Hierarchical User and Item Representation with Three-Tier Attention for Recommendation

Chuhan Wu[1], **Fangzhao Wu**[2], **Junxin Liu**[1], **and Yongfeng Huang**[1]
[1]Department of Electronic Engineering, Tsinghua University Beijing 100084, China
[2]Microsoft Research Asia
{wuch15,ljx16,yfhuang}@mails.tsinghua.edu.cn
wufangzhao@gmail.com

Abstract

Utilizing reviews to learn user and item representations is useful for recommender systems. Existing methods usually merge all reviews from the same user or for the same item into a long document. However, different reviews, sentences and even words usually have different informativeness for modeling users and items. In this paper, we propose a hierarchical user and item representation model with three-tier attention to learn user and item representations from reviews for recommendation. Our model contains three major components, i.e., a sentence encoder to learn sentence representations from words, a review encoder to learn review representations from sentences, and a user/item encoder to learn user/item representations from reviews. In addition, we incorporate a three-tier attention network in our model to select important words, sentences and reviews. Besides, we combine the user and item representations learned from the reviews with user and item embeddings based on IDs as the final representations to capture the latent factors of individual users and items. Extensive experiments on four benchmark datasets validate the effectiveness of our approach.

1 Introduction

Learning accurate user and item representations is very important for recommender systems (Tay et al., 2018). Many of existing recommendation methods learn user and item representations based on the ratings that users gave to items (Koren et al., 2009; Mnih and Salakhutdinov, 2008). For example, Koren et al. (2009) proposed a matrix factorization method based on SVD to learn latent representations of users and items from the rating matrix between users and items. However, since the numbers of users and items in online platforms are usually huge, and the rating matrix between users and items is usually very sparse, it is quite difficult for those rating based recommendation methods to learn accurate user and item representations (Zheng et al., 2017; Tay et al., 2018).

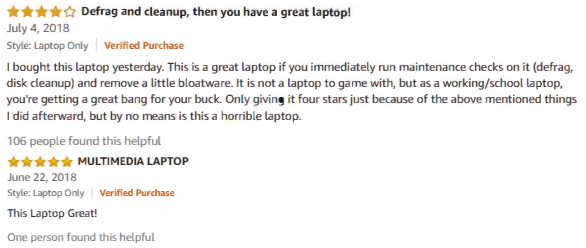


Figure 1: Two example reviews.

Luckily, in many online platforms such as Amazon and IMDB, there are rich reviews written by the users to express their opinions on items. These reviews can provide rich information of items. For example, if sentences like "bad battery life" and "battery capacity is low" frequently appear in the reviews of a smartphone, then we can infer the performance of this item in battery life is not good. The reviews also contain rich information of users. For example, if a user frequently mentions "the price is too high" and "very expensive" in his/her reviews for different items, then we can infer this user may be sensitive to price. Thus, these reviews can help enhance the learning of user and item representations especially when ratings are sparse, which is beneficial for improving the performance of recommender systems (Zheng et al., 2017).

Utilizing reviews to learn user and item representations for recommendation has attracted increasing attentions (Zheng et al., 2017; Catherine and Cohen, 2017). For example, Zheng et al. (2017) proposed a DeepCoNN method to learn the representations of users and items from reviews using convolutional neural networks (CNN), and achieved huge improvement in recommendation performance. These methods usually concatenate the reviews from the same user or the

Proceedings of NAACL-HLT 2019, pages 1818–1826
Minneapolis, Minnesota, June 2 - June 7, 2019.

same item into a long document. However, different reviews usually have different informativeness in representing users and items. For example, in Fig. 1 the first review is much more informative than the second one. Distinguishing informative reviews from noisy ones can help learn more accurate user and item representations. In addition, different sentences in the same review may also have different informativeness. For example, in Fig. 1 the sentence "it is not a laptop to game with" contains more important information than "I bought this laptop yesterday". Besides, different words in the same sentence may also have different importance. For example, in "this is a great laptop if you ..." the word "great" is more important than "you" in modeling this item.

In this paper, we propose a hierarchical user and item representation model with three-tier attention (*HUITA*) to learn informative user and item representations from reviews for recommendation. In our approach, the hierarchical user and item representation model contains three major components, i.e., a sentence encoder to learn sentence representations from words, a review encoder to learn review representations from sentences, and a user/item encoder to learn user/item representations from the all reviews posted by this user or for this item. In addition, we propose to incorporate a three-tier attention network into our model to select important words, sentences and reviews to learn more informative user and item representations. Besides, we combine the user and item representations learned from the reviews with the user and item embeddings based on their IDs as the final representations to capture the latent factors of each individual users and items. We conduct extensive experiments on four benchmark datasets. The results show our approach can effectively improve the performance of recommendation and outperform many baseline methods.

