluation metrics: root MSE (RMSE) and mean absolute error (MAE) for rating prediction, normalized discounted cumulative gain (NDCG), and hit ratio (HR) for personalized

[1] https://nijianmo.github.io/amazon/index.html

ranking. Here, we investigate the top-K recommendation problem and report the ranking results of top-5, for example, NDCG@5 and HR@5. For RMSE and MAE, a lower value indicates better performance, while larger values are better for NDCG@5 and HR@5.

2) Explanation Generation Task: As to explanation performance, following [5], we measure the generated explanations in terms of text quality and explainability. For the former, we adopt three commonly used metrics, BLEU [38], ROUGE [39], and USR [10]. We report BLEU-1 and BLEU-4, and Precision, Recall, and F1 of ROUGE-1 and ROUGE-2 (short for B1, B4, R1-P, R1-R, R1-F, R2-P, R2-R, and R2-F, respectively). For the latter, we adopt the other three metrics proposed by [10]: feature matching ratio (FMR), feature coverage ratio (FCR), and feature diversity (DIV). For DIV, lower values indicate better performance. For others, the higher the scores are, the better the performance is.

D. Implementation Details

We implemented our model in Pytorch. Following [5], we set the embedding size d to 512 and the number of layers and attention heads H in Transformer are both 2. The dimension size of the position-wise feed-forward network in Transformer is set to 2048. The number of layers of the MLP network in the recommendation module is set to 2. We set the weights of different loss function λ_r, λ_{cp}, λ_e, and λ_{cl} to 0.1, 1.0, 1.0, and 1.0, respectively. Grid search is applied to choose the margin ρ over the ranges $\{-1.0, -0.6, -0.3, 0, 0.3, 0.6, 1.0\}$. We optimize the model via stochastic gradient descent and set the batch size to 128. We set the initial learning rate to 1.0 and decrease it by a factor of 0.25 when the loss on the validation set does not decrease. We use an early stopping strategy when there is no improvement on the validation set for five episodes. According to [5], we set the maximum length of generated explanations $|\hat{E}_{u,i}|$ to 15, which is reasonable since the average words per ground-truth explanation in these two datasets are 14.14 and 10.48, respectively. For all considered baselines, the hyperparameters are set according to the suggestions from the settings specified in the original publications. We train all models on a single NVIDIA GeForce GTX 3090 GPU.

E. Explanation Performance

We first analyze the explanation performance with respect to several explainable recommendation baselines. Table II summarizes the best results of all considered baselines on the two datasets. Bold scores indicate the best performance in each column, and underlines indicate the second best. $*$ denotes the statistical significance over the best baseline for $p < 0.05$ via t-test. The last row in each dataset shows the improvements of MMCT relative to the best baseline.

The ACMLM method shows the worst performance in all metrics except USR on the two datasets. This is because ACMLM is a fine-tuned BERT [35], which is achieved by predicting masked tokens. It is quite different from the conventional autoregressive generation. Although it can produce diverse sentences (high USR), the text quality and explainability of them are relatively poor, which indicates that these sentences are less meaningful. In contrast, MMCT can generate high-quality sentences. In terms of USR, our MMCT model significantly outperforms all considered baselines except for ACMLM. This shows that introducing contrastive training in the decoding process can generate more diverse sentences.

Our MMCT consistently outperforms all considered baselines in terms of text quality (BLEU and ROUGE). They measure the quality of the generated explanations at the word level. On the one hand, BLEU is a precision-based metric, which measures how well a generated sentence matches the ground-truth explanation. Our MMCT model achieves a larger BLEU score, which means more overlapping n-grams between generated explanations by MMCT and the ground truth. On the other hand, ROUGE is a recall-based metric, our MMCT model has a higher ROUGE score, which indicates that more content in ground-truth sentences is included in the generated explanations. This further demonstrates our MMCT model is able to produce high-quality texts that are much closer to the ground truth. Moreover, MRG also incorporates multimodal information. However, it is an LSTM-based generative method that may suffer from the notorious long-term dependency problem, and its ability to generate sentences is inferior to the Transformer in MMCT. GRU-based methods such as NETE and SAER are unable to incorporate multimodal information, which slightly performs worse than MRG. PETER+ is a personalized Transformer generative method, and it also incorporates item features. However, it ignores the item image and sentiment feature and cannot generate supervision signals to guide the explanation generation during the decoding process. Therefore, it is slightly inferior to our MMCT model.

