Towards Controllable Explanation Generation for Recommender Systems via Neural Template

Lei Li
Hong Kong Baptist University
Hong Kong, China
csleili@comp.hkbu.edu.hk

Li Chen
Hong Kong Baptist University
Hong Kong, China
lichen@comp.hkbu.edu.hk

Yongfeng Zhang
Rutgers University
New Brunswick, USA
yongfeng.zhang@rutgers.edu

ABSTRACT

It has been commonly agreed that the explanation associated with recommendation can be effective in increasing the recommender systems (RS)'s transparency and thus users' satisfaction and acceptance. Among the various types of explanation in RS, the commonly used textual explanation can be roughly classified into two categories, i.e., template-based and generation-based. As for the former, the fixed template may lose flexibility, while, though the latter may enrich the explanation, it may produce less useful content due to the lack of controllability. In this work, we combine the advantages of the two types of method by developing a neural generation approach named Neural Template (NETE) whose explanations are not only flexible but also controllable and useful. Our human evaluation results confirm that the explanations from our model are perceived helpful by users. Furthermore, our case study illustrates that the explanation generation process is controllable. To demonstrate the controllability of our model, we present a demo that can be easily viewed on a Web browser.

CCS CONCEPTS

• **Information systems → Recommender systems**; • **Computing methodologies → Neural networks**; **Natural language generation**.

KEYWORDS

Explainable recommendation, natural language generation, neural networks

ACM Reference Format:
Lei Li, Li Chen, and Yongfeng Zhang. 2020. Towards Controllable Explanation Generation for Recommender Systems via Neural Template. In *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3366424.3383540

1 INTRODUCTION

Recommender systems (RS) have been widely deployed on online platforms, ranging from e-commerce, video-sharing to social media, e.g., Amazon[1], Youtube[2] and Instagram[3], since they can help users find their interested items to consume from a large collection of products. They can also benefit service providers, e.g., increasing their revenue. Over the years, there emerges a variety of recommendation algorithms, including collaborative filtering (CF) [8], matrix factorization [7] and deep neural networks [4]. Though most of them have been demonstrated effective, it is difficult for the latter two methods to illustrate why a product is recommended, because they represent users and items as latent factors/vectors that cannot directly reflect users' preferences for explicit product features.

In recent years, we have witnessed the growing interests in the explainability of RS, because it has been shown that providing explanations has many advantages, such as helping users better understand recommendations and convincing them to try or buy [9]. Among the various explanation forms, the commonly used textual explanation can be broadly grouped into two categories, i.e., *template-based* and *generation-based*. As shown in Table 1, the explanation based on CF (such as that appearing in Amazon) follows a pre-defined template (e.g., "*customers who bought this item also bought*"). Although the slot of template-based explanation from Explicit Factor Models (EFM) [11] can be filled in by a feature, it is still lack of flexibility as the "backbone" of the template is fixed permanently. On the other hand, the generation-based explanation from Attribute-to-sequence (Att2Seq) [3] may enhance the flexibility of explanations, but as shown in the example in Table 1 may be less useful for users to assess the value of a recommended product because it only expresses one's personal feeling rather than describes features of the product.

Without pre-specified information, it is hard for neural generation systems to produce appropriate textual content. Therefore, it is necessary to maintain the controllability of neural generation systems by introducing relevant information, e.g., product features and

Table 1: Explanations provided by different methods. Collaborative Filtering (CF) [8] and Explicit Factor Models (EFM) [11] provide template-based explanations, while Attribute-to-sequence (Att2Seq) [3] and our Neural Template (NETE) model generate explanations. In addition, EFM and NETE both show the effect of filling a feature (i.e., "variety") in templates. The reference is for generation-based methods.

CF	Customers who bought this item also bought.
EFM	You may be interested in ***variety***, on which this product performs well.
Att2Seq	I'm not sure if i need to go back.
NETE	They have a ***variety*** of things to choose from.
Reference	They have a huge ***variety*** of things.

