Review-aware Explainable Recommendation System with Aspect Matching

Siwei MA † Fan MO ‡ and Hayato YAMANA §

† ‡ Graduate School of Fundamental Science and Engineering 3-4-1 Okubo, Shinjuku-ku, Tokyo, 169-8555 Japan

§ Faculty of Science and Engineering, Waseda University 3-4-1 Okubo, Shinjuku-ku, Tokyo, 169-8555 Japan

E-mail: † ‡ § {lulumsw, bakubonn,yamana}@yama.info.waseda.ac.jp

Abstract An explainable recommendation system is one kind of recommendation system that provides a recommendation and explains why the system recommends this item to a target user. Existing works on generating the explanation usually focus on generating template-based explanations or neglect aspects of reviews that the target user prefers. In this paper, we assume that each sentence in the review will mention one aspect of the item's features, and the users would mention the aspects they are concerned about more than unconcerned aspects. Based on the above observations, we propose a textual explanation generation method for the recommendation system, which extracts the aspect of review sentences by unsupervised clustering and generates explanations based on the sentences whose aspect matches the target user's concern. Evaluation results show that our proposed method can generate explanations 8.2% more similar to the user's real reaction to the item than popular-based explanation.

Keyword Explainable recommendation system, Personalization, Aspect Matching

1. Introduction

Recommendation systems have been wildly used in the internet service industries to provide personalized recommendations for users. Under the data explosion background in recent years, this technology can filter the contents that the system predicts users would like from many items. On the one hand, recommendation systems enable companies to show more target-accurate commodities to users to maximize the benefit. On the other hand, users can find their preferred goods more efficiently, which increases their satisfaction with the web service.

In the field of recommendation systems, explainable recommendation systems let system engineers and users understand why such an item is recommended to them. It solves the problem of "why" by providing recommendations and providing the explanation with the recommendation. Specifically, providing explanations can enhance the system's transparency and effectiveness and help the user make the decision faster [1]. In recent years some e-commerce companies have already launched the explainable system, such as Amazon and JD.com. The online experiment on JD.com shows that the recommendation with an explanation can improve recommendation persuasiveness and conversion rate [2].

At present, the research of explainable recommendation systems can be classified into two categories: the recommendation of explainable items and the generation of explanation to recommendation items. Research in the first category [3][4] focuses on devising the interpretable models to ensure the explainability of recommendation. In contrast, the latter research category [2][5][6][7] only concentrates on the result of recommendation and how to generate the explanation of the result better, ignoring the recommendation progress. The recent interest towards the generation of explanation is to generate textual explanation based on user reviews, such as [8][9][10]. However, there is a deficiency of the sentence-level explanation that focuses on the sentences' information aspects. For example, Chen's work[11] presented a method to show users good reviews as an explanation, which contains many aspects of the item's information. Another example is Zhang's work[9], which extracted the aspects' words and put them into the template. This kind of template-level explanation cannot provide enough information in limited words.

In this paper, we fill in the gap of aspect-based sentence-level explanation. We propose an aspect matching textual explanation generation system that exploits users' interesting aspects by analyzing their history reviews and generates the explanation based on the reviews that correspond to the user's interesting aspects of the recommended items. In this way, we provide users with more personalized explanations, thus increase their satisfaction with the recommendation system.

To implement our idea, we first use an unsupervised neural attention model to extract the aspects from reviews and cluster the reviews by aspects. After aspect extraction,

the user's favorite aspect is defined by selecting the most frequently mentioned aspects. The reviews of the recommended item are extracted, and the sentences that match with the user's favorite aspect are ranked. Finally, an explanation is generated based on the ranked sentences.

The paper's outline is as follows: In Section 2, explainable recommendation systems and related work are introduced. Our proposed system is explained in Section 3. The experiment and results are shown in Section 4. Finally, we conclude in Section 5.