As to explainability (FMR, FCR, and DIV), they measure the quality of generated explanations at the feature level. The results show that our MMCT model consistently outperforms all considered baselines in terms of FMR and FCR. Especially, MMCT accomplishes significant performance gains of 20.59% on the MTV dataset and 7.32% on the CSJ dataset in terms of FCR metric against the strongest baselines. This indicates that our model is not only able to generate specific item features, but also covers more feature information. It further illustrates that incorporating item images and sentiment features into the decoding process can effectively guide the generation of specific words. Regarding the explainability metric DIV, MMCT is also very competitive. This further demonstrates that our MMCT model can generate high-quality explanations while ensuring the diversity of sentences (high USR) and features (low DIV) and can also cover more features that users may be interested in.

F. Recommendation Performance

We evaluate the recommendation performance in terms of rating prediction (by RMSE and MAE) and personalized ranking (by NDCG@5 and HR@5) on the two datasets. The results are shown in Table III.

For the rating prediction task, PMF achieves poor performance on the two datasets. This indicates that the matrix factorization method without considering additional side information, which is insufficient to capture the complex

TABLE II

Comparison of the Different Generation Baselines in Terms of Explainability and Text Quality on Two Datasets. BLEU and ROUGE Are Percentage Values (i.e., 9.52 Means 9.52%), While the Others Are Absolute Values (i.e., 0.10 Means 0.10)

	Explainability			Text Quality								
	FMR↑	FCR↑	DIV↓	USR↑	B1↑	B4↑	R1-P↑	R1-R↑	R1-F↑	R2-P↑	R2-R↑	R2-F↑
	MTV											
ACMLM	0.10	0.31	2.07	**0.96**	9.52	0.22	11.65	10.39	9.69	0.71	0.81	0.64
MRG	0.78	0.34	**1.20**	0.49	20.13	3.13	34.63	24.30	26.56	9.11	6.40	6.85
NETE	0.71	0.19	1.93	0.57	18.76	2.46	33.87	21.43	24.81	7.58	4.77	5.46
SAER	0.80	0.28	1.22	0.38	19.82	3.05	34.94	24.16	26.48	9.02	6.34	6.84
PETER+	0.77	0.31	**1.20**	0.46	19.75	3.06	34.71	23.99	26.35	9.04	6.23	6.71
MMCT	**0.84***	**0.41***	1.22	0.58	**20.85***	**3.22***	**35.02**	**25.04***	**27.15***	**9.13**	**6.52***	**6.90***
Improv.	5.00%	20.59%	–	–	3.58%	2.88%	0.23%	3.05%	2.22%	0.22%	1.88%	0.73%
	CSJ											
ACMLM	0.21	0.36	0.12	**0.53**	11.35	1.03	15.64	16.32	14.96	3.32	3.65	3.06
MRG	0.93	0.41	0.05	0.17	23.18	4.38	38.86	30.34	31.21	12.39	9.14	9.20
NETE	0.89	0.28	0.07	0.18	22.35	3.86	37.46	28.36	29.48	10.94	7.53	8.34
SAER	0.93	0.29	**0.04**	0.14	22.00	4.32	39.31	29.40	31.28	12.20	9.02	9.02
PETER+	0.94	0.37	**0.04**	0.16	21.70	4.15	39.40	29.33	30.83	12.34	8.77	9.15
MMCT	**0.95**	**0.44***	**0.04**	0.21	**23.53***	**4.44***	**39.70***	**30.53***	**31.34**	**12.54***	**9.16**	**9.30***
Improv.	1.06%	7.32%	–	–	1.51%	1.37%	0.76%	0.63%	0.19%	1.62%	0.22%	1.09%

TABLE III

Recommendation Performance Comparison of All Considered Baselines in Terms of RMSE, MAE, NDCG@5(%), and HR@5(%)