[1]https://www.amazon.com/
[2]https://www.youtube.com/
[3]https://www.instagram.com/

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366424.3383540

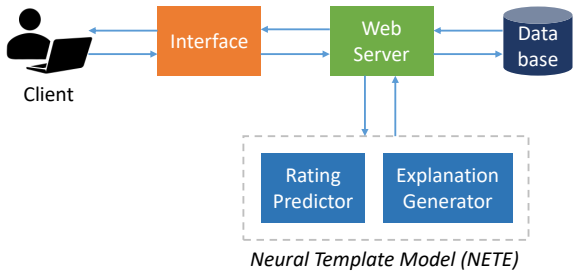


Figure 1: Our system's architecture.

user preference. In this way, the generated explanation is not only relevant to the target product but also to the user, being more useful for justifying recommendations. To this end, we have developed a neural generation system called **Ne**ural **Te**mplate (NETE) that is able to generate explanations according to the specified product feature, combining the merits of both pre-defined template and generation-based model (see Table 1). Our main contributions are summarized as follows:

- To the best of our knowledge, we are the first one who combines the advantages of text generation system and pre-defined template for producing explainable recommendation, as existing works only study one of the two types of approach. Because of such combination, our model can produce controllable explanations.
- We conduct human evaluation to demonstrate that the explanations generated by our model are indeed helpful to users, which demonstrates the value of controllable explanation generation.
- Our case study illustrates that our model is controllable, and our web demo can be easily examined by others in terms of controllability.

2 SYSTEM ARCHITECTURE

In this section, we present an overview of our system in Figure 1, where the major functioning component in our system is the trained model named "Neural Template" (NETE) that consists of two modules, i.e., a rating predictor and an explanation generator, which are responsible for respectively predicting a rating for an item and generating a corresponding natural language explanation. For the former module, we could easily adopt any effective recommendation algorithms to perform rating prediction, such as matrix factorization [7] and neural collaborative filtering [4], given that our focus mainly lies on the task of explanation generation.

For the latter, we adopt gated recurrent units (GRU) [2] to realize explanation generation, as it has been shown more effective than recurrent neural network and more efficient than long short-term memory (LSTM) [5]. The ground-truth explanation that we use is a review sentence containing at least one product feature. An advantage of such sentence is that it not only contains the user's preferences and opinions but also is highly related to the target product. To obtain these useful features, we apply a state-of-the-art sentiment analysis toolkit [12] to user reviews. When the training set is ready, we employ two GRUs for respectively controlling the information flow of the context words and the feature word at each time step, in a similar way to [10], so as to seamlessly integrate the pre-specified feature into the generation process. The initial state of the GRUs encodes the information of the target user and item resulting from multi-layer perceptron, so that the generated content can reflect the attributes of the user and the item. Moreover, the initial state also includes the predicted rating for that user-item pair obtained from the rating predictor for sentiment control of the generation. More specifically, ratings lower than the median (i.e., 3) represent negative sentiment, otherwise positive.

Table 2: Statistics of our datasets

	TripAdvisor	Yelp2019
# of users	9,765	27,147
# of items	6,280	20,266
# of reviews	320,023	1,293,247
# of features	5,069	7,340
Avg. # of reviews / user	32.77	47.64
Avg. # of reviews / item	50.96	63.81
Avg. # of words / explanation	13.01	12.32

