Using Review Text and External Knowledge for Explainable Recommendation

Peng Bai[1]*, Yuqi Tian[1], Xiudi Chen[1], Weixin Xie[2], Rui Lang[1]

[1]School of Informatics Xiamen University, China
[2]Medical College of Xiamen University, China
{baipeng,tianyuqi}@stu.xmu.edu.cn,{765890527,1105413061,814530456}@qq.com
31520210156933, 31520211154014, 31520211154028, 24520210157075, 31520211154016

Abstract

The current research of recommender system mainly focuses on improving the accuracy of the recommendation, but pays less attention to explainability. Explainability is essential to enhance users' trust and satisfaction, which can even increase the likelihood of buying items. In the existing explainable recommender systems, the mining of explicit and implicit features is not comprehensive, and the interaction among these features is not considered plentifully. In addition, the generated recommended reason text is not personalized and content-rich enough. It is necessary to improve the quality of recommended reason text because it is difficult to meet the needs of different users by using low-quality text. In this paper, we propose a new method that fuses aspect sentiment from review text and external knowledge to predict rating and generate personalized, content-rich recommended reasons text, which applies fine-tuning BERT to solve aspect-based sentiment analysis and extends Transformer to generate recommended reason text. The experiment results on real-world datasets demonstrated that our method was effective, and our model was superior to the baseline models on various metrics. For rating prediction task, our model can achieve an improvement of 0.6% on average in terms of RMSE. For recommended reason generation task, our model can achieve an improvement of 9.2% to 11.3% over state-of-the-arts in terms of BLEU.

Intorduction

With the rapid development of the Internet, people are enjoying the great convenience brought by the information era. Meanwhile, they are facing the troubles caused by information overload. The birth of the recommender system has largely made up for the shortcomings of search engine, and it can actively recommend items to users. At present, the recommender system has been widely used in various fields, such as e-commerce, news, movies, food, music, travel, etc.

Research on recommender system can generally be divided into collaborative filtering (CF) (Goldberg et al. 1992), content-based (Pazzani and Billsus 2007), and hybrid methods (Burke 2002). CF achieved further success after integrated with the Latent Factor Model (LFM). In many LFMs, Matrix Factorization (MF) (Koren, Bell, and Volinsky 2009) and its variants are particularly successful in rating prediction tasks. However, the latent factors in LFM have no intuitive meaning, which makes it difficult to understand why an item has a good prediction or why it is recommended. Some content-based recommender systems are weakly explainable. Over the years, many powerful neural network recommendation algorithms have been proposed (He and Chua 2017)(He et al. 2017)(Ma et al. 2018). These recommendation algorithms improved accuracy, but they do not pay attention to explainability, these models sometimes like black boxes, which are difficult to explain.

To make the recommendation model easier to understand, research on explainable recommender system have attracted the attention of industry and academia. An explainable recommender system can be defined as giving recommended reason while recommending items to users. Explainability can improve the transparency, persuasiveness and effectiveness of the recommender system, and it can also enhance users' trust, satisfaction, and even increase users' purchase rates (Rago, Cocarascu, and Toni 2018). In explainable recommender systems, both accuracy and explainability should be considered and optimized. Many methods mine users' reviews to better understand users/items and generate recommended reasons. However, in the existing explainable recommender systems, the features in mining reviews are insufficient, and the user-item feature interaction is not comprehensive. In addition, the recommended reasons are not personalized and informative enough to arouse users' interest. The quality of the recommended reasons need to be improved. In this paper, we mainly optimized the accuracy and explainability of recommendation.

To solve above problems, we designed a novel explainable recommendation model called Aspect Sentiment and External Knowledge for Explainable Recommendation (AKER), which improves the accuracy and explainability of the recommender system by fusing aspect sentiment features of reviews and external knowledge. Our main contributions are summarized as follows:

- We construct an explainable recommendation model, which can generate recommended reasons while recommending items. Explicit aspect features and users' opinions are extracted from reviews. We take full advantage of aspect features when predicting rating and generating recommended reasons.