	Rating Prediction				Personalized Ranking			
	MTV		CSJ		MTV		CSJ	
	RMSE↓	MAE↓	RMSE↓	MAE↓	NDCG@5↑	HR@5↑	NDCG@5↑	HR@5↑
PMF	1.03	0.81	1.15	0.98	0.086	0.052	0.004	0.018
SVD++	0.96	0.72	1.09	0.89	0.095	0.060	0.005	0.020
NRT	0.95	0.71	1.06	0.86	0.146	0.135	0.009	0.037
MRG	0.95	**0.70**	1.06	0.87	0.102	0.077	0.006	0.025
NETE	0.96	0.73	1.07	0.86	0.160	**0.143**	0.009	0.044
SAER	0.95	0.72	1.05	0.85	0.169	0.107	0.007	0.024
PETER+	0.95	0.71	1.05	0.86	0.152	0.122	0.010	0.042
MMCT	**0.94**	**0.70**	**1.04**	**0.84**	**0.171**	0.137	**0.011**	**0.056**

relations between users and items, further limiting the performance. SVD++ slightly outperforms PMF because of the incorporated neighborhood information. The performance of other considered recommendation baselines is relatively close under RMSE and MAE. This is because the rating prediction task is only evaluated on a very small number of unobserved items compared with the personalized ranking task, while not all items are sorted by predicted score, which may cause selection bias in the data [40]. It is consistent with the results shown in [10]. In addition, our MMCT model slightly outperforms the considered baselines. On the one hand, this is due to the mutual promotion between the generation task and recommendation task; on the other hand, the incorporated multimodal information leads to better recommendation performance.

As to the personalized ranking task, we first apply our proposed MMCT model to get the predicted rating scores of each user in the testing set for all items and then sort them in descending order to obtain the top five items with the highest scores. We calculate the corresponding two metrics (e.g., NDCG@5 and HR@5) based on the top-5 results. The most obvious observation is that the performance gap among each considered recommendation baseline widens compared to the rating prediction task. Concretely, multimodal-based methods (NRT, MRG, NETE, SAER, and PETER+) generally perform better than rating-only methods (PMF and SVD++), which shows that incorporating rich multimodal information can effectively improve recommendation performance. Generally speaking, our MMCT model outperforms other recommendation baselines in most cases.

TABLE IV

Efficiency Comparison of Different Models in Terms of Training Minutes on the MTV Dataset

	Time	Epochs	Time/Epoch
ACMLM	216.8	**4**	54.2
MRG	277.2	22	12.6
NETE	206.1	20	10.3
MMCT	**102.7**	19	**5.4**

In summary, our proposed MMCT model generates higher-quality explanations on the basis of guaranteed recommendation performance. On the one hand, it provides a more reliable reason for user decision-making; on the other hand, it describes user preferences better and then makes more accurate recommendations. In addition, this also shows that the training process of the two tasks (rating prediction task and explanation generation task) are complementary, so it is beneficial to train them together.

G. Efficiency Analysis

In this subsection, we investigate the efficiency of different models. We compare the training minutes on the same machine (NVIDIA Geforce RTX 3090) and dataset (MTV), and the results are shown in Table IV. Both ACMLM and our model MMCT are based on Transformer, our model takes less time to train (5.4 min per epoch). This is due to the fact that has only

TABLE V

RESULTS OF ROBUSTNESS ANALYSIS ON THE CSJ DATASET

	Explainability			Text Quality						Recommendation	
	FMR↑	FCR↑	DIV↓	USR↑	B1↑	B4↑	R1-P↑	R1-R↑	R1-F↑	NDCG@5(%)↑	HR@5(%)↑
(Ori)MMCT	**0.95**	**0.44**	**0.04**	**0.21**	**23.53**	**4.44**	**39.70**	**30.53**	**31.34**	**0.011**	**0.056**
FGSM	0.89	0.35	0.10	$0.08_{(-162.50\%)}$	22.51	4.13	38.24	29.46	30.46	$0.006_{(-83.33\%)}$	$0.024_{(-133.33\%)}$
MAG-GAN	0.86	0.31	0.12	$0.06_{(-250.00\%)}$	21.77	3.82	38.27	28.75	30.42	$0.004_{(-175.00\%)}$	$0.018_{(-211.11\%)}$
Adv-BPR	0.87	0.33	0.10	$0.09_{(-133.33\%)}$	22.96	4.21	39.23	30.16	31.05	$0.005_{(-120.00\%)}$	$0.022_{(-154.55\%)}$
GRIP-GAN	0.91	0.38	0.08	0.13	22.74	4.22	38.52	29.75	30.79	0.008	0.037
APR	0.94	0.42	0.05	0.19	23.08	4.39	39.54	30.38	31.26	0.010	0.049

two layers, compared to the 12-layer ACMLM. In addition, since ACMLM adopts pretrained BERT for fine-tuning, it only takes four epochs to converge. While other models need more epochs since they are trained from scratch. Both LSTM-based MRG and GRU-based NETE utilize recurrent neural networks, which take more time than Transformer-based MMCT that can be parallelized. More concretely, compared with LSTM, GRU simplifies the cell state and has fewer parameters, so NETE takes less time than MRG.

