CCF Transactions on Pervasive Computing and Interaction (2019) 1:285–295
https://doi.org/10.1007/s42486-019-00023-0

REGULAR PAPER

Helpfulness-aware review based neural recommendation

Suyu Ge[1] · Tao Qi[1] · Chuhan Wu[1] · Fangzhao Wu[2] · Xing Xie[2] · Yongfeng Huang[1]

Received: 1 July 2019 / Accepted: 4 November 2019 / Published online: 14 November 2019

Abstract
Reviews contain rich information of user interests and item characteristics. Incorporating reviews into recommendation has attracted increasing attention in recent years, which can help learn more accurate user and item representations for recommendation. Existing review based recommendation methods usually utilize the content of reviews while ignoring the helpfulness scores associated with them. Since different reviews have different informativeness and many reviews are noisy and even misleading, incorporating the helpfulness information of reviews can help better exploit the reviews for recommendation. In this paper, we propose a helpfulness-aware review based recommendation approach. The core of our approach is a review encoder and a user/item encoder. In the review encoder we learn representations of reviews from their content in a hierarchical way. We first learn sentence representations from words and then learn review representations from sentences, using a hierarchical attention network to select important words and sentences. In the user/item encoder, we learn representations of users/items from their reviews using an attention network. The query vector of the attention network comes from the helpfulness scores of these reviews. Since many reviews do not have helpfulness scores, we propose a neural helpfulness prediction model to predict the helpfulness scores of these reviews. The neural helpfulness prediction model is trained on the limited reviews with helpfulness scores voted by users. Extensive experiments on four benchmark datasets show that incorporating the helpfulness of reviews can effectively improve the performances of review based neural recommendation methods.

Keywords Recommender systems · Review helpfulness · Hierarchical attention network

1 Introduction

Recommender systems (RS) are very important for alleviating information overload in many online shopping platforms such as Amazon (McAuley and Leskovec 2013; Bao et al. 2014). A core task in recommender systems is learning accurate representations of users and items to capture user interests and item characteristics (Cheng et al. 2017). Existing recommendation methods usually rely on the rating matrix to learn user and item representations (Koren 2008; Mnih and Salakhutdinov 2008; Koren et al. 2009). For example, classic Matrix Factorization (MF) method such as Functional matrix factorizations (FMF) Zhou et al. (2011) constructs a decision tree from user profiles for matrix calculation. However, since the rating matrix between users and items are usually very sparse, it is very challenging to learn accurate user and item representations from them for recommendation (Luo et al. 2015; Bell et al. 2007).

Besides the ratings, in many e-commerce platforms there are also reviews posted by users to express their opinions towards the items (Mudambi and Schuff 2010; Bhatt et al. 2015). These reviews are usually in large quantities and contain rich information of users and items (He et al. 2015). For example, Hidden Factors as Topics (HFT) McAuley and Leskovec (2013) employed topic modeling technique to capture latent feature from user reviews and calculated similarities between matrices. Upon it, Ratings Meet Reviews

✉ Suyu Ge
gesy17@mails.tsinghua.edu.cn

Tao Qi
qit16@mails.tsinghua.edu.cn

Chuhan Wu
wuch15@mails.tsinghua.edu.cn