2 Related Work

Learning user and item representations from reviews for recommendation has attracted many attentions (McAuley and Leskovec, 2013; Ling et al., 2014; Bao et al., 2014; Zhang et al., 2014; Diao et al., 2014; He et al., 2015; Tan et al., 2016; Ren et al., 2017). Many of the existing methods focus on extracting topics from reviews to model users and items. For example, McAuley and Leskovec (2013) proposed a Hidden Factors as Topics (HFT) method to use the topic modeling technique LDA to discover the latent aspects of users and items from the reviews. Ling et al. (2014) proposed a Ratings Meet Reviews (RMR) method to enhance the representations of users and items by extracting topics from review texts and aligning the dimensions of these topics with the latent user representations obtained from the rating matrix using matrix factorization. Bao et al. (2014) proposed a TopicMF approach to jointly model user and item representations using rating scores via matrix factorization and using review texts via non-negative matrix factorization (NMF) to obtain topics. However, these methods only extract the topic information from reviews, and a large amount of important semantic information is not captured. In addition, these methods are usually based on topic models and cannot effectively model the contexts and orders of words in reviews, both of which are important for inferring user preferences and item properties.

In recent years, several deep learning based methods have been proposed to learn user and item representations from reviews for recommendation (Zhang et al., 2016; Zheng et al., 2017; Catherine and Cohen, 2017; Seo et al., 2017b,a; Chen et al., 2018; Tay et al., 2018). For example, Zheng et al. (2017) proposed a DeepCoNN method which uses CNN to learn representations of users and items from their reviews. Catherine and Cohen (2017) proposed a TransNets method to learn user and item representations from reviews using CNN and regularize these representations to be close to the representations of the review written by the target user to the target item. Seo et al. (2017b) proposed to learn user and item representations via CNN network as well as attention network over word embeddings. These methods concatenate all the reviews from the same user or for the same item into a long document, and cannot distinguish informative reviews from noisy ones. Chen et al. (2018) proposed to model the usefulness of reviews using review-level attention to enhance the learning of user and item representations. However, their method regards each review as a long sentence, and cannot distinguish informative sentences and words from less informative ones. Different from the aforementioned methods, in our approach we propose a hierarchical framework to learn user and item representations from reviews for recommendation. Our

model first learns sentence representations from words, then learns review representations from their sentences, and finally learns user/item representations from their reviews. Our model also contains a three-tier attention network to jointly select important words, sentences and reviews to learn more informative user and item representations. Experiments on benchmark datasets validate the advantage of our approach over existing methods in recommendation.

3 Our Approach

In this section, we introduce our *HUITA* approach to learn user and item representations from reviews for recommendation. The architecture of our approach is shown in Fig. 2. There are three major modules in our approach. The first one is *sentence encoder* which learns representations of sentences from words. The second one is *review encoder* which learns representations of reviews from sentences. And the third one is *user/item encoder*, which learns the representations of users and items from their reviews. Next we introduce each module in detail.

3.1 Sentence Encoder

The *sentence encoder* module is used to learn representations of sentences from words. According to Fig. 2, there are three layers in this module.

The first layer is word embedding. It is used to convert a sequence of words into a sequence of low-dimensional dense vectors which contain semantic information of these words. Denote a sentence s contains M words $[w_1, w_2, ..., w_M]$. Through the word embedding layer the sentence s is transformed into a vector sequence $[\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_M]$ using a word embedding matrix $\mathbf{E} \in \mathcal{R}^{V \times D}$, where V and D represent the vocabulary size and the word embedding dimension, respectively. The word embedding matrix $\mathbf{E}$ is initialized using pretrained word embeddings, and fine-tuned during model training.