H. Robustness Analysis

In order to verify the robustness of MMCT, we compared three attack methods and two defense methods on the CSJ dataset. Among them, FGSM [41] and MAG-GAN [42] aim at attacking the representations of users, items, and words, and GRIP-GAN [43] aims to learn a general robust inverse perturbation for these representations first, then defend against these attacks via a generative adversarial network (GAN). APR [44] is an adversarial training method in the field of personalized recommendation, Adv-BPR [44] aims to attack the representation of users and items by adding adversarial noise to them, and APR conducts adversarial training against Adv-BPR. We set ϵ to 0.5 in FGSM and Adv-BPR, to control the magnitude of adversarial perturbations. The switch η is set to 1 to generate dynamic GRIPs in GRIP-GAN. The experimental results are shown in Table V. The first row shows the original results of MMCT.

It can be observed that the three attack methods, FGSM, MAG-GAN, and Adv-BPR, lead to a sharp drop in recommendation performance and sentence diversity (lower USR). In particular, MAG-GAN decreased by 175%, 211.11%, and 250% in terms of NDCG@5, HR@5, and USR relative to MMCT, which is due to the fact that the recommendation task mainly depends on the representation quality of users and items, when adding noise into them, will degrade its performance. The context prediction task also depends on item representations, and low-quality item representations lead to low USR, which reduces the diversity of generated explanations. These attacks have relatively little impact on text quality and explainability. One possible reason is that when generating explainable text, the image information of the item also provides a part of the supervision signal so that better explanations can also be obtained. In addition, Adv-BPR achieves better text quality and explainability than FGSM and MAG-GAN, since it only perturbs user and item representations, but not word representations. As for defense methods, GRIP-GAN outperforms the above attack methods, which shows that the generated GRIPs by feeding different random noise are able to defend against unconstrained attacks. APR outperforms GRIP-GAN, which shows that the perturbation method in GRIP-GAN is less suitable for perturbing representations, and it may be more suitable for image perturbation in the field of computer vision. In summary, the recommendation task in the MMCT model is weak against attacks, while the supervisory signal from item image information makes the explainable text generation task more robust.

I. Ablation Study

To investigate the respective contribution of our architecture components and various modalities to the overall performance of MMCT, we conduct an ablation study on these two datasets. Specifically, we design three variants to verify the effectiveness of various model components, including removing rating prediction (w/o $\mathcal{L}_r$), context prediction (w/o $\mathcal{L}_{cp}$), and contrastive training objective (w/o $\mathcal{L}_{cl}$). And we verify the effectiveness of various modalities by removing the sentiment feature (w/o $\mathbf{s}_{u,i}$) and visual feature (w/o $\mathbf{e}_v$). Table VI shows the performance in terms of explainability, text quality, and recommendation on MTV and CSJ datasets, respectively. To highlight the effect of each component, we focus on the most obvious metrics of decline in each row. The percentage of relative decline compared with MMCT is shown in the lower right corner brackets.

It can be seen that removing either component will degrade both the explanation and recommendation performance. In particular, the removal of $\mathcal{L}_r$ leads to the recommendation performance drop dramatically. More concretely, the metrics of RMSE and MAE decreased by 240.43% and 324.49%, respectively, compared with MMCT on the MTV dataset, and by 231.73% and 291.67% on the CSJ dataset. And the performance of explainability and text quality also drops slightly. This shows that the MSE loss plays a key role in the rating prediction task and also can improve the explanation performance to a certain extent. When disabling $\mathcal{L}_{\mathrm{cp}}$, the metric of USR decreases most, about 222.22% and 40.00% on MTV and CSJ datasets, respectively, which is consistent with that mentioned in PETER. This indicates that the context prediction task effectively bridges the connection between ID distribution and word distribution, further generating diverse explanations. The contrastive objective $\mathcal{L}_{\mathrm{cl}}$ aims to generate discriminative and diverse sentences; when disabling, the metric USR will be decreased by 13.73% and 31.25% on MTV and CSJ datasets, respectively. In addition, the performance of feature-level also drops on the two datasets (lower FMR, lower FCR, and higher DIV). It can be verified that contrastive