2. Related work

There are several different styles of explanation for a recommended item. In the early stage of research for recommendation systems, a user-based or item-based explanation is easy to generate with a collaborative filtering recommendation system. The recommendation itself is based on a similar user or item. Herlocker et al. [5] developed a system to explain the result of an automatic collaborative filtering system and compared some different display styles of explanation, including showing the histogram of similar users' (neighbors)' ratings to the target user. The evaluation result showed that this kind of explanation improves the acceptance of the recommendations. While in Sarwar et al. [12] 's research, the explanation in item-based collaborative filtering recommendation system could be provided by telling users the recommendation item is similar to the item they chose before. However, users may not understand the relationship between the recommended item and the explained item.

Compare to user-based or item-based explanation, the feature-based explanation can show the contents of the item more directly. Vig et al. [6] presented the explanation using the item's tags and how much the user shows interest. The authors also conducted a user study experiment to show the tag explanation improves the effectiveness of recommendation. Like this, Tintarev [7] provided a prototype recommendation system that provides explanations using predefined categories information. Nonetheless, the feature-based explanation requires a tag-include dataset or predefined categories information, which may not be available for most datasets.

Since more and more internet service users rate and comment on e-commerce websites, textual explanations using the review resource have become popular in recent years. This explanation can be classified into two groups, aspect-level explanation and sentence-level explanation [8].

Aspect-level explanations show users the aspect words about the item, which is similar to feature-based explanation. Nevertheless, the difference is that aspect-level explanation does not use the direct information (tags or categories) of the item; instead, it uses the textual reviews' information. For instance, Zhang et al. [2] presented an explainable recommendation system that provides a word cloud that shows the pair of aspect words and corresponding opinion words. Both aspect word and opinion word are extracted from previous user reviews. Wu and Ester [13] also present the word cloud style of explanation using the extracted words based on latent topic modeling. Though the word cloud style of explanation is easy to generate, it cannot provide the persuasive explanation that users can understand intuitively.

For sentence-level explanation, Zhang et al. [9] constructed the textual explanation, which puts the extracted feature word into a template. This template-based method ensured the readability of the explanation but cannot provide enough information in limited words. Chang et al. [10] used crowd service to generate semi-template personalized explanations, putting the selected aspect labels into templates and using the sentences from previous reviews. However, it used crowdsourcing, which cannot automatically generate explanation sentences. Chen et al. [11] use deep cooperative neural networks to analyze the textual reviews and calculate each review's helpfulness. Helpful reviews can be provided as review-level explanations of recommendation. Nevertheless, this kind of review-level explanation may contain redundant information or a negative attitude toward the item. Besides, this kind of review-level explanation is not personalized for a different user. In conclusion, recent works about sentence-level explanation cannot generate personalized explanation with rich information from previous reviews. Our work focused on this problem and tried to propose a model to solve that.

3. Proposed method

In this work, we propose a generation system that produces personalized sentence-level explanation, which automatically generates the explanation for a recommendation based on reviews and the target user's interesting aspect. Our methodology combines an unsupervised neural attention model to extract the aspects

and a generation method that can explain the user's most interesting aspect.

This section will introduce the explanation generation system by presenting the aspect extraction model and the generation method separately. We use an unsupervised neural attention model to extract the aspects from reviews and cluster the reviews by aspects. Then we generate the explanation by analyzing the target user's preference and using the reviews of the target item.

3.1. Aspect extraction and review clustering

We use an unsupervised aspect-extracting model, Attention-based AspectExtraction (ABAE) produced by He et al. [14] to extract the aspects. In this model, we adopt word embeddings in the review sentences to construct the sentence embeddings. An attention mechanism filters out the non-aspect related words. After calculating the sentence embeddings, we re-construct the sentence embedding as a linear combination of all aspect embeddings. This unsupervised learning process's training goal is to keep the least possible distortion of the original sentence embeddings and re-constructed vectors. After the training progress, the aspect set A is calculated, where the size of A is K.

For each sentence s in the training set, the model calculates a vector $\overrightarrow{prob} \in R^K$ to present the probability of the sentence belongs to the aspects, i.e., $prob_i$ shows the probability that the sentence s belongs to the i-th aspect A_i. As shown in Eq.1, the aspect with the highest probability is assigned as the aspect a_s that the sentence belonged to. We can cluster all the sentences into K clusters according to their aspects. Each review sentence belongs to an aspect after review clustering.

$$a_s = \arg max_i (prob_i) \quad \text{(Equation 1)}$$

3.2. Explanation extraction with aspect matching

In this work, we assume that users mention the aspects they are concerned about more often than unconcerned aspects, and each user has different concerned aspects. Thus, by counting one user's mentioned frequency, we can get the user's preference tendency, which is the most frequently mentioned aspect.