*Corresponding Author

- We propose a novel recommended reason generation model, which uses bi-directional attention mechanism to effectively fuse aspect sentiment and external knowledge. This model improves the quality of recommended reason. In terms of implementation, we mainly extend Transformer model.
- We analyze and evaluate the results of the experiments. The results show that our method improves accuracy and explainability compared to the baseline models, meanwhile, can generate personalized, content-rich and high-quality recommended reason text.

The rest of this paper is organized as follows. The related works are introduced in Section II. The proposed method is introduced in Section III. The extensive experiments and overall results are discussed in Section IV. Finally, we conclude our work in Section V.

Related Work

The existing explainable recommender systems are mainly mining review text information to make rating accurate or enhance explainability. In the presentation form, there are recommended reason labels or recommended reason text. Hidden Factors and Hidden Topics (HFT) model (McAuley and Leskovec 2013) combines product ratings with review text. HFT aligns hidden factors in product ratings with hidden topics in product reviews and these topics are used to identify useful and representative reviews. In terms of implementation, MF and Latent Dirichlet Allocation (LDA) are used to predict product ratings. However, HFT can not generate recommended reason text. In order to solve the problem that the latent features of LFM is difficult to explain the recommendation results to users, Explicit Factor Model (EFM) (Zhang et al. 2014) extracts clear user opinions and explicit item features about all aspects of the item from the reviews. EFM fills the aspect words into the designed template for intuitive explanation.

Some researchers turn the recommendation problem into a graph or tree problem. A generic algorithm for ranking on tripartite graphs (TriRank) (He et al. 2015) extracts aspects from reviews to construct a use-item-aspect ternary relation. These relationships are modeled as heterogeneous three-party graph, so recommendation task becomes one of the vertex rankings issue. Because knowledge graphs can provide rich structured information, some works (Zhang et al. 2016)(Yang et al. 2018)(Ai et al. 2018) have also begun to combine the recommender system with knowledge graph to enhance explainability. A method called Policy-Guided Path Reasoning (PGPR) (Xian et al. 2019) conducts explicit reasoning with knowledge for decision making so that the recommended reason are generated and supported by an explainable causal inference procedure. PGPR couples recommendation and explainability by providing actual paths in a knowledge graph. Building regression trees (Tao et al. 2019) on users and items respectively from user-generated reviews are used to enhance explainability. With the growth of regression tree, the latent factors are gradually refined under the regularization imposed by the tree structure. As a result, model can track the creation of latent profiles by looking into the path of each factor on regression trees.

Explainable recommendation usually optimizes the accuracy and explainability. Some studies divide this problem into recommendation task and explanation task and use multi-task learning methods to optimize two tasks. Deep Cooperative Neural Networks (DeepCoNN) (Zheng, Noroozi, and Yu 2017) combines Convolutional Neural Networks (CNN) with Factorization Machine (FM) (Rendle 2010) to make rating prediction based on the review text. DeepCoNN consists of two parallel neural networks. One network focuses on learning users' behavior from reviews, and another network learns items' attributes from reviews. A model named NRT (Li et al. 2017) applies deep neural networks to predict rating and generate item abstract tips in e-commerce field. They use improved LFM to predict ratings and apply Gate Recurrent Unit (GRU) to translate potential representations of users and items into concise tips. Some studies have introduced attention mechanism into the recommendation model. Deep Explicit Attentive Multi-View Learning (DEAML) model (Gao et al. 2019) can accurately predict ratings, infer multi-level user profiles and solve the problem of constrained tree node selection through dynamic programming algorithm. Moreover, DEAML can generate personalized explanations from multi-level functions. Co-Attentive Multi-task Learning (CAML) model (Chen et al. 2019b) makes full use of the correlation between the recommendation task and the explaination task. CAML is composed of encoder-selector-decoder architecture. The selector is a multi-pointer co-attention selector, which can effectively control the cross information transfer of two tasks.