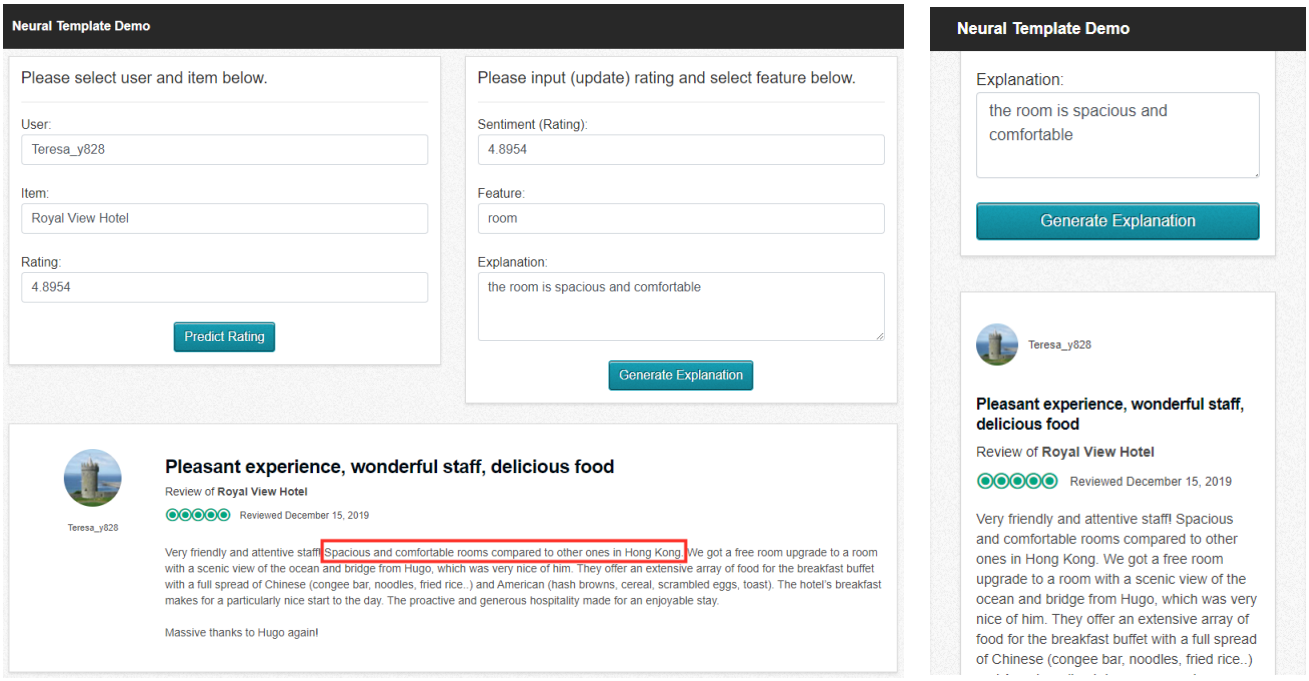
The back-end of our system is built on Python[4], and the front-end is an HTML web page. We use Bootstrap[5] to make the interface compatible with different devices, e.g., mobile phone and pad. Our model NETE is implemented using TensorFlow[6]. After the training process on GPU machine, we save its weight parameters, so that we can restore it on the server. Our demo is deployed on Django[7], a Python-based open-source web framework. At the same time, we use MongoDB[8], a JSON-like NoSQL database, to store user reviews.

As shown in Figure 1, after a client sends a request via the interface to our server, our system returns the following information related to the current recommendation: the predicted rating, the generated explanation from the trained model and the target user review from the database.

3 DEMONSTRATION

Figure 2 shows the demo of our model. The sub-figure (a) is the web interface that clients can visit via a browser, e.g., Chrome. At the same time, we provide a mobile version in the sub-figure (b) that can adapt to mobile devices, e.g., iPhone. When using our demo website, a client can select "User" and "Item" information from the drop-down menus. After s/he clicks the "Predict Rating" button, our system displays the predicted rating score for the selected user-item pair. Because the rating can be used to adjust the sentiment of the generated explanation, it is also displayed in the right column for the client to edit. After that, s/he can further select a feature of the item that is of her/his interest. When the button "Generate Explanation" is pressed, the explanation will be displayed (e.g., "*the room is spacious and comfortable*"). Moreover, if the user's review for that item already exists, it will be shown at the bottom of the interface, accompanied by review rating, review date and review title.

[4]https://www.python.org/
[5]https://getbootstrap.com/
[6]https://www.tensorflow.org/
[7]https://www.djangoproject.com/
[8]https://www.mongodb.com/



(a) Web version on personal computers.

(b) Mobile version that adapts to mobile devices.

Figure 2: Interface of our neural template based explanation demo.

4 EVALUATION

4.1 Datasets

We prepare two datasets, TripAdvisor[9] for hotel and Yelp2019[10] for restaurant, whose statistics are shown in Table 2. We construct the former by collecting reviews from its website, while the latter is publicly available in Yelp Challenge 2019. Since Yelp2019 dataset is about food that people need to consume every day, we conduct human evaluation on it to verify the helpfulness of the generated explanations from our model. Note that because TripAdvisor contains more detailed information, we use it for demonstration.