To formulate the problem, let U as the set of all users and M as the set of all items. Each review that user u commented for item m is expressed as r_{um}. Note that each review r_{um} may include one or several sentences $r_{um} = (s_1, s_2, \dots)$. We have a target user $u \in U$ and the recommendation item $m \in M$ that needs to be explained. Firstly, we extract sentences from user u's previous review set $PR_u = \{r_{um} | m \in M\}$ in the training set. The belonging aspects of each sentence in the reviews are also extracted. Then we could count the appearance frequency of each aspect, the appearance frequency of aspect i is shown as f_{u_i}. After that, user u's most interested aspect a_u is selected by finding the aspect that has the highest appearance frequency:

$$a_u = \arg max_i (f_i) \quad \text{(Equation 2)}$$

After that, we extract the sentences from item m's history reviews $PR_m = \{r_{um} | u \in U\}$. The sentences of the reviews with the same aspect as a_u make up the candidate sentence set, where each sentence s in $Cand_{um}$ followed the Eq. 3.

$$Cand_{um} = \{s | \exists s \in \{r | r \in PR_m\}, a_s = a_u\} \quad \text{(Equation 3)}$$

Fig. 1 shows an example process of extracting the candidate sentence set.

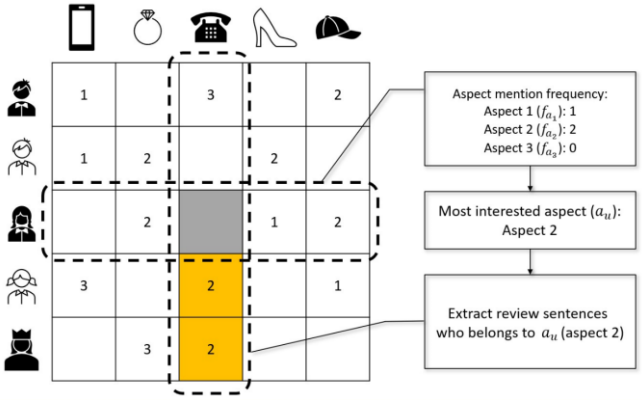

Figure 1 Process of extracting candidate sentence set, where the number in the block shows the belonging aspect of the review that the user gave to the item. The grey block is the target need to generate an explanation and the yellow blocks makeup the candidate sentence set.

To generate the explanation, we need to rank the sentences in $Cand_{um}$. We choose to rank the sentences by their helpfulness since reviews with more helpfulness tend to be more persuasive, increasing their satisfaction. The helpfulness is defined by the helpfulness votes for a review in the dataset. After that, the top l_u sentences form the final explanation, where the number of sentences in the explanation l_u is set to the average length of the user u's previous reviews to ensure the explanation fits the user's reading habit.

4. Experiment

4.1. Dataset and preprocessing

We use the Amazon review dataset [15] to evaluate the proposed method. This real-world dataset includes the

ratings, text, and helpfulness votes for every review in the range of May 1996 to Oct 2018. To save the calculation cost, we use the 5-core subsets (the dense subset that guarantees each user and each item has at least five reviews) in the categories Cell Phones and Accessories and Digital Music. We found no significant difference in the results of different categories, so these two categories are chosen randomly. The statistics about the datasets are provided in Table 1.

Table 1 Statistics about datasets

	Cellphone	Cellphone(after preprocessing)	Music	Music(after preprocessing)
#Reviews	1,128,437	45,772	158,827	29,899
#User	157,212	9,484	16,470	3,680
#Item	48,186	1,519	11,790	1,047

We need to keep the explanation convincing because we generate the explanation simultaneously with the recommendation of the item to the user. Thus, negative review texts are not used to generate an explanation. We filter out the negative reviews (rating is lower than four on a five-point scale) and review with blank text. Besides, we also filter out some users and items that only have less than 10 reviews in the dataset to ensure that each user and item have history reviews. This is because we focus on generating personalized explanations based on the history reviews of the user and item, so the cold users with very few reviews are not included in the evaluation. The preprocessed dataset is used to do the 5-fold cross-validation in the experiment. The dataset's division is done by randomly holding reviews and treating them as ground truth, which is not included in the training set. The ratio of the test set and the train set is 1:4.