For the existing models, the users' preference is usually regarded as a static explanation, but there are still some shortcomings. So Dynamic Explainable Recommender (DER) system (Chen, Zhang, and Qin 2019) appeared, which makes rating prediction accurate and enhances explainability. In DER, a time-aware GRU is used to model user dynamic preferences and CNN is used to analyze the information of item review. In addition, In explainable recommender systems, data mining of review text is very important. One of the methods is Aspect-Based Sentiment Analysis (ABSA), which can get fine-grained information of the user or item. Based on predecessor works, We consider digging out explicit aspect sentiment information from the review text, and fusing it with external knowledge to enhance accuracy and explainability of the recommender system in this paper.

METHOD

Overview

In this section, we describe details of the method used in our model. As shown in Figure 1, our model AKER consists of two parts, one is the rating prediction and the other is the recommended reason generation. A basic work of these two parts is ABSA. we apply fine-tuning BERT to get the aspect and aspect sentiment polarity from the reviews. Translation-based Factorization Machines (TransFM) (Pasricha and McAuley 2018) is used to predict rating. TransFM

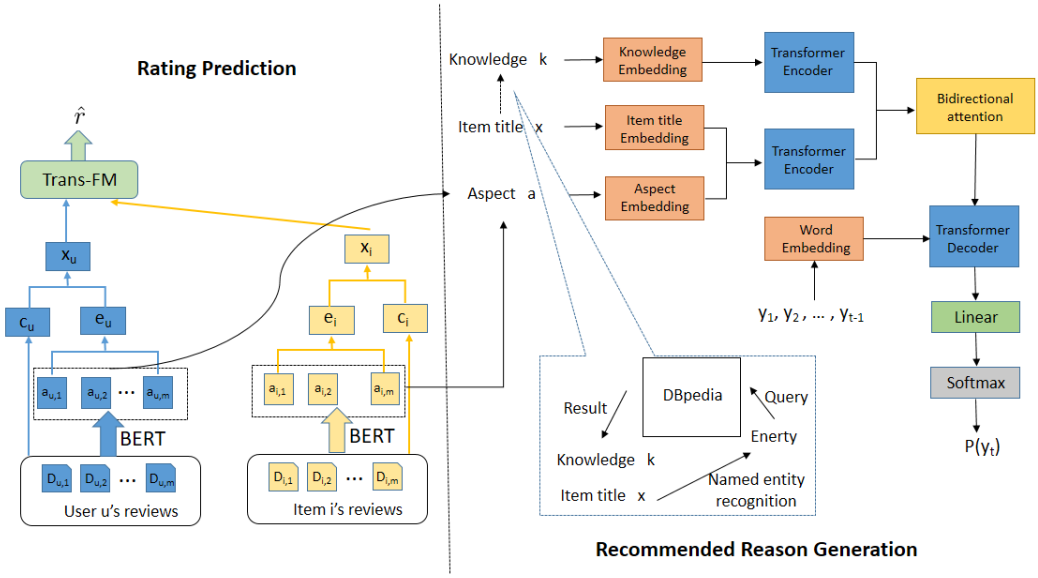

Figure 1: The structure of our proposed model for rating prediction and recommended reason generation.

combines translation and metric-based approaches for sequential recommendation. In the task of recommended reason generation, we introduce the basic Transformer encoder-decoder model, aspect fusion, and knowledge fusion. Aspect fusion is responsible for fusing aspects and the item title. After aspect fusion, knowledge fusion is responsible for fusing relevant knowledge obtained from the knowledge graph. So personalized and content-rich recommended reason can be generated through these two fusions.

Rating prediction

The model input contains user set U, the item set I, the review set D, the aspect a, the knowledge k and item title x. We have obtained the explicit aspect features $\mathbf{a}_{u,i}$, $\mathbf{a}_{i,j}$ from the reviews by using BERT. Then we use them to make the embeddings $\mathbf{e}_u$ and $\mathbf{e}_i$. We also consider the implicit features representation $\mathbf{c}_u$ and $\mathbf{c}_i$ by using a Generative Feature Language Model (GFLM) (Karmaker Santu, Sondhi, and Zhai 2016) to mine implicit features from reviews with a general and unsupervised manner. GFLM is based on statistical learning and automatically optimizes parameters through the expectation maximization algorithm. Both explicit and implicit features are considered. The embeddings of user and item are represented as $\mathbf{x}_u = [\mathbf{e}_u; \mathbf{c}_u]$ and $\mathbf{x}_i = [\mathbf{e}_i; \mathbf{c}_i]$, where $[;]$ is just a splicing symbol. We use TransFM to predict ratings. Specifically, TransFM learns the embedding and translation space of each feature dimension. It replaces the inner product with the squared Euclidean distance to measure the interaction strength between features. The prediction rating $\hat{r}$ is expressed as follows:

$$\hat{r} = y(\mathbf{x}) = \omega_0 + \sum_{i=1}^{n} \omega_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} d^2(\mathbf{v_i} + \mathbf{v_i^{'}}, \mathbf{v_j}) x_i x_j. \tag{1}$$

where $\omega_0 \in R$ is a global bias, $x_i \in R$ is the i-th feature of $\mathbf{x} = [\mathbf{x}_u, \mathbf{x}_i]$. $\omega_i \in R$ is the linear term for feature x_i. x_j is similar to x_i and it is a representation of another feature. $\mathbf{v_i} \in R^k$ is the embedding vector for feature x_i. Similarly, $\mathbf{v_j} \in R^k$ is the embedding vector for feature x_j, and $\mathbf{v_i^{'}} \in R^k$ is the translation vector for feature x_i. $d^2(\mathbf{v_i} + \mathbf{v_i^{'}}, \mathbf{v_j})$ represents the squared Euclidean distance between the vectors $\mathbf{v_i} + \mathbf{v_i^{'}}$ and $\mathbf{v_j}$

$$d^2(\mathbf{v_i} + \mathbf{v_i^{'}}, \mathbf{v_j}) = (\mathbf{v_i} + \mathbf{v_i^{'}} - \mathbf{v_j}) \cdot (\mathbf{v_i} + \mathbf{v_i^{'}} - \mathbf{v_j}). \tag{2}$$

The loss function of rating prediction is expressed as follows:

$$\pounds_r = \frac{1}{2\,|\Omega|} \sum_{(u,i)\epsilon\Omega} (\hat{r} - r)^2. \tag{3}$$

where Ω is the training set, $\hat{r}$ is the prediction rating and r is the ground truth rating.

Recommended reason generation

Recommended reason generation is another important part, which is used to generate recommended reason text. In this process, we improved the Transformer model proposed in literature (Vaswani et al. 2017). Our goal is to generate personalized, content-rich, high-quality recommended reason by fully fusing aspect, item title and external knowledge. Transformer is the seq2seq model proposed by Google Brain, and has now achieved a wide range of applications. It is an encoder-decoder framework and has the advantages of parallel computing, low computational complexity and high model explainability. So we do not use sequences based on recurrent neural networks.

Aspect Fusion In the past the recommended reason generation was general and lacked personalization. To solve this problem, we take aspect as the input of the Transformer

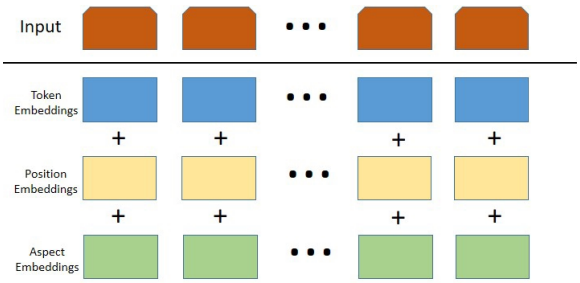

Figure 2: The input embeddings of the model are the sum of the item title token embeddings, the position embeddings and the aspect embeddings

.

model. In this way, the focus of recommended reason generation by different users is different, which is more personalized and user-friendly. Chen *et al.* (Chen et al. 2019a) have used Transformer to complete the task of generating product descriptions by adding attributes and knowledge. The generation effect is very good and effective. So we continue firmly to study in this direction. Specifically, in addition to the item title, we add aspect extracted from the reviews. The training of this step can be expressed as follows:

$$P(y|x,a) = \prod_{t=1}^{m} P(y_t|y_1, y_2, \ldots, y_{t-1}, x, a). \quad (4)$$

where x is the item title, a is item aspect, y is the output sequence, and t is the decoding time step. The fusing representation is illustrated in Figure 4. We embed the input item title $x = (x_1, x_2, \ldots, x_n)$, and get $e = (e_1, e_2, \ldots, e_n)$. The core of aspect fusion is that aspect embedding is added to e_i at each time stamp.