The second layer is a convolutional neural network (CNN). CNN is an effective neural architecture for capturing local information (LeCun et al., 2015). We employ a word-level CNN to capture the local contexts of words to learn their contextual representations. Denote $\mathbf{c}_i^w$ as the contextual representation of the word w_i, which is computed as follows:

$$\mathbf{c}_i^w = \mathrm{ReLU}(\mathbf{U}_w \times \mathbf{e}_{(i-K_w):(i+K_w)} + \mathbf{b}_w), \quad (1)$$

where $\mathbf{e}_{(i-K_w):(i+K_w)}$ is the concatenation of the word embedding vectors from the position $i - K_w$ to $i + K_w$. $\mathbf{U}_w \in \mathcal{R}^{N_w \times (2K_w+1)D}$ and $\mathbf{b}_w \in \mathcal{R}^{N_w}$ are the parameters of the filters in CNN network, where N_w is the number of CNN filters and $2K_w + 1$ is the window size. ReLU is the nonlinear activation function (Glorot et al., 2011). The output of the CNN layer is a sequence of contextual word representations $[\mathbf{c}_1^w, \mathbf{c}_2^w, ..., \mathbf{c}_M^w]$.

The third layer is a word-level attention network. Different words in the same sentence may have different informativeness for modeling users and items. For example, in the sentence "The laptop I bought yesterday is too heavy", the word "heavy" is more informative than the word "yesterday" in representing this laptop. Thus, we use a word-level attention network to help our model select and attend to important words based on their contextual representations to build more informative sentence representations for user and item modeling. The attention weight of the i_{th} word in the sentence s is computed as follows:

$$a_i^w = \tanh(\mathbf{v}_w \times \mathbf{c}_i^w + b_w), \quad (2)$$

$$\alpha_i^w = \frac{\exp(a_i^w)}{\sum_{j=1}^{M} \exp(a_j^w)}, \quad (3)$$

where $\mathbf{v}_w \in \mathcal{R}^{N_w}$ and $b_w \in \mathcal{R}$ are the parameters in the attention network. α_i indicates the relative importance of the i_{th} word evaluated by the attention network. The final representation of the sentence s is the summation of the contextual word representations weighted by their attention weights as follows:

$$\mathbf{s} = \sum_{i=1}^{M} \alpha_i^w \mathbf{c}_i^w. \quad (4)$$

3.2 Review Encoder

The *review encoder* module aims to build the representations of each review based on the representation of sentences in these reviews. There are two major layers in the *review encoder* module.

The first layer is a sentence-level CNN network. Neighboring sentences usually have some relatedness with each other. For example, in a laptop review "It is not a laptop to game with. But as a working laptop, you will get a great bang for your buck", the two neighboring sentences have close relatedness and they both describe the performance of the laptop in different scenarios.