In addition to this, since one review contains several sentences, and each sentence could belong to different aspects, we also split the reviews into sentences in the train set. Moreover, each sentence is treated as a short review of the user.

4.2. Evaluation metrics

In the experiment, we use the existing text review of the user commented on the item as ground truth and compute the textual similarity of ground truth and generated explanation. The more similarity that explanation has means the explanation fits the user's expectation more. We used two evaluation metrics, Word mover's distance [16] and Semantic textual similarity [17]. Both metrics are wildly used sentence similarity measurement metrics in the natural language processing field.

Word mover's distance uses word embeddings of two sentences to calculate the dissimilarity of them. It calculates the distance that the embedded words in one sentence need to move to the embedded words in the other sentence in the word2vec embedding space and takes the minimum value of the distance as the dissimilarity. The smaller distance means more similar to two sentences.

The semantic textual similarity is based on a universal sentence encoder, encoding sentences into embedding vectors. The semantic similarity is calculated directly by computing the cosine similarity of sentence embeddings. Eq. 4 shows how semantic textual similarity calculates, where u and v represent the sentence embeddings of two sentences.

$$sim(u, v) = \frac{u \cdot v}{\|u\|\|v\|} \qquad \text{(Equation 4)}$$

4.3. Baseline methods

To validate the performance of our proposed method, we use two explanation generation methods as baselines.

(1) Random generation, which randomly takes one sentence from the history reviews of the recommended item m without regarding aspects.

(2) Popular-based generation, which ranks the sentences of history reviews of the recommended item m in the order of the helpfulness votes and takes the top-l sentence as the explanation.

To ensure the fairness of evaluation, all the methods take l as the user's previous reviews' average length.

4.4. Evaluation result

The similarity evaluation result when the number of aspects is set to 10 for both the proposed method and baselines are concluded in Table 2 and Table 3. Moreover, some examples of the explanation we generated are shown in Table 4. We conduct a paired t-test between each baseline and our proposed method for both metrics on the 5-fold dataset and confirm that our proposed method's outperformance is statistically significant ($p<0.01$).

Table 2 Evaluation results for Cellphones and Accessories dataset

	Word mover's distance	Semantic textual similarity
Aspect-based generation (Proposed)	**2.347**	**0.503**
Random generation (Baseline)	2.384	0.498
Popular generation (Baseline)	2.403	0.492

Table 3 Evaluation results for Digital Music dataset

	Word mover's distance	Semantic textual similarity
Aspect-based generation (Proposed)	**0.800**	**0.504**
Random generation (Baseline)	0.917	0.448
Popular generation (Baseline)	0.907	0.466

Table 4 Examples of explanation generated by three methods

	Cellphone	Music
Ground truth	Easy to install and fits fine	This song gets stuck in your head! Reminds me of Empire Of The Son's Walking On A Dream!
Aspect-based generation (Proposed)	Works great and easy to install.	I could listen to this song all day! I sometimes put the song on repeat and who knows how many times I'll listen to it.
Random generation (Baseline)	Great product!	It's a great song as has an interesting vibe to it. So it must be cool.
Popular generation (Baseline)	easy to install but wish it was slightly bigger but it's good enough.	Really love this song! The song is fun, funky and up beat.

From the result, we make the following observations:

(1) Our proposed method outperforms baselines and generates an explanation more similar to the ground truth in Cellphone datasets.

(2) The popular-based method works worse in the Cellphone dataset, but the random-based method works worse in the Music dataset.

(3) The differences between the three generation methods are distinct in the two datasets. In the Cellphone dataset, the proposed method outperforms the popular-based method 2.2% for the semantic textual similarity. Meanwhile, in the Music dataset, our proposed method outperforms the popular-based method 8.2%.