Knowledge Fusion We introduce content fusion in our model. Now, the aspect and its polarity, item title, knowledge have been extracted. Then we use attention mechanisms to combine them. In the process, the BI-Directional Attention Flow (BIDAF) network (Seo et al. 2016) is used in our model. Chen *et al.* (Chen et al. 2019a) did similar work, but only product descriptions are generated. The BIDAF network include six layers, which are Character Embedding Layer, Word Embedding Layer, Contextual Embedding Layer, Attention Flow Layer, Modeling Layer and Output Layer. We focus on the Attention Flow Layer, because in this layer we construct the bi-directional attention: item title-to-external knowledge attention and external knowledge-to-item title attention.

Attention Flow Layer is important for BIDAF network. This layer is in charge of linking and fusing information in context and query words. The input of this layer is the context vector representation of context H and query U. In our work, H is the high-level representation that fuses item title x and aspect a. U is the high-level representation of external knowledge k. The output of this layer is the query aware vector representation of context words G and the context embedding of the previous layer. The key to above two attentions is shared similarity matrix S. $S \in R^{n*u}$ is the result of the interaction of $H \in R^{n*d}$ and $U \in R^{u*d}$. n is the number of the title words, u is the number of the knowledge words and d is the dimension of the every word. The similarity matrix is expressed as follows:

$$S_{tj} = \alpha(H_{:t}, U_{:j}) = w_S^{\mathrm{T}}[H_{:t}; U_{:j}; H \circ U]. \quad (5)$$

where S_{tj} represents the similarity between the t-th title word and the j-th knowledge word. α represents the function that encodes the similarity between the two input vectors. $H_{:t}$ is the t-th column vector of H. $U_{:j}$ is the j-th column vector of U. $w_S \in R^{3d}$ is a trainable weight vector. $\circ$ is elementwise multiplication. $[;]$ is vector concatenation across row and multiplication is matrix multiplication.

Item title-to-external knowledge Attention represents which knowledge words are most relevant to item title word. Let $a_t \in R^u$ represents the attention weights on the knowledge words by the t-th title word, $\sum_j a_{tj} = 1$ for all t. The attention weight is computed by $a_t = \mathrm{softmax}(S_{t:})$, and subsequently each attended knowledge vector is $\widetilde{U_{:t}} = \sum_j a_{tj} U_{:j}$. Hence $\widetilde{U} \in R^{n*d}$ contains the attended knowledge vectors for the entire title.

External knowledge-to-item title Attention represents which item title words have the closest similarity to knowledge word. We obtain the attention weights on the title words by $b = \mathrm{softmax}(max(S_{t:}))$, where the maximum function (max) is performed across the column. Then the attended title vector is $\widetilde{h} = \sum_t b_t H_{:t}$. This vector represents the weighted sum of the most important words in the title with respect to the knowledge. $\widetilde{h}$ is tiled n times across the column, thus giving $\widetilde{H} \in R^{n*d}$.

Finally, we get G by combining the contextual embeddings with the attention vectors, and each column vector can be considered as the query-aware representation of each context word. G can be expressed as follows:

$$G_{:t} = \beta(H_{:t}, \widetilde{U_{:t}}, \widetilde{H_{:t}}) = [H_{:t}; H_{:t} \circ \widetilde{U_{:t}}; H_{:t} \circ \widetilde{U_{:t}}] \quad (6)$$

where $G_{:t} \in R^{d_G}$, $G_{:t}$ is the t-th column vector. β is a trainable vector function that integrates its input vectors, and d_G is the output dimension of the β function.