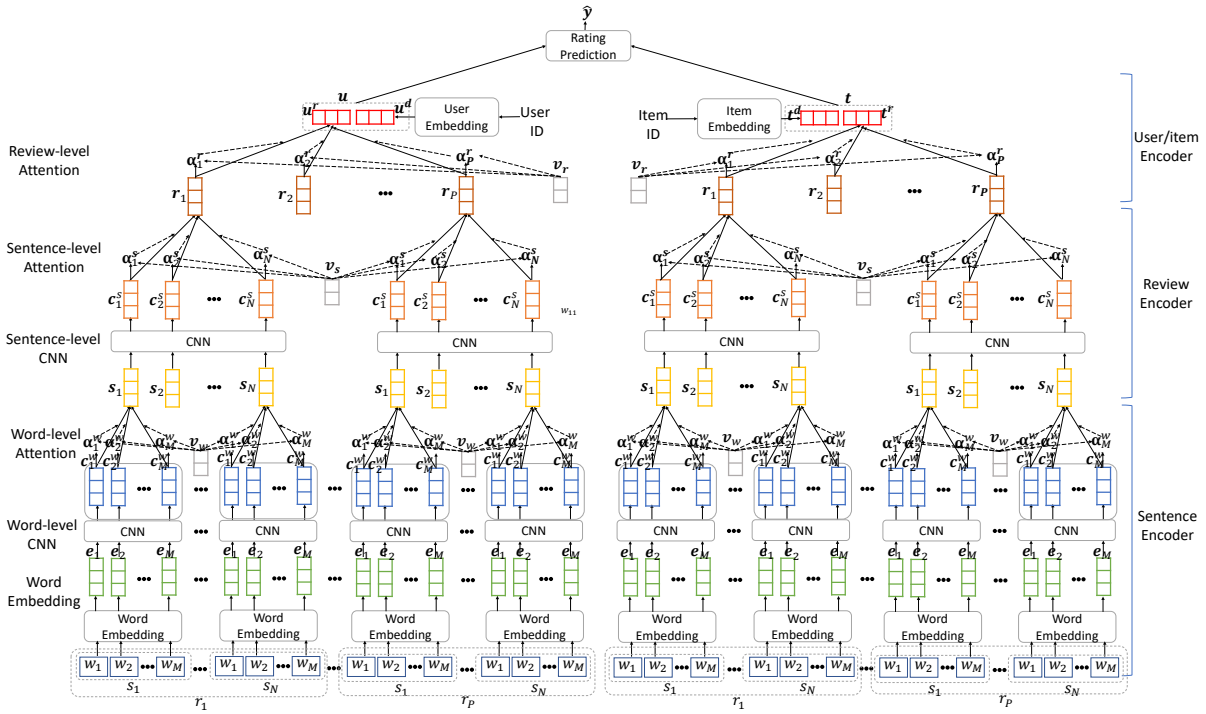


Figure 2: The framework of our *HUITA* approach for recommendation.

Thus, we employ a sentence-level CNN network to learn the contextual sentence representations by capturing the local contexts of sentences. Denote a review r contains N sentences $[s_1, s_2, ..., s_N]$. Denote the contextual representation of sentence s_i as $\mathbf{c}_i^s$, which is computed as follows:

$$\mathbf{c}_i^s = \text{ReLU}(\mathbf{U}_s \times \mathbf{s}_{(i-K_s):(i+K_s)} + \mathbf{b}_s), \quad (5)$$

where $\mathbf{U}_s \in \mathcal{R}^{N_s \times (2K_s+1)N_w}$ and $\mathbf{b}_s \in \mathcal{R}^{N_s}$ are parameters of the sentence-level CNN filters. $\mathbf{s}_{(i-K_s):(i+K_s)}$ is the concatenation of sentence representation vectors from position $i - K_s$ to $i + K_s$. N_s is the number of filters in sentence CNN network and $2K_s + 1$ is the window size.

The second layer is a sentence-level attention network. Different sentences in a review may have different informativeness for modeling users and items. For example, the sentence "it is not a laptop to game with" is more informative than the sentence "I bought this laptop yesterday" in learning the representation of this laptop. Thus, we use sentence-level attention network to help our model select and attend to important sentences to learn more informative review representations. The attention weight of sentence s_i in the review r is formulated as follows:

$$a_i^s = \tanh(\mathbf{v}_s \times \mathbf{c}_i^s + b_s), \quad (6)$$

$$\alpha_i^s = \frac{\exp(a_i^s)}{\sum_{j=1}^{N} \exp(a_j^s)}, \quad (7)$$

where $\mathbf{v}_s \in \mathcal{R}^{N_s}$ and $b_s \in \mathcal{R}$ are the parameters of the attention network. The final contextual representation of the review r is the summation of the contextual representations of sentences weighted by their attention weights, which is formulated as:

$$\mathbf{r} = \sum_{i=1}^{N} \alpha_i^s \mathbf{c}_i^s. \quad (8)$$

3.3 User/Item Encoder

The *user/item encoder* module is used to build the representations of users or items based on the representations of their reviews. Different reviews usually have different informativeness in modeling users or items. For example, in Fig. 1, the first review contains much more information of the laptop than the second review, and should has more contributions in building the representation of this laptop. Thus, we use a review-level attention network to distinguish informative reviews from less informative ones. Denote a user u has P reviews $[r_1, r_2, ..., r_P]$. Then the attention weight of the review r_i is computed as follows:

$$a_i^r = \tanh(\mathbf{v}_r \times \mathbf{r}_i + b_r), \quad (9)$$

$$\alpha_i^r = \frac{\exp(a_i^r)}{\sum_{j=1}^{P} \exp(a_j^r)}, \quad (10)$$

where $\mathbf{v}_r \in \mathcal{R}^{N_s}$ and $b_r \in \mathcal{R}$ are the parameters of the review-level attention network. The user representation learned from the reviews is the summation of the contextual representations of reviews weighted by their attention weights:

$$\mathbf{u}^r = \sum_{i=1}^{P} \alpha_i^r \mathbf{r}_i. \quad (11)$$

Although the user representation $\mathbf{u}^r$ learned from reviews contain rich information of users, there are some latent characteristics of users which are not described in their reviews but can be inferred from the rating patterns. Thus, we also represent users using the embedding of their IDs to capture the latent factors of users, which are motivated by traditional recommendation methods (Koren et al., 2009). The final representation of user u is the concatenation of the user representation $\mathbf{u}^r$ learned from reviews and the user embedding $\mathbf{u}^d$ inferred from user ID, as follows:

$$\mathbf{u} = [\mathbf{u}^r, \mathbf{u}^d]. \quad (12)$$

The representations of items can be computed in a similar way. Denote the representation of item t learned from reviews as $\mathbf{t}^r$, and the item embedding inferred from item ID as $\mathbf{t}^d$. Then the final representation of this item is as follows:

$$\mathbf{t} = [\mathbf{t}^r, \mathbf{t}^d]. \quad (13)$$

3.4 Rating Prediction

In recommender systems the recommendations are made based on the predicted ratings that a user will give to an item. In our *HUITA* approach, the rating score of a user-item pair is predicted based on the representations of users and items as follows:

$$\hat{y} = \mathrm{ReLU}(\mathbf{w}^T(\mathbf{u} \odot \mathbf{t}) + b), \quad (14)$$

where $\odot$ is item-wise dot product, $\mathbf{w}$ and b are parameters in the rating prediction layer.

In the model training stage, we optimize the model parameters to minimize the difference between gold rating and predicted ratings. We use the mean squared error as the loss function:

$$\mathcal{L} = \frac{1}{N_P} \sum_{i=1}^{N_P} (\hat{y}_i - y_i)^2, \quad (15)$$

where N_P denotes the number of user-item pairs in training data, $\hat{y}_i$ and y_i are the predicted rating score and the gold rating score respectively of the i_{th} user-item pair.

Dataset	#users	#items	#reviews
Toys_and_Games	19,412	11,924	167,597
Kindle_Store	68,223	61,935	982,619
Movies_and_TV	123,960	50,052	1,679,533
Yelp 2017	199,445	119,441	3,072,129

Table 1: Statistics of datasets used in our experiments.

4 Experiments

4.1 Datasets and Experimental Settings

We conducted experiments on four widely used benchmark datasets in different domains to evaluate the effectiveness of our approach. Following (Chen et al., 2018), we used three datasets from the Amazon collection[1](He and McAuley, 2016), i.e., **Toys_and_Games**, **Kindle_Store**, and **Movies_and_TV**. Another dataset is from Yelp Challenge 2017[2] (denoted as **Yelp 2017**), which is a large-scale restaurant review dataset. Following (Chen et al., 2018), we only kept the users and items which have at least 5 reviews. The detailed statistics of the four datasets are summarized in Table 1. The ratings in these datasets are in [1, 5].

In our experiments, the dimension of word embeddings was set to 300. We used the pre-trained Google embedding (Mikolov et al., 2013) to initialize the word embedding matrix. The word-level CNN has 200 filters and their window size is 3. The sentence-level CNN has 100 filters with window size of 3. We applied dropout strategy (Srivastava et al., 2014) to each layer of our model to mitigate overfitting. The dropout rate was set to 0.2. Adam (Kingma and Ba, 2014) was used as the optimization algorithm. The batch size was set to 20. We randomly selected 80% of the user-item pairs in each dataset for training, 10% for validation and 10% for test. All the hyperparameters were selected according to the validation set. We independently repeated each experiment for 5 times and reported the average performance in Root Mean Square Error (RMSE).

4.2 Performance Evaluation

We evaluate the performance of our approach by comparing it with several baseline methods. The methods to be compared include:

- *PMF*: Probabilistic Matrix Factorization, which models users and items based on

[1]http://jmcauley.ucsd.edu/data/amazon
[2]https://www.yelp.com/dataset_challenge

	PMF	NMF	SVD++	HFT	DeepCoNN	Attn+CNN	NARRE	HUITA
Rating score	✓	✓	✓	✓	✓	✓	✓	✓
Review text				✓	✓	✓	✓	✓
Word context & order					✓	✓	✓	✓
Review attention							✓	✓
Word/sentence attention						✓*		✓

Table 2: Information used in different methods. *Only word attention is modeled.