The result indicates the proposed method, which generates recommendation explanations based on reviews with aspect matching, could better show the user's real reaction; thus, the explanation corresponds to the user's expectation.

5. Conclusion and future work

This paper proposed a method to generate a textual explanation that can match a target user's preference aspects. We first extract the aspects from users' reviews and cluster the reviews by aspects. Then we detect the most interested aspect of the target user and generate the explanation using the history reviews of the recommended items that correspond to the target user's most interesting aspect. According to our experiment, our textual explanation generation method outperforms the random generation method 1.0% and popular-based generation method 2.2% on the Cellphone dataset, over the random generation method 12.5% and the popular-based generation 8.2% on the Music dataset.

Though our proposed method achieves a better explanation, there are some places worth further improvements. Firstly, the rank algorithm for ranking the candidate sentences of aspect-based explanations can be improved. Secondly, the similarity metrics cannot measure users' satisfaction or the quality of personalization in the recommendation system. Hence, we plan to improve the generation method of explanation for future work and conduct a user study experiment to measure users' satisfaction and persuasion.

References

[1] Tintarev, N. and Masthoff, J., 2007, October. Effective explanations of recommendations: user-centered design. In *Proceedings of the 2007 ACM conference on Recommender systems* (pp. 153-156).

[2] Zhang, Y. and Chen, X., 2018. Explainable recommendation: A survey and new perspectives. *arXiv preprint arXiv:1804.11192*.

[3] Lu, Y., Dong, R., and Smyth, B., 2018, September. Why I like it: multi-task learning for recommendation and explanation. In *Proceedings of the 12th ACM Conference on Recommender Systems* (pp. 4-12).

[4] McInerney, J., Lacker, B., Hansen, S., Higley, K.,

Bouchard, H., Gruson, A., and Mehrotra, R., 2018, September. Explore, exploit, and explain: personalizing explainable recommendations with bandits. In *Proceedings of the 12th ACM Conference on Recommender Systems* (pp. 31-39).

[5] Herlocker, J.L., Konstan, J.A., and Riedl, J., 2000, December. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM conference on Computer Supported Cooperative Work* (pp. 241-250).

[6] Vig, J., Sen, S., and Riedl, J., 2009, February. Tagsplanations: explaining recommendations using tags. In *Proceedings of the 14th international conference on Intelligent User Interfaces* (pp. 47-56).

[7] Tintarev, N., 2007, October. Explanations of recommendations. In *Proceedings of the 2007 ACM conference on Recommender System*s (pp. 203-206).

[8] Zhang, Y. and Chen, X., 2018. Explainable recommendation: A survey and new perspectives. *arXiv preprint arXiv:1804.11192.*

[9] Zhang, Y., Lai, G., Zhang, M., Zhang, Y., Liu, Y., and Ma, S., 2014, July. 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* (pp. 83-92).

[10] Chang, S., Harper, F.M., and Terveen, L.G., 2016, September. Crowd-based personalized natural language explanations for recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems* (pp. 175-182).

[11] Chen, C., Zhang, M., Liu, Y., and Ma, S., 2018, April. Neural attentional rating regression with review-level explanations. In *Proceedings of the 2018 World Wide Web Conference* (pp. 1583-1592).

[12] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J., 2001, April. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web* (pp. 285-295).

[13] Wu, Y. and Ester, M., 2015, February. Flame: A probabilistic model combining aspect based opinion mining and collaborative filtering. In *Proceedings of the Eighth ACM international conference on Web Search and Data Mining* (pp. 199-208).

[14] He, R., Lee, W.S., Ng, H.T. and Dahlmeier, D., 2017, July. An unsupervised neural attention model for aspect extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 388-397).

[15] Ni, J., Li, J., and McAuley, J., 2019, November. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing* (EMNLP-IJCNLP) (pp. 188-197).

[16] Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K., 2015, June. From word embeddings to document distances. In *International conference on machine learning* (pp. 957-966).

[17] Cer, D., Yang, Y., Kong, S.Y., Hua, N., Limtiaco, N., John, R.S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C. and Sung, Y.H., 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175.*