Each dataset is randomly split into training (80%), validation (10%) and testing (10%) sets, and there is at least one sample in the training set for each user/item. We stop training our model when the loss on the validation set reaches the minimum.

4.2 Human Evaluation on Explanation

To evaluate the quality of explanations and to investigate whether they are truly helpful to users in the context of recommendation, we conduct a small-scale user study on Yelp2019 dataset.

Specifically, we prepare two questions, each of which contains 20 cases that are randomly sampled from the testing set. We invite 10 volunteers (most are postgraduate students in our department) to evaluate the results. The first question (Q1) is a pair-wise evaluation task, where for each case we ask the participant to choose one from two explanations respectively generated by our model NETE and the baseline Attribute-to-sequence (Att2Seq) [3] in terms of the explanation's similarity to the given reference. Notice that, to fairly compare the two models' performance, we hide the model's details and inform the participant that the answers are randomly shuffled.

Att2Seq is a state-of-the-art text generation method with an encoder-decoder framework, where the encoder encodes three attributes, including user, item and ground-truth rating, and the decoder is a two-layer LSTM for review generation. In our implementation, we replace the LSTM with GRU for consistency with our model, and the ground-truth text is the same as in our model. In addition, we remove its attention mechanism, because it impairs the quality of generated content in our experiments. We omit the comparison with [1, 6], because their settings are quite different from our model's. For example, in terms of data sources, [1] makes use of a group of features, and [6] requires review tips (summaries).

This task is to investigate whether our model can generate high-quality explanations relative to the baseline. After that, a point-wise evaluation (Q2) on our model NETE's explanations is performed, for which we ask the participant to judge whether the explanation is helpful for them to evaluate the feature of a recommended restaurant or not. The two questions are listed below:

- **Q1**: Which answer is closer to the reference sentence in terms of the semantic similarity?
- **Q2**: Assume you are interested in one feature of a restaurant, do you think the generated explanation is helpful for you to evaluate that feature?

We depict the results in Figure 3. The bar chart of Q1 shows that our model obtains on average 8 votes across those 20 cases,

[9]https://www.tripadvisor.com/
[10]https://www.yelp.com/dataset/challenge

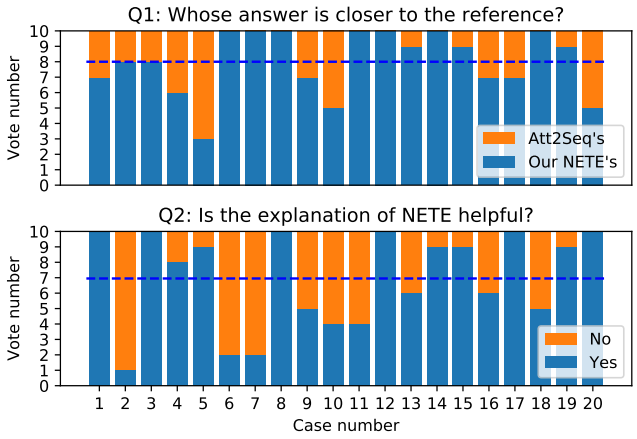


Figure 3: Results of human evaluation on generated explanations on Yelp2019 dataset. The blue dotted line shows the average votes on blue option across 20 cases.

which is obviously higher than 2 votes obtained by Att2Seq, which may be because our model is able to fuse the specified feature into generation, making the results more meaningful. For Q2, the votes are different from case to case, but on average 7 participants agree that our model explains the given features clearly. This verifies that to a large extent our model is competent to generate useful explanations that can help users better understand the recommended product.

4.3 Case Study on Controllability

We present a case study in Table 3, where there are two cases from two different users. Comparing the first case with the second, we can see that varying users while keeping the other inputs (i.e., feature and rating) basically identical, can largely change the content of the generated explanations, which thus infers that our model is able to capture the differences among users. When we manually change the input rating (i.e., the sentiment) for the first case, we find that the sentiment of the generated contents is transformed from positive to negative. For example, the sentence "*ask for higher floors*" becomes "*it was not a high floor*", when the rating is adjusted from 4.09 (positive) to 2.00 (negative). Moreover, in the first case, when we vary features (i.e., "*floors*" and "*rooms*") for the same user, it can be seen that the explanations discuss different topics. This indicates that fusing feature information into the generation can be useful for producing more relevant and suitable explanation. Overall, it proves that our model is controllable, since it is able to capture the variance of different types of input.