Methods	Toys_and_Games	Kindle_Store	Movies_and_TV	Yelp 2017
PMF	1.3076	0.9914	1.2920	1.3340
NMF	1.0399	0.9023	1.1125	1.2916
SVD++	0.8860	0.7928	1.0447	1.1735
HFT	0.8925	0.7917	1.0291	1.1699
DeepCoNN	0.8890	0.7876	1.0128	1.1642
Attn+CNN	0.8805	0.7796	0.9984	1.1588
NARRE	0.8769	0.7783	0.9965	1.1559
HUITA	**0.8649**	**0.7464**	**0.9631**	**1.1246**

Table 3: RMSE scores of different methods on different datasets. Lower RMSE score means better performance.

ratings via matrix factorization (Mnih and Salakhutdinov, 2008).

- *NMF*: Non-negative Matrix Factorization for recommendation based on rating scores (Lee and Seung, 2001).
- *SVD++*: The recommendation method based on rating matrix via SVD and similarities between items (Koren, 2008).
- *HFT*: Hidden Factor as Topic (HFT), a method to combine reviews with ratings via LDA (McAuley and Leskovec, 2013).
- *DeepCoNN*: Deep Cooperative Neural Networks, a neural method to jointly model users and items from their reviews via CNN (Zheng et al., 2017).
- *Attn+CNN*: Attention-based CNN, which uses both CNN and attention over word embeddings to learn user and item representation from reviews (Seo et al., 2017b).
- *NARRE*: Neural Attentional Rating Regression with Review-level Explanations, which uses attention mechanism to model the informativeness of reviews for recommendation (Chen et al., 2018).
- *HUITA*: our proposed hierarchical user and item representation approach with three-tier attention for recommendation with reviews.

In Table 2, we show a simple comparison of different methods in terms of the information considered in each method. Traditional recommendation methods such as *PMF*, *NMF* and *SVD* are solely based on rating scores, and other methods *HFT*, *DeepCoNN*, *Attn+CNN*, *NARRE* and *HUITA* can exploit both rating scores and reviews for recommendation. Among the latter methods, *HFT* is based on topic models and cannot capture the contexts and orders of words. *DeepCoNN* and *Attn+CNN* simply concatenate reviews into a long document, and cannot model the informativeness of different reviews. Although *NARRE* can model review helpfulness via attention, it simply merges all sentences in a review together, and does not model the informativeness of different sentences and words. Different from these methods, our *HUITA* approach learns user and item representations from reviews in a hierarchical manner, and uses a three-tier attention network to select and attend to important words, sentences and reviews.

The results of different methods are shown in Table 3. We have several observations from the results. First, the methods which exploit reviews (i.e., *HFT*, *DeepCoNN*, *Attn+CNN*, *NARRE* and *HUITA*) usually perform better than the methods only based on rating scores (i.e., *PMF*, *NMF* and *SVD++*). It validates reviews can provide rich information of user preferences and item properties, and is important to learn informative user and item representations and can benefit recommendation.

Methods	Toys_and_Games	Kindle_Store	Movies_and_TV	Yelp 2017
All	0.8649	0.7464	0.9631	1.1246
-word attention	0.8721	0.7610	0.9744	1.1337
-sentence attention	0.8714	0.7569	0.9715	1.1308
-review attention	0.8685	0.7523	0.9720	1.1294

Table 4: The effectiveness of different levels of attentions. The evaluation metric is RMSE.

Second, among the method which can exploit reviews, the neural network based methods (e.g., *DeepCoNN*, *Attn+CNN*, *NARRE* and *HUITA*) usually outperform the *HFT* method which is based on topic models. This is probably because in *HFT* the reviews are represented using bag-of-words features, and the contextual information and the orders of words are lost. This result validates the neural network based method can better capture the semantic information in reviews to model users and items for recommendation.

Third, the methods considering review helpfulness (i.e., *NARRE*) and word importance (i.e., *Attn+CNN*) usually outperform *DeepCoNN*. This result implies that different words and different reviews have different importance for modeling users and items from reviews. Distinguishing important reviews and words from less important ones is beneficial to learn more accurate user and item representations for recommendation.

Fourth, our approach can consistently outperform all the baseline methods compared here. This is because different from baseline methods such as *Attn+CNN* and *DeepCoNN* which merge all reviews into a long document and *NARRE* which merges all sentences into a long sentence, our *HUITA* approach learns user and item representations in a hierarchical manner. *HUITA* first learns sentence representations from words, then learns review representations from sentences, and finally learns user/item representations from reviews. Besides, our approach incorporates a three-tier attention network to jointly select and attend to important words, sentences and reviews. Thus, our approach can learn more informative user and item representations from reviews for recommendation.