Experiments

In this section, experiments have been carried out to evaluate our proposed model. Firstly, we introduce the datasets, the baseline model for comparison, and the evaluation metrics. Secondly, we present experimental results through a series of evaluation metrics and analyze them.

Datasets

In the experiments, we use the public datasets of Yelp and Amazon. The ratings of these datasets are all integers between [1, 5]. Specifically, we choose Yelp Challenge 2016 [1], which contains 684,295 users, 85,901 items, and 2,730,103 reviews.

The other dataset (i.e. Amazon dataset) contains item reviews and metadata from Amazon, which includes 142.8

[1] https://www.yelp.com/dataset/challenge

million reviews from May 1996 to July 2014. We actually choose "Small" subsets 5-core: Electronics and Health & Personal Care [2] for experiment. 5-core means that these data have been reduced to extract the 5-core, and each user and product has 5 reviews. Electronics contains 192,415 users, 63,125 items, and 1,689,188 reviews. Health & Personal Care contains 82,640 users, 33,368 items, and 1,131,687 reviews.

Models for Comparison

We introduce the baseline models for comparison in this subsection. In order to evaluate the accuracy of rating prediction, we compared our model with the following models: Probabilistic Matrix Factorization(PMF) (Mnih and Salakhutdinov 2008), SVD++ (Koren 2008), NRT, DEAML, CAML, DER. In order to evaluate the explainability of the recommendation results, we compared our model with other generation models: EFM (Zhang et al. 2014), DEAML, NRT, DER CAML.

Evaluation Metrics

In order to evaluate the accuracy of the rating prediction, we use Root Mean Square Error (RMSE). RMSE can be expressed as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{u,i}(\hat{r}_{u,i} - r_{u,i})} \tag{7}$$

where N is the number of ratings between users and items, $\hat{r}_{u,i}$ is the predicted rating and $r_{u,i}$ is the ground truth rating.

In order to evaluate the explainability. Bilingual Evaluation Understudy(BLEU) (Papineni et al. 2002) and Recall-Oriented Understudy for Gisting Evaluation(ROUGE) (Lin and Hovy 2003) are regarded as the objective evaluation metrics.

BLEU is mainly based on precision. The higher the BLEU score is, the better the generation quality is. BLEU is defined as follows:

$$BLEU = BP \exp(\sum_{n=1}^{N} W_n \log P_n) \tag{8}$$

$$BP = \begin{cases} 1, & l_c > l_r \\ \exp(1 - lr/lc), & l_c \leq l_r \end{cases} \tag{9}$$

where BP is the brevity penalty, lc is the length of the generated recommended reason, and lr is the length of the shortest user review on the item. BLEU needs to calculate the 1-gram, 2-gram, ..., N-gram precision of the generated text. P_n refers to the precision of N-gram and W_n refers to the weight of N-gram.

ROUGE-N is mainly based on recall.

$$ROUGE - N = \frac{\sum_{g_n \in s} C_m(g_n)}{\sum_{g_n \in s_g} C(g_n)} \tag{10}$$

where g_n is N-gram, $C(g_n)$ is the number of n-grams in s. $C_m(g_n)$ is the number of n-grams co-occurring in s_g and s.

[2] http://jmcauley.ucsd.edu/data/amazon

	Electronics	Health & Personal Care	Yelp Challenge 2016
PMF	1.609	1.267	1.743
SVD++	1.185	1.035	1.326
NRT	1.102	0.989	1.265
DEAML	1.092	0.982	1.231
CAML	1.074	0.968	1.167
DER	1.045	0.943	1.126
OURS	**1.039**	**0.936**	**1.119**

Table 1: RMSE of rating prediction results in our model and comparative models.

Result Analysis

Rating prediction Table 1 shows the rating prediction results of our model and comparative models. Our model consistently outperforms all comparison methods under the RMSE metric on all datasets. This is because PMF and SVD++ only take the rating matrix as input, NRT and CAML just consider the features in the review text. In addition, DEAML optimizes the importance and relevance of nodes in the hierarchy, but the nodes may not be comprehensive. DER learns important review information, but misses some feature information. Our model is more comprehensive and effective in the representing of features, which are obtained from the review text. Also, implicit features and feature interactions are considered in our model. So our model consistently achieves the highest accuracy on all three datasets and averages 0.6% better than DER.