However, admittedly, our model is not perfect. The sentiment of the explanation cannot always be controlled. For instance, in the second case, when we manually set the sentiment to negative, the generated content can still be positive, possibly because this user never give negative comments. Lacking such valuable information makes our model less sensitive to this user's sentiment. We leave the improvement on this aspect in future work.

Table 3: Explanations generated by our model NETE on TripAdvisor dataset. The first line of each group shows the ground-truth. The other lines show the specified features and ratings, and the corresponding explanations where we highlight the specified feature in green. The underlined ratings are predicted, while the others are manually set. Ratings below 3 denote negative sentiment, otherwise positive.

Rating	Feature	Explanation
4		***The view from some rooms and higher floors is hard to beat.***
4.09 (+1)	floors	Ask for higher floors .
2.00 (-1)	floors	It was not a high floor .
4.09 (+1)	rooms	The rooms are very comfortable.
2.00 (-1)	rooms	The rooms are not very comfortable.
3		***Rooms on the higher floors have a nice view.***
3.73 (+1)	floors	Rooms on the higher floors are better.
2.00 (-1)	floors	I was given a room on the higher floors and the rooms are very spacious.
3.73 (+1)	rooms	The rooms are spacious and the rooms are very comfortable.
2.00 (-1)	rooms	The rooms are very small and the rooms are very spacious.

5 CONCLUSION

In this work, we present a controllable explanation generation demo. This system aims to unify the merits of both neural generation systems and pre-defined templates in order to achieve both flexibility and controllability simultaneously. Our human evaluation demonstrates that the proposed Neural Template (NETE) model can in practice produce useful explanations, and the demo and case study additionally showcase the controllability of our system. This is our first step towards producing controllable explanations. In the future, we will integrate more attributes into our system to make its generated explanations more expressive. We also plan to conduct more experiments on other datasets and against strong baselines, to further prove our model's robustness and competitiveness. In particular, we intend to verify the controllability of our system quantitatively through human evaluation.

ACKNOWLEDGMENTS

The authors would like to thank Dr. William Cheung, Dr. Baoyao Yang, Dr. Meng Pang, Qingxiong Tan, Qi Tan, Jinfu Ren, Guozhong Li, Chunpeng Zhou, Dong Qian and Rui Shao for voluntarily participating in their human evaluation.

REFERENCES

[1] Hanxiong Chen, Xu Chen, Shaoyun Shi, and Yongfeng Zhang. 2019. Generate Natural Language Explanations for Recommendation. In *Proceedings of SIGIR'19 Workshop on ExplainAble Recommendation and Search*. ACM.

[2] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1724–1734.

[3] Li Dong, Shaohan Huang, Furu Wei, Mirella Lapata, Ming Zhou, and Ke Xu. 2017. Learning to Generate Product Reviews from Attributes. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, Vol. 1. 623–632.

[4] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 173–182.

[5] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural computation* 9, 8 (1997), 1735–1780.

[6] Piji Li, Zihao Wang, Zhaochun Ren, Lidong Bing, and Wai Lam. 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*. ACM, 345–354.

[7] Andriy Mnih and Ruslan R Salakhutdinov. 2008. Probabilistic matrix factorization. In *Advances in neural information processing systems*. 1257–1264.

[8] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*. ACM, 285–295.

[9] Nava Tintarev and Judith Masthoff. 2015. Explaining Recommendations: Design and Evaluation. In *Recommender Systems Handbook* (2 ed.), Bracha Shapira (Ed.). Springer, Chapter 10, 353–382.

[10] Lili Yao, Yaoyuan Zhang, Yansong Feng, Dongyan Zhao, and Rui Yan. 2017. Towards implicit content-introducing for generative short-text conversation systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2190–2199.

[11] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. 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*. ACM, 83–92.

[12] Yongfeng Zhang, Haochen Zhang, Min Zhang, Yiqun Liu, and Shaoping Ma. 2014. Do users rate or review? Boost phrase-level sentiment labeling with review-level sentiment classification. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. ACM, 1027–1030.