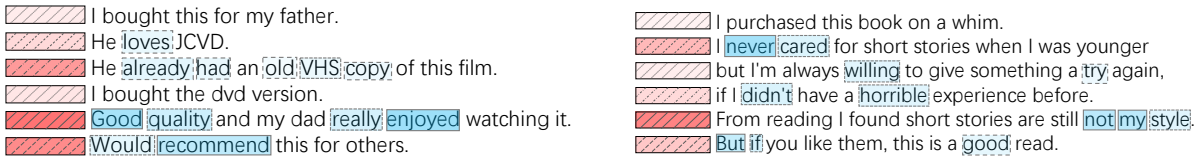
4.3 Effectiveness of Three-Tier Attention

In this section, we conducted experiments to explore the effectiveness of the three-tier attention network in our approach. We compare three variants of our model by removing one kind of attention each time to evaluate its contribution to the performance. The results are shown in Table 4.

According to Table 4, the word-level attention can effectively improve the performance of our approach. This is because different words in reviews have different importance in modeling users and items. Therefore, recognizing and highlighting the important words using the word-level attention network can help learn more informative sentence representations. In addition, the sentence-level attention is also useful. This may be because different sentences have different informativeness. For example, in a laptop review the sentence "this laptop is expensive" is more informative than "I bought this laptop yesterday" in representing this laptop. The sentence-level attention network can help to select important sentences to build review representations. Besides, the review-level attention is also useful in our *HUITA* approach. This is because different reviews have different informativeness in representing users and items. And distinguishing informative reviews from the less informative ones can help learn more accurate representations of users and items. Moreover, combining all the three levels of attentions can further improve the performance of our approach, which validates the effectiveness of our three-tier attention architecture.

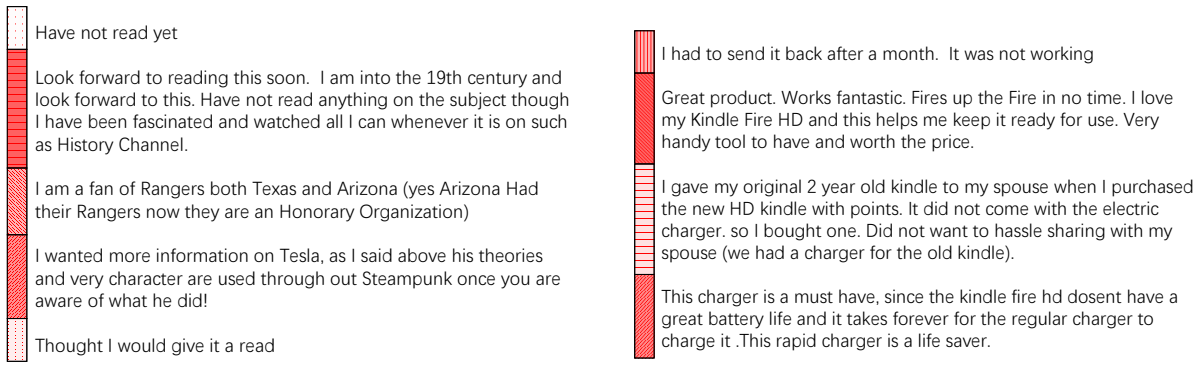
4.4 Case Study

In this section, we conducted several case studies to further explore whether our approach can select informative words, sentences and reviews to learn informative user and item representations for recommendation. First, we want to explore the effectiveness of the word- and sentence-level attention networks. The visualization of the attention weights in the word- and sentence-level attention networks is shown in Fig. 3. From Fig. 3 we can see that our word-level attention network can effectively select and attend to important words. For example, in Fig. 3(a) the words "Good", "quality" and "recommend" are assigned higher attention weights than "bought" and "dad", since "Good", "quality" and "recommend" can better model the properties of the film. In addition, our model can



(a) Movies_and_TV (b) Kindle_Store

Figure 3: Visualization of attention weights in two randomly selected reviews from the Movies_and_TV and Kindle_Store datasets respectively. Red boxes to the left of the reviews represent sentence-level attention weights, and blue boxes on the individual words represent word-level attention weights. Darker color represents higher attention weights.



(a) User (b) Item

Figure 4: Visualization of attention weights of reviews from a randomly selected user and item in the Kindle_Store dataset. Vertical bars represent review-level attention weights and darker color represents higher attention weights.

effectively select informative sentences using the sentence-level attention network. For example, in Fig. 3(b) the sentence "From reading I found short stories are still not my style" is assigned high attention weight since it is informative for representing this user and is important for recommendation, while the sentence "I purchased this book on a whim" has low attention weight since it contains limited information of users and items. Thus, these results validate that our approach is effective in selecting informative words and sentences in reviews for recommendation through the word- and sentence-level attention networks.