Explainability of the recommendation results Our model not only solves the problem of rating prediction, but also generates recommended reasons for users. Table 2 shows the recommended reason generation results of our model and baseline models. In terms of BLEU and ROUGE, our model consistently outperforms the baseline models on different datasets. Taking BLEU as an example, our model is 9.2% to 11.3% higher than the state-of-the-art method CAML. The results illustrate the effectiveness of our encoder-decoder generation model fusing knowledge and aspect sentiment. Our method always outperforms CAML, because it can learn deep user-item interactions, extract detailed aspect-level features, and incorporate external knowledge as a supplement. To summarise, our model generates personalized and content-rich recommended reason, which improves explanation quality.

Conclusion

An explainable recommendation model called AKER is designed in this paper, which makes full use of the aspect sentiment information in the reviews and can predict rating accurately and generate personalized, content-rich, high-quality recommended reason simultaneously. Aiming at the generation of recommendation reasons, a recommended reason generation model is proposed by using bi-directional attention mechanism to effectively fuse item title, aspect and external knowledge. Experiment results show that our model is superior to state-of-the-art baselines on both the accuracy and explainability. In the future, we will consider adding social relationships to the explainable recommender system.

Datasets	Metrics	EFM	DEAML	NRT	DER	CAML	**Ours**	**Improvement(%)**
Electronics	BLEU	1.21	1.23	1.33	1.45	1.97	**2.19**	**+11.3%**
	ROUGE-1	15.86	16.02	17.39	18.41	19.26	**19.74**	**+2.5%**
	ROUGE-2	3.39	3.43	3.50	3.62	3.81	**3.89**	**+2.1%**
	ROUGE-L	15.01	15.25	15.71	15.99	16.75	**16.95**	**+1.2%**
	ROUGE-SU4	5.18	5.43	5.97	6.13	6.47	**6.62**	**+2.3%**
Health & Personal Care	BLEU	1.56	1.57	1.60	1.73	2.04	**2.23**	**+9.2%**
	ROUGE-1	17.81	17.93	18.09	18.65	19.32	**19.76**	**+2.3%**
	ROUGE-2	4.25	4.27	4.30	4.36	4.58	**4.68**	**+2.2%**
	ROUGE-L	15.59	15.87	16.01	16.32	16.69	**16.94**	**+1.5%**
	ROUGE-SU4	6.03	6.18	6.29	6.43	6.71	**6.86**	**+2.3%**
Yelp Challenge 2016	BLEU	1.17	1.23	1.31	1.47	1.58	**1.73**	**+9.7%**
	ROUGE-1	12.48	12.67	13.31	13.85	14.24	**14.70**	**+3.2%**
	ROUGE-2	2.57	2.72	3.05	3.22	3.50	**3.73**	**+6.5%**
	ROUGE-L	11.32	11.76	12.13	12.35	12.90	**13.25**	**+2.7%**
	ROUGE-SU4	4.13	4.35	4.60	4.74	5.03	**5.24**	**+4.2%**

Table 2: Evaluate generated recommended reason in terms of BLEU and ROUGE.

Reference

Pazzani, M. J.; and Billsus, D. 2007. *Content-Based Recommendation Systems*, 325–341. Berlin, Heidelberg: Springer-Verlag. ISBN 9783540720782.

Goldberg, D.; Nichols, D.; Oki, B. M.; and Terry, D. 1992. Using Collaborative Filtering to Weave an Information Tapestry. *Commun. ACM*, 35(12): 61–70

Burke, R. 2002. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4): 331–370

Koren, Y.; Bell, R.; and Volinsky, C. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer*, 42(8): 30–37

He, X.; Liao, L.; Zhang, H.; Nie, L.; and Chua, T. S. 2017. Neural Collaborative Filtering. *International World Wide Web Conferences Steering Committee*

Ai, Q.; Azizi, V.; Chen, X.; and Zhang, Y. 2018. Learning heterogeneous knowledge base embeddings for explainable recommendation. *Algorithms*, 11(9): 137.