TABLE VI
RESULTS OF ABLATION STUDY ON MTV AND CSJ DATASET. "W/O" MEANS DISABLING THE CORRESPONDING COMPONENT

	Explainability			Text Quality						Recommendation	
	FMR↑	FCR↑	DIV↓	USR↑	B1↑	B4↑	R1-P↑	R1-R↑	R1-F↑	RMSE↓	MAE↓
	MTV										
w/o $\mathcal{L}_r$	0.80	0.33	1.34	0.45	20.69	3.14	34.56	24.93	26.95	$3.20_{(-240.43\%)}$	$2.97_{(-324.49\%)}$
w/o $\mathcal{L}_{cp}$	0.79	0.21	1.32	$0.18_{(-222.22\%)}$	19.01	2.87	34.80	23.55	26.09	0.95	0.71
w/o $\mathcal{L}_{cl}$	0.83	0.39	1.30	$0.51_{(-13.73\%)}$	20.33	3.13	33.77	24.97	26.52	0.95	0.71
w/o $\mathbf{s}_{u,i}$	0.78	0.34	1.24	0.53	20.13	3.13	34.63	24.30	26.56	0.96	0.71
w/o $\mathbf{e}_v$	0.80	0.28	1.23	0.54	19.82	3.05	34.94	24.16	26.48	0.95	**0.70**
MMCT	**0.84**	**0.41**	**1.22**	**0.58**	**20.85**	**3.22**	**35.02**	**25.04**	**27.15**	**0.94**	**0.70**
	CSJ										
w/o $\mathcal{L}_r$	0.94	0.38	0.05	0.18	22.95	4.37	39.38	30.49	31.18	$3.45_{(-231.73\%)}$	$3.29_{(-291.67\%)}$
w/o $\mathcal{L}_{cp}$	0.93	0.41	**0.04**	$0.15_{(-40.00\%)}$	21.89	4.37	39.21	29.68	31.15	1.05	**0.84**
w/o $\mathcal{L}_{cl}$	0.94	0.41	0.05	$0.16_{(-31.25\%)}$	23.14	4.34	39.07	30.27	31.24	1.05	0.85
w/o $\mathbf{s}_{u,i}$	0.93	0.41	0.05	0.20	23.18	4.38	38.86	30.34	31.21	1.06	0.86
w/o $\mathbf{e}_v$	0.93	0.29	**0.04**	0.18	22.00	4.32	38.73	29.40	31.28	1.05	0.85
MMCT	**0.95**	**0.44**	**0.04**	**0.21**	**23.53**	**4.44**	**39.70**	**30.53**	**31.34**	**1.04**	**0.84**

TABLE VII
FOUR DIFFERENT CASES ON THE CSJ DATASET

Case	Image	Rating	Average β	Explanation	
1		5.0	0.83	Ground-truth	*The price is great*
				MMCT	*The price was great and the quality is good* $< eos >$
2		4.0	0.41	Ground-truth	*This is a very comfortable slide in shoe*
				MMCT	*I have a wide foot and this shoe is great* $< eos >$
3		2.0	0.57	Ground-truth	*Well-made but larger sizes not available*
				MMCT	*I ordered a size larger than I normally wear* $< eos >$
4		1.0	0.45	Ground-truth	*It was very small in the waist*
				MMCT	*They are a little tight in the waist* $< eos >$

training objective aims to generate diverse explanations in terms of sentence level and feature level.

Next, we analyze the multimodal information incorporated in our model, including sentiment feature $\mathbf{s}_{u,i}$ and the item's visual feature $\mathbf{e}_v$. It can be found that no matter which modality is removed, both the explanation and recommendation performance will drop. Specifically, the magnitude of the drop is relatively close to the MTV dataset. While on the CSJ dataset, the performance drops more when removing visual feature $\mathbf{e}_v$ relative to sentiment feature $\mathbf{s}_{u,i}$. This indicates that in the field of fashion, users pay more attention to the appearance of items, and visual features perform a more critical role in the process of explanation generation than sentiment features.

Overall, MMCT shows the best performance in terms of explainability, text quality, and recommendation. This further demonstrates the effectiveness of each component and modality in our model. And it can generate high-quality explanations and make more accurate recommendations.

J. Case Study

To gain an intuitive sense of the working of our MMCT model, in Table VII, we present four examples of explanations generated by MMCT on the CSJ dataset. For each case, we display the item image, user's rating (from 1 to 5), average β during the decoding process, and explanation obtained by the ground-truth and MMCT model. It is worth noting that the rating <3 denotes negative sentiment and ≥3 for positive sentiment.