Second, we want to explore the effectiveness of the review-level attention in our *HUITA* approach. The visualization of the review-level attention weights is shown in Fig. 4. From Fig. 4 we can see that our approach can effectively select and attend to informative reviews. For example, the second review in Fig. 4(a) is assigned high attention weight by our approach since it reveals rich information of user preferences. However, the first review in Fig. 4(a) receives low attention weight since it contain limited information of users. Thus, these results validate the effectiveness of our approach in selecting informative reviews to learn more accurate representations of users and items from reviews for recommendation.

5 Conclusion

In this paper, we propose a hierarchical user and item representation model with three-tier attention to learn user and item representations from reviews for recommendation. In our approach, we use a sentence encoder to learn sentence representations from words, a review encoder to learn review representations from sentences, and a user/item encoder to learn user/item representations from reviews. In addition, we incorporate a three-tier attention network into our model to select and attend to informative words, sentences and reviews to learn more accurate representations of users and items. Besides, we combine the user and item representations learned from the reviews with the embeddings of user and item IDs as the final representations of users and items to capture the latent factors of individual users and items. The experiments on four benchmark datasets validate that our approach can effectively improve the performance of recommendation and consistently

outperform many baseline methods.

Acknowledgments

This work was supported by the National Key Research and Development Program of China under Grant number 2018YFC1604002, and the National Natural Science Foundation of China under Grant numbers U1836204, U1705261, U1636113, U1536201, and U1536207.

References

Yang Bao, Hui Fang, and Jie Zhang. 2014. Topicmf: Simultaneously exploiting ratings and reviews for recommendation. In *AAAI*, volume 14, pages 2–8.

Rose Catherine and William Cohen. 2017. Transnets: Learning to transform for recommendation. In *RecSys*, pages 288–296. ACM.

Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. 2018. Neural attentional rating regression with review-level explanations. In *WWW*, pages 1583–1592.

Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J Smola, Jing Jiang, and Chong Wang. 2014. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *KDD*, pages 193–202.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 315–323.

Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *WWW*, pages 507–517.

Xiangnan He, Tao Chen, Min-Yen Kan, and Xiao Chen. 2015. Trirank: Review-aware explainable recommendation by modeling aspects. In *CIKM*, pages 1661–1670.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *KDD*, pages 426–434.

Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer*, (8):30–37.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature*, 521(7553):436.

Daniel D Lee and H Sebastian Seung. 2001. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562.

Guang Ling, Michael R Lyu, and Irwin King. 2014. Ratings meet reviews, a combined approach to recommend. In *RecSys*, pages 105–112.

Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *RecSys*, pages 165–172.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119.

Andriy Mnih and Ruslan R Salakhutdinov. 2008. Probabilistic matrix factorization. In *NIPS*, pages 1257–1264.

Zhaochun Ren, Shangsong Liang, Piji Li, Shuaiqiang Wang, and Maarten de Rijke. 2017. Social collaborative viewpoint regression with explainable recommendations. In *WSDM*, pages 485–494.

Sungyong Seo, Jing Huang, Hao Yang, and Yan Liu. 2017a. Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In *RecSys*, pages 297–305. ACM.

Sungyong Seo, Jing Huang, Hao Yang, and Yan Liu. 2017b. Representation learning of users and items for review rating prediction using attention-based convolutional neural network. In *MLRec*.

Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15(1):1929–1958.

Yunzhi Tan, Min Zhang, Yiqun Liu, and Shaoping Ma. 2016. Rating-boosted latent topics: Understanding users and items with ratings and reviews. In *IJCAI*, pages 2640–2646.

Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2018. Multi-pointer co-attention networks for recommendation. In *KDD*, pages 2309–2318.

Wei Zhang, Quan Yuan, Jiawei Han, and Jianyong Wang. 2016. Collaborative multi-level embedding learning from reviews for rating prediction. In *IJCAI*, pages 2986–2992.

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 *SIGIR*, pages 83–92.

Lei Zheng, Vahid Noroozi, and Philip S Yu. 2017. Joint deep modeling of users and items using reviews for recommendation. In *WSDM*, pages 425–434. ACM.