

NRPA: Neural Recommendation with Personalized Attention

Hongtao Liu
College of Intelligence and
Computing, Tianjin University
htliu@tju.edu.cn

Fangzhao Wu
Microsoft Research Asia
wufangzhao@gmail.com

Wenjun Wang
College of Intelligence and
Computing, Tianjin University
wjwang@tju.edu.cn

Xianchen Wang
College of Intelligence and
Computing, Tianjin University
wangxc@tju.edu.cn

Pengfei Jiao*
Center for Biosafety Research and
Strategy, Tianjin University
pjiao@tju.edu.cn

Chuhan Wu
Electronic Engineering
Tsinghua University
wuch15@mails.tsinghua.edu.cn

Xing Xie
Microsoft Research Asia
xing.xie@microsoft.com

ABSTRACT

Existing review-based recommendation methods usually use the same model to learn the representations of all users/items from reviews posted by users towards items. However, different users have different preference and different items have different characteristics. Thus, the same word or the similar reviews may have different informativeness for different users and items. In this paper we propose a neural recommendation approach with personalized attention to learn personalized representations of users and items from reviews. We use a review encoder to learn representations of reviews from words, and a user/item encoder to learn representations of users or items from reviews. We propose a personalized attention model, and apply it to both review and user/item encoders to select different important words and reviews for different users/items. Experiments on five datasets validate our approach can effectively improve the performance of neural recommendation.

CCS CONCEPTS

• Information systems → Recommender systems.

KEYWORDS

Neural recommendation; Personalized attention; Review mining

ACM Reference Format:

Hongtao Liu, Fangzhao Wu, Wenjun Wang, Xianchen Wang, Pengfei Jiao, Chuhan Wu, and Xing Xie. 2019. NRPA: Neural Recommendation with Personalized Attention. In *42nd Int'l ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR'19), July 21–25, 2019, Paris, France*. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3331184.3331371>

*Corresponding Author: Pengfei Jiao, pjiao@tju.edu.cn

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SIGIR '19, July 21–25, 2019, Paris, France

© 2019 Association for Computing Machinery.

ACM ISBN 978-1-4503-6172-9/19/07...\$15.00

<https://doi.org/10.1145/3331184.3331371>

1 INTRODUCTION

Recommender Systems (RS) are an information filtering systems that can learn user's interests and hobbies based on their historical behavior records, and predict users preference or ratings for items, which are ubiquitous today at e-commerce platforms such as Amazon and Netflix.

A number of works have been proposed for recommendation systems. Collaborative Filtering (CF) [3] techniques are one of the most popular recommender methods, which are extensively used in industry. Many of CF techniques are based on matrix factorization (MF) that decomposes the user-item rating matrix into two matrices corresponding to latent features of users and items [6]. However, these methods represent users and items only based on numeric ratings while the ratings suffer from the natural sparsity. Using text reviews to model user preference and item features is one approach to alleviate the above issues [1, 2, 4, 8, 9]. For example, ConvMF [2] integrates convolutional neural network into probabilistic matrix factorization to exploit both ratings and item description documents. TARMF [4] utilizes attention-based recurrent neural networks to extract topical information from reviews and integrates textual features into probabilistic matrix factorization to enhance the performance of recommendation.

Despite their significant improvement of performance in recommendation, most existing methods learn the representations from reviews for all users or items using the same model and ignore the deep personalized feature of users and items. As a concrete example, suppose that User A cares more about the price of items than the quality and User B cares more quality than price, both of them write a similar review such as "this camera with a high price is easy to use." and then User A would give the camera a unsatisfied rating since the price is high while User B would vote a satisfied rating. Thus, the same reviews are of different informativeness in terms of different users or items and it is necessary to be more personalized when learning representations from reviews for users or items. As a result, we should exploit the individuality of users and items for neural recommendation.