Chen, X.; Zhang, Y.; and Qin, Z. 2019. Dynamic Explainable Recommendation Based on Neural Attentive Models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33: 53–60.

Liu, P.; Zhang, L.; and Gulla, J. A. 2019. Dynamic attention-based explainable recommendation with textual and visual fusion. *Information Processing and Management*, 57(6).

He, X.; and Chua, T.-S. 2017. Neural Factorization Machines for Sparse Predictive Analytics. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, 355–364. New York, NY, USA: Association for Computing Machinery. ISBN 9781450350228.

Ma, W.; Zhang, M.; Wang, C.; Luo, C.; and Liu, Y. 2018. Your Tweets Reveal What You Like: Introducing Cross-media Content Information into Multi-domain Recommendation. In *Twenty-Seventh International Joint Conference on Artificial Intelligence IJCAI-18*.

Rago, A.; Cocarascu, O.; and Toni, F. 2018. Argumentation-based recommendations: Fantastic explanations and how to find them. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 1949–1955.

McAuley, J.; and Leskovec, J. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, 165–172.

Zhang, Y.; Lai, G.; Zhang, M.; Zhang, Y.; Liu, Y.; and Ma, S. 2014. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, 83–92.

He, X.; Chen, T.; Kan, M.-Y.; and Chen, X. 2015. Trirank: Review-aware explainable recommendation by modeling aspects. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 1661–1670.

Zhang, F.; Yuan, N. J.; Lian, D.; Xie, X.; and Ma, W.-Y. 2016. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 353–362.

Xian, Y.; Fu, Z.; Muthukrishnan, S.; De Melo, G.; and Zhang, Y. 2019. Reinforcement knowledge graph reasoning for explainable recommendation. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, 285–294.

Tao, Y.; Jia, Y.; Wang, N.; and Wang, H. 2019. The fact: Taming latent factor models for explainability with factorization trees. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 295–304.

Zheng, L.; Noroozi, V.; and Yu, P. S. 2017. Joint deep mod-

eling of users and items using reviews for recommendation. In *Proceedings of the tenth ACM international conference on web search and data mining*, 425–434.

Rendle, S. 2010. Factorization machines. In *2010 IEEE International conference on data mining*, 995–1000. IEEE.

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

Chen, C.; Zhang, M.; Liu, Y.; and Ma, S. 2018. Neural attentional rating regression with review-level explanations. In *Proceedings of the 2018 World Wide Web Conference*, 1583–1592.

Gao, J.; Wang, X.; Wang, Y.; and Xie, X. 2019. Explainable recommendation through attentive multi-view learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 3622–3629.

Chen, Z.; Wang, X.; Xie, X.; Wu, T.; Bu, G.; Wang, Y.; and Chen, E. 2019b. Co-attentive multi-task learning for explainable recommendation. In *Twenty-Eighth International Joint Conference on Artificial Intelligence IJCAI-19*.

Devlin, J.; Chang, M. W.; Lee, K.; and Toutanova, K. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

Pasricha, R.; and McAuley, J. 2018. Translation-Based Factorization Machines for Sequential Recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, 63–71. New York, NY, USA: Association for Computing Machinery. ISBN 9781450359016.

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

Chen, Q.; Lin, J.; Zhang, Y.; Yang, H.; Zhou, J.; and Tang, J. 2019a. Towards Knowledge-Based Personalized Product Description Generation in E-Commerce. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery amp; Data Mining*, KDD '19, 3040–3050. New York, NY, USA: Association for Computing Machinery. ISBN 9781450362016.

Seo, M.; Kembhavi, A.; Farhadi, A.; and Hajishirzi, H. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Mnih, A.; and Salakhutdinov, R. R. 2008. Probabilistic matrix factorization. In *Advances in neural information processing systems*, 1257–1264.

Koren, Y. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 426–434.

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

Lin, C.-Y.; and Hovy, E. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, 150–157.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.