Specifically, for the first case, we can see that the user mainly focuses on the price of the shirt according to the user's review text "*The price is great*" and is satisfied with the price (the rating is 5.0). However, the feature of *price* cannot be conveyed by the item image. Therefore, when generating an explanation, we mainly focus on the sentiment feature from the rating. Moreover, the average value of β learned by MMCT is 0.83 during the process of generating an explanation. This indicates that our MMCT model pays more attention to the sentiment feature than the visual feature when decoding. In the second case, the user's review is "*This is a very comfortable slide in shoe.*" It can be seen that, on the one hand, the user prefers this shoe (the rating is 4.0), and on the other hand, the *wide* feature of the shoe can be obtained from the image, so the average value of β learned by our MMCT model is 0.41. This shows that in the process of generating an explanation, we pay more attention to the item image.

For the latter two cases, the user conveys negative sentiment (rating <3). In particular, for the third case, the user reviews this pant that *"Well-made but larger sizes not available."* This indicates that the user is satisfied with the material of the pants, but the size is too large, thus giving a negative rating of 2.0. Meanwhile, the average value of β learned by MMCT is 0.57, and it pays more attention to the sentiment feature than the visual feature. In the last case, according to the review text *"It was very small in the waist,"* we can see that this user is not satisfied with this dress's waist. In the process of generating an explanation, on the one hand, the *small* feature of this dress can extract from the item image; on the other hand, the negative sentiment can be conveyed by the rating (it's 1.0); thus, the average value of β learned by MMCT is 0.45. Therefore, this shows that both the sentiment feature and visual feature can be beneficial to explanation generation.

In summary, our MMCT model can incorporate multimodal information reasonably, including sentiment features and visual features, to learn supervision signals to guide explanation generation. And the weight parameter β can be adjusted adaptively to generate reasonable explanations for different users. In this way, we can generate high-quality explanations to assist users in their decision-making.

VI. Conclusion

We proposed the MMCT model for personalized recommendation and natural language explanation generation, which incorporates multimodal information, including sentiment features, item features, item images, and refined user reviews, to generate high-quality explanations. During the decoding process, we proposed a dynamic fusion mechanism to induce multimodal information as supervision signals to guide the explanation generation. In addition, to learn discriminative and diverse sentences, we developed a contrastive objective for the training of the generation. Comprehensive experiments on two real-world datasets showed the effectiveness of our model in terms of explanation performance and recommendation performance. While an ablation analysis established the relative components, and the case study showed the working of our model from an intuitive sense. In the future, we expect to evaluate our model on other datasets to test its robustness and consider its fairness based on some sensitive domains involving valuable resource allocation, such as education, loan, and employment.

References

[1] X. Wang, Y. Chen, J. Yang, L. Wu, Z. Wu, and X. Xie, "A reinforcement learning framework for explainable recommendation," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2018, pp. 587–596.

[2] X. Chen, Y. Zhang, and Z. Qin, "Dynamic explainable recommendation based on neural attentive models," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 1, pp. 53–60.

[3] J. Gao, X. Wang, Y. Wang, and X. Xie, "Explainable recommendation through attentive multi-view learning," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 1, pp. 3622–3629.

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

[5] L. Li, Y. Zhang, and L. Chen, "Personalized transformer for explainable recommendation," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 4947–4957.

[6] G. Peake and J. Wang, "Explanation mining: Post hoc interpretability of latent factor models for recommendation systems," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 2060–2069.

[7] X. Li, W. Jiang, W. Chen, J. Wu, G. Wang, and K. Li, "Directional and explainable serendipity recommendation," in *Proc. Web Conf.*, Apr. 2020, pp. 122–132.

[8] K. Zhao et al., "Leveraging demonstrations for reinforcement recommendation reasoning over knowledge graphs," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 239–248.

[9] S.-J. Park, D.-K. Chae, H.-K. Bae, S. Park, and S.-W. Kim, "Reinforcement learning over sentiment-augmented knowledge graphs towards accurate and explainable recommendation," in *Proc. 15th ACM Int. Conf. Web Search Data Mining*, Feb. 2022, pp. 784–793.

[10] L. Li, Y. Zhang, and L. Chen, "Generate neural template explanations for recommendation," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2020, pp. 755–764.

[11] A. Yang, N. Wang, H. Deng, and H. Wang, "Explanation as a defense of recommendation," in *Proc. 14th ACM Int. Conf. Web Search Data Mining*, Mar. 2021, pp. 1029–1037.

[12] E. Dinan et al., "The second conversational intelligence challenge (ConvAI2)," in *The NeurIPS Competition*. Cham, Switzerland: Springer, 2020, pp. 187–208.

[13] Y. Zhang, H. Zhang, M. Zhang, Y. Liu, and S. Ma, "Do users rate or review? Boost phrase-level sentiment labeling with review-level sentiment classification," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2014, pp. 1–14.

[14] Y. Tao, Y. Jia, N. Wang, and H. Wang, "The FacT: Taming latent factor models for explainability with factorization trees," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2019, pp. 295–304.

[15] N. Wang, H. Wang, Y. Jia, and Y. Yin, "Explainable recommendation via multi-task learning in opinionated text data," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 165–174.

[16] Q. Ai, V. Azizi, X. Chen, and Y. Zhang, "Learning heterogeneous knowledge base embeddings for explainable recommendation," *Algorithms*, vol. 11, no. 9, p. 137, Sep. 2018.

[17] Y. Xian et al., "EX3: Explainable attribute-aware item-set recommendations," in *Proc. 15th ACM Conf. Recommender Syst.*, Sep. 2021, pp. 484–494.

[18] P. Li, Z. Wang, L. Bing, and W. Lam, "Persona-aware tips generation?" in *Proc. World Wide Web Conf.*, May 2019, pp. 1006–1016.

[19] J. Ni, J. Li, and J. McAuley, "Justifying recommendations using distantly-labeled reviews and fine-grained aspects," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 188–197.

[20] L. Li, Y. Zhang, and L. Chen, "Personalized prompt learning for explainable recommendation," 2022, *arXiv:2202.07371*.

[21] P. Li, Z. Wang, Z. Ren, L. Bing, and W. Lam, "Neural rating regression with abstractive tips generation for recommendation," in *Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Aug. 2017, pp. 345–354.

[22] Q.-T. Truong and H. Lauw, "Multimodal review generation for recommender systems," in *Proc. World Wide Web Conf.*, May 2019, pp. 1864–1874.

[23] Z. Liu, Y. Ma, Y. Ouyang, and Z. Xiong, "Contrastive learning for recommender system," 2021, *arXiv:2101.01317*.

[24] Z. Liu, Y. Ma, M. Hildebrandt, Y. Ouyang, and Z. Xiong, "CDARL: A contrastive discriminator-augmented reinforcement learning framework for sequential recommendations," *Knowl. Inf. Syst.*, vol. 64, no. 8, pp. 2239–2265, Aug. 2022.

[25] Z. Liu, Y. Ma, M. Schubert, Y. Ouyang, and Z. Xiong, "Multimodal contrastive pre-training for recommendation," in *Proc. Int. Conf. Multimedia Retr.*, Jun. 2022, pp. 99–108.

[26] Y. Yang, C. Huang, L. Xia, and C. Li, "Knowledge graph contrastive learning for recommendation," 2022, *arXiv:2205.00976*.

[27] R. Qiu, Z. Huang, H. Yin, and Z. Wang, “Contrastive learning for representation degeneration problem in sequential recommendation,” in *Proc. 15th ACM Int. Conf. Web Search Data Mining*, Feb. 2022, pp. 813–823.

[28] C. Zhou, J. Ma, J. Zhang, J. Zhou, and H. Yang, “Contrastive learning for debiased candidate generation in large-scale recommender systems,” in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2021, pp. 3985–3995.

[29] J. Wu et al., “Self-supervised graph learning for recommendation,” in *Proc. 44th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2021, pp. 726–735.

[30] W. Wei, C. Huang, L. Xia, Y. Xu, J. Zhao, and D. Yin, “Contrastive meta learning with behavior multiplicity for recommendation,” in *Proc. 15th ACM Int. Conf. Web Search Data Mining*, Feb. 2022, pp. 1120–1128.

[31] J. Yu, H. Yin, X. Xia, T. Chen, L. Cui, and Q. V. H. Nguyen, “Are graph augmentations necessary? Simple graph contrastive learning for recommendation,” 2021, *arXiv:2112.08679*.

[32] Y. Dong, J.-B. Cordonnier, and A. Loukas, “Attention is not all you need: Pure attention loses rank doubly exponentially with depth,” in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 2793–2803.

[33] Y. Su et al., “TaCL: Improving BERT pre-training with token-aware contrastive learning,” 2021, *arXiv:2111.04198*.

[34] A. Radford et al., “Learning transferable visual models from natural language supervision,” in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[35] J. D. M.-W. C. Kenton and L. K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.

[36] A. Mnih and R. R. Salakhutdinov, “Probabilistic matrix factorization,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 20, 2007, pp. 1–14.

[37] Y. Koren, “Factorization meets the neighborhood: A multifaceted collaborative filtering model,” in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2008, pp. 426–434.

[38] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “BLEU: A method for automatic evaluation of machine translation,” in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, 2001, pp. 311–318.

[39] C. Y. Lin, “ROUGE: A package for automatic evaluation of summaries,” in *Proc. Workshop Text Summarization Branches Out, Post Conf. Workshop ACL*, 2004, pp. 74–81.

[40] H. Steck, “Evaluation of recommendations: Rating-prediction and ranking,” in *Proc. 7th ACM Conf. Recommender Syst.*, Oct. 2013, pp. 213–220.

[41] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” 2014, *arXiv:1412.6572*.

[42] J. Chen, H. Zheng, H. Xiong, S. Shen, and M. Su, “MAG-GAN: Massive attack generator via GAN,” *Inf. Sci.*, vol. 536, pp. 67–90, Oct. 2020.

[43] H. Zheng, J. Chen, H. Du, W. Zhu, S. Ji, and X. Zhang, “GRIP-GAN: An attack-free defense through general robust inverse perturbation,” *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 6, pp. 4204–4224, Nov. 2022.

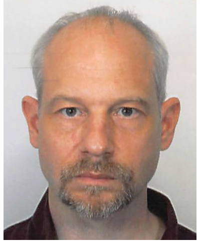
[44] X. He, Z. He, X. Du, and T.-S. Chua, “Adversarial personalized ranking for recommendation,” in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 355–364.

Zhuang Liu received the master’s degree from Beihang University, Beijing, China, in 2020, where he is currently pursuing the Ph.D. degree with the Laboratory of Engineering, Research Center of the Advanced Computer Application Technology, School of Computer Science.

His main research interests are recommender systems and network representation learning.

Yunpu Ma received the master’s degree in high-energy physics from the Ludwig-Maximilians-University of München, Munich, Germany, in 2016.

In 2016, he joined Siemens, München, as a Ph.D. Student of computer science. His current research is primarily centered on different topics of artificial intelligence and quantum machine learning. One topic is the cognitive perspective of semantic and episodic knowledge graphs. He is also working on quantum algorithms, causal inference, Natural Language Processing (NLP), and graph-related algorithms.



Matthias Schubert is a Professor of artificial intelligence at the Institute for Informatics, LMU Munich, Munich, Germany. He is one of the Founders of the Data Science Laboratory at LMU, Vice-Spokesperson of the Elite Program Master Data Science and one of the original PIs of the Munich Center for Machine Learning. He authored over 100 scientific publications being cited more than 3000 times. His research interests include machine learning for graphs, deep learning for spatial information systems, and sequential decision problems in spatiotemporal applications.

Yuanxin Ouyang received the B.Sc. and Ph.D. degrees from Beihang University, Beijing, China, in 1997 and 2005, respectively.

She is a Professor at Beihang University. Her area of research covers recommender systems, data mining, social networks, and service computing.

Wenge Rong received the B.Sc. degree from the Nanjing University of Science and Technology, Nanjing, China, in 1996, the M.Sc. degree from Queen Mary College, University of London, London, U.K., in 2003, and the Ph.D. degree from the University of Reading, Berkshire, U.K., in 2010.

He is a Professor with the School of Computer Science and Engineering, Beihang University, Beijing, China. He has many years of working experience as a Senior Software Engineer in numerous research projects and commercial software products. His area of research covers machine learning, natural language processing, and information management.

Zhang Xiong is a Professor with the School of Computer Science and Engineering, Beihang University, Beijing, China, and the Director of the Advanced Computer Application Research Engineering, Center of National Educational Ministry of China. He has published over 100 refereed papers in international journals and conference proceedings. His research interests and publications span from smart cities, knowledge management, information systems, intelligent transportation systems, and so on.

Dr. Xiong won the National Science and Technology Progress Award.