To this end, we propose a Neural Recommendation with hierarchical Personalized Attention (NRPA) model to learn personalized representation for users and items. Specifically, our NRPA contains

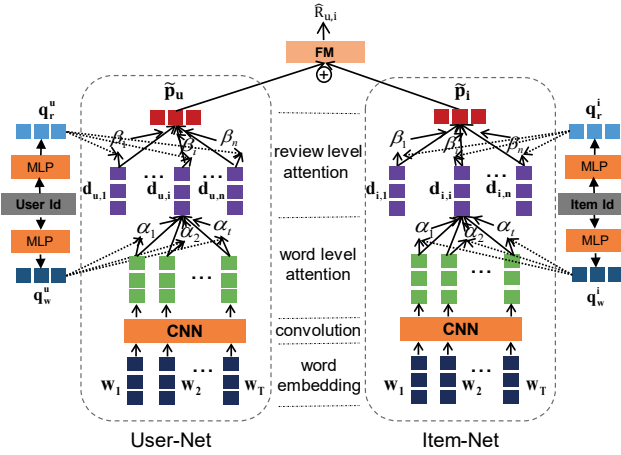


Figure 1: The framework of our NRPA approach.

two components, i.e., a review encoder to learn representations of reviews, a user/item encoder to learn representations of user/item from their reviews. In review encoder, we utilize Convolutional Neural Network (CNN) to extract semantic features of reviews from words, and then use personalized word-level attention to select more important words in a review for each user/item. In user/item encoder we apply personalized review-level attention to learn the user/item representation via aggregating all the reviews representations according to their weights. Moreover, the word- and review-level attention vectors of a user/item are generated by two multi-layer neural networks with the user/item ID embedding as input. The two attention vectors can be seen as an indicator for each user and item under hierarchical views (i.e., word and review level). At last, we combine the representations of a user and a target item and feed them into a Factorization Machine [6] layer to predict the rating that the user would vote the item. We conduct extensive experiments on five benchmark datasets for recommendation with reviews. The results validate the effectiveness of our personalized attention.

2 PROPOSED METHOD

In this section, we introduce our NRPA approach in detail. Our approach contains three major components, i.e., a User-Net to learn user representations, an Item-Net to learn item representations, and a rating prediction module to predict the rating scores based on user and item representations. Both User-Net and Item-Net contain two modules, i.e., a review encoder to learn representations of reviews from words and a user/item encoder to learn representations of users and items from reviews. The overview of our NRPA approach is shown in Figure 1.

2.1 Review Encoder

We utilize word embedding to map each word into low-dimensional vectors and use Convolutional Neural Network (CNN) to extract the semantic features of text reviews. Then, we introduce a personalized attention into our model to highlight important words. Some notations used in the following section are defined as follows: U

and I indicates the user set and item set, respectively. $R \in \mathcal{R}^{|U| \times |I|}$ denotes the rating matrix and $D \in \mathcal{R}^{|U| \times |I|}$ means the text review collection. $d_{u,i} = \{w_1, \dots, w_T\}$ denotes the review written by user u for item i .

Word Embedding and Convolution. Given a review $d_{u,i}$, we first embed each word w_k in $d_{u,i}$ to a d_w dimensional vector w_k via word embedding. Then, we transform the review $d_{u,i}$ into a matrix $M_{u,i} = [w_1, w_2, \dots, w_T]$. Afterwards, we perform convolution operator to calculate the feature matrix of the review $C \in \mathcal{R}^{K \times T}$:

$$C_j = \sigma(W_j * M_{u,i} + b_j), 1 \leq j \leq K, \quad (1)$$

where $*$ is the convolution operator, K is the number of filters and W_j is the weight matrix of the j -th filter. Each column in C (denoted as $z_k \in \mathcal{R}^K$) represents the semantic feature of the k -th word in the review.

Personalized Attention Vector over Word Level. We first explore how to generate the attention vector q_u^w for the user u which can embody the personalization. Since each user or item has the unique ID feature, we first represent all users and items into low-dimensional vectors via an embedding layer based on their IDs. As shown in Figure 1, given the ID embedding of the user u , we utilize Multilayer Perceptron (MLP) to generate the personalized attention vector for user u , denoted as:

$$q_u^w = \text{ReLU}(W_1 u_{id} + b_1), \quad (2)$$

where W_1 is the weight matrix of MLP, b_1 is the bias term and u_{id} is the ID embedding of user u .

User-Specific Attention over Word Level. As mentioned above, not all words of a review are equally important for the representation of the review meaning. To highlight the important words, we employ the attention pooling mechanism in word level, denoted as:

$$g_k = q_u^w A z_k, \quad (3)$$

$$\alpha_k = \frac{\exp(g_k)}{\sum_{j=1}^T \exp(g_j)}, \alpha_k \in (0, 1), \quad (4)$$

where A is the harmony matrix in attention, q_u^w is the attention vector specifically for the user u obtained above, and z_k is the representations of the k -th word above. α_i is attention weight of the i -th word in the review. Similarly, the item i has a unique attention vector denoted as q_i^w in Item-Net. Afterwards, we obtain the representation of the i -th review of user u via aggregating feature vectors of all words:

$$d_{u,i} = \sum_{j=1}^T \alpha_j z_j. \quad (5)$$

2.2 User and Item Encoder

After we have obtained all the review representations of users and items above, we will explore how to aggregate them together to represent users or items. As stated above, different reviews are of different importance for representation of user. Besides, the information of a review varies from different users. Hence, we introduce a user-specific attention mechanism to focus on the useful reviews for each user.

Personalized Attention Vectors over Review Level. Based on the user id embedding \mathbf{u}_{id} , we utilize another MLP layer to generate the personalized review-level attention vector for user u :

$$\mathbf{q}_r^u = \text{ReLU}(\mathbf{W}_2 \mathbf{u}_{id} + \mathbf{b}_2), \quad (6)$$

where ReLU is the activation function, \mathbf{W}_2 is the weight matrix, \mathbf{b}_2 is the bias term.

User-Specific Attention over Review Level. Given the review set $d_u = \{d_{u,1}, d_{u,2}, \dots, d_{u,N}\}$, we apply attention to highlight those informative reviews and de-emphasize those meaningless. To be specific, we compute the weight β_j of the j -th review of the i -th user as follows:

$$e_j = \mathbf{q}_r^u \mathbf{A}_2 \mathbf{d}_{u,j}, \quad (7)$$

$$\beta_j = \frac{\exp(e_j)}{\sum_{k=1}^N \exp(e_k)}, \beta_j \in (0, 1), \quad (8)$$

where \mathbf{A}_2 is the matrix in attention; \mathbf{q}_r^u is the query vector for the user u and each users have a unique attention vector to find informative reviews. Afterwards, we obtain the text feature $\tilde{\mathbf{p}}_u$ of user u via aggregating all the reviews according to their weights:

$$\tilde{\mathbf{p}}_u = \sum_{j=1}^N \beta_j \mathbf{d}_{u,j}. \quad (9)$$

Similarly, we can get the feature of item i , denoted as $\tilde{\mathbf{p}}_i$.

2.3 Rating Prediction

In this section we predict the ratings based on $\tilde{\mathbf{p}}_u$ and $\tilde{\mathbf{p}}_i$. First, we concatenate $\tilde{\mathbf{p}}_u$ and $\tilde{\mathbf{p}}_i$ and feed into a Factorization Machine (FM) [6] to predict rating:

$$\hat{\mathbf{o}} = \tilde{\mathbf{p}}_u \oplus \tilde{\mathbf{p}}_i, \quad (10)$$

$$\hat{R}_{u,i} = \hat{w}_0 + \sum_{i=1}^{|\hat{\mathbf{o}}|} \hat{w}_i \hat{o}_i + \sum_{i=1}^{|\hat{\mathbf{o}}|} \sum_{j=i+1}^{|\hat{\mathbf{o}}|} \langle \hat{v}_i, \hat{v}_j \rangle \hat{o}_i \hat{o}_j, \quad (11)$$

where \oplus is the concatenation operation; \hat{w}_0 and \hat{w}_i are both parameters in FM.

3 EXPERIMENTS

3.1 Datasets and Experimental Settings

Our experiments are conducted on five benchmark datasets. The first two datasets Yelp 2013 (denoted as Yelp13) and Yelp 2014 (denoted as Yelp14) are selected from Yelp Dataset Challenge¹. The other three datasets Electronics, Video Games and Gourmet Foods are selected from Amazon dataset², and we denote them as Elec., Games and Foods respectively. Note that all datasets contain reviews with ratings (from 1 to 5). The details of the datasets are shown in Table 1. We randomly split each dataset into training set, validation set and test set with 80%, 10% and 10% respectively.

In our experiments, we use validation dataset to tune the hyperparameters in our model. The word embedding vectors are 300-dimensional. The dimension of ID embedding is set to 32. The number of filters in CNN and the dimension of attention vectors is 80. The window size of CNN is set to 3. Following previous

¹<https://www.yelp.com/dataset/challenge>

²<http://jmcauley.ucsd.edu/data/amazon/>

Table 1: Statistics of the five datasets in our experiments.

Dataset	#users	#items	#ratings	density
Yelp13	1,631	1,633	78,966	2.965
Yelp14	4,818	4,194	231,163	1.144
Elec.	192,403	63,001	1,689,188	0.014
Games	24,303	10,672	231,780	0.089
Foods	14,681	8,713	151,254	0.118

Table 2: Comparisons between NRPA and baselines.

	Yelp13	Yelp14	Elec.	Games	Foods
PMF	0.985	1.053	1.411	1.297	1.251
CTR	0.975	1.013	1.284	1.147	1.139
ConvMF+	0.917	0.954	1.241	1.092	1.084
DeepCoNN	0.880	0.910	1.232	1.130	0.985
NARRE	0.879	0.906	1.215	1.112	0.986
TARMF	0.875	0.909	1.147	1.043	1.019
NRPA	0.872	0.897	1.047	1.014	0.953

works [1, 10], we utilize Mean Squared Error (MSE) as the evaluation metric.

3.2 Performance Evaluation

We evaluate our method NRPA with the following baseline methods:

- **PMF** [5] models the latent factors for users and items by introducing Gaussian distribution.
- **CTR** [7] learns interpretable latent structure from user-generated content to integrate probabilistic modeling into collaborative filtering.
- **ConvMF+** [2] incorporates convolutional neural network into Matrix Factorization to learn item features from item review documents.
- **DeepCoNN** [10] models users and items via combining all their associated reviews by convolutional neural network.
- **NARRE** [1] is a newly proposed method that introduces neural attention mechanism to build the recommendation model and select highly-useful reviews simultaneously.
- **TARMF** [4] is a recommendation model which utilizes attention-based recurrent neural networks to extract topical information from review documents.

The MSE results of all methods are shown in Table 2. Our model NRPA outperforms all the baseline methods among all the five datasets which indicates the robust effectiveness of our personalized attention in modeling users and items. Besides, we can observe that (1) The methods with reviews perform better than those methods with only ratings (i.e., PMF and CTR). The reason may be that the reviews with the rich semantic textual information are powerful in capturing the feature of users and items. (2) Though both NARRE and TARMF utilize the attention mechanism to focus on more important information, our method NRPA achieves a better performance than them. We conclude that our model NRPA with hierarchical attention can exploit the deep personalized features (i.e., word level and review level) of users and items, which can represent users and items more precise. This result is consistent with our

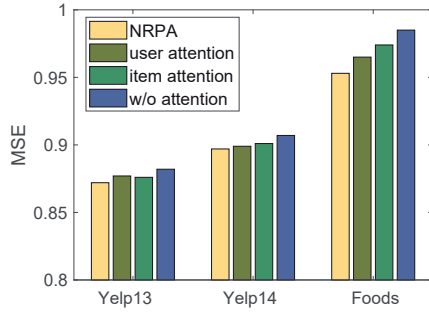


Figure 2: Effectiveness of user and item attentions.

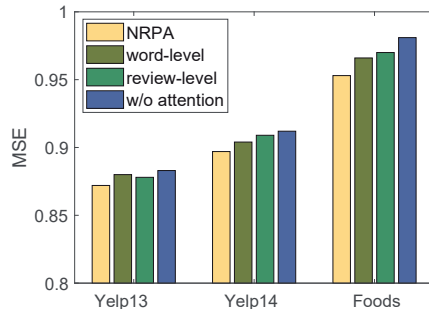


Figure 3: Effectiveness of word- and review-level attentions.

intuitive motivation that users and items should be characterized by individuation in recommendation.

3.3 Effectiveness of Personalized Attention

In this section we further explore the effectiveness of our personalized attention module. First we evaluate the effect of user attention and item attention respectively. From the results in Figure 2 (experiments on three datasets for space limitation), we can find that comparing with the variant without attention (i.e., average weight for all reviews), both variants with user attention and item attention can improve the performance of rating prediction in recommendation. This is because different users or items always have their unique preference or features. And our personalized attention can effectively capture the personality of users and items, which is beneficial for learning a precise representation of users and items.

Besides, we explore the effectiveness of word-level attention and review-level attention. As shown in Figure 3, we can observe that the variants with only word-level attention and only review-level attention can both perform better than the model without any attention. This is because word level attention can recognize those important words for each user or item; review level attention can help to focus on the more informative reviews during modeling user preference and item features.

3.4 Parameter Analysis

Since our personalized attention vectors are generated by the user and item id embedding, this section explores the effect of varying dimension of the id embedding. From the result in Figure 4, we can

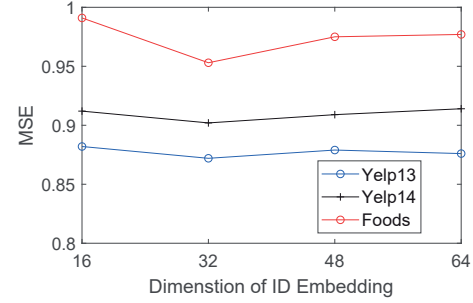


Figure 4: The influence of ID embedding dimension.

see that as the dimension increases, the MSE first decrease, then reaches the best, and decreases afterwards. When dimension is too smaller, the attention vectors may not learn the diversity of users and items enough. However if the dimension becomes too large, the model may suffer from overfitting. The optimal value of the ID dimension is 32 regardless of different datasets.

4 CONCLUSION

In this paper, we propose a neural recommendation approach with personalized attentions to learn personalized user and item representations from reviews. The core of our approach is a personalized attention model whose query vectors are learned from the embeddings of user and item IDs. We apply this attention model to both review encoder and user/item encoder to select different important words and reviews for different users and items. In this way the different preference of users and different characteristics of items can be better captured. The experiments on five benchmark datasets show that our approach can effectively improve the performance of neural recommendation.

ACKNOWLEDGMENTS

This work was supported by the National Key R&D Program of China (2018YFC0809800, 2018YFC0831000), the National Natural Science Foundation of China (91746205, 91746107, 51438009).

REFERENCES

- [1] Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. 2018. Neural attentional rating regression with review-level explanations. In *WWW*. 1583–1592.
- [2] Donghyun Kim, Chanyoung Park, Jinoh Oh, Sungyoung Lee, and Hwanjo Yu. 2016. Convolutional matrix factorization for document context-aware recommendation. In *RecSys*. 233–240.
- [3] Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon.com recommendations: Item-to-item collaborative filtering. *Internet Computing* (2003), 76–80.
- [4] Yichao Lu, Ruihai Dong, and Barry Smyth. 2018. Coevolutionary Recommendation Model: Mutual Learning between Ratings and Reviews. In *WWW*. 773–782.
- [5] Andriy Mnih and Ruslan R Salakhutdinov. 2008. Probabilistic matrix factorization. In *NIPS*. 1257–1264.
- [6] Steffen Rendle. 2010. Factorization machines. In *ICDM*. 995–1000.
- [7] Chong Wang and David M Blei. 2011. Collaborative topic modeling for recommending scientific articles. In *KDD*. 448–456.
- [8] Xianchen Wang, Hongtao Liu, Peiyi Wang, Fangzhao Wu, Hongyan Xu, Wenjun Wang, and Xing Xie. 2019. Neural Review Rating Prediction with Hierarchical Attentions and Latent Factors. In *DASFAA*. 363–367.
- [9] Chuhan Wu, Fangzhao Wu, Junxin Liu, and Yongfeng Huang. 2019. Hierarchical User and Item Representation with Three-Tier Attention for Recommendation. In *NAACL*.
- [10] Lei Zheng, Vahid Noroozi, and Philip S Yu. 2017. Joint deep modeling of users and items using reviews for recommendation. In *WSDM*. 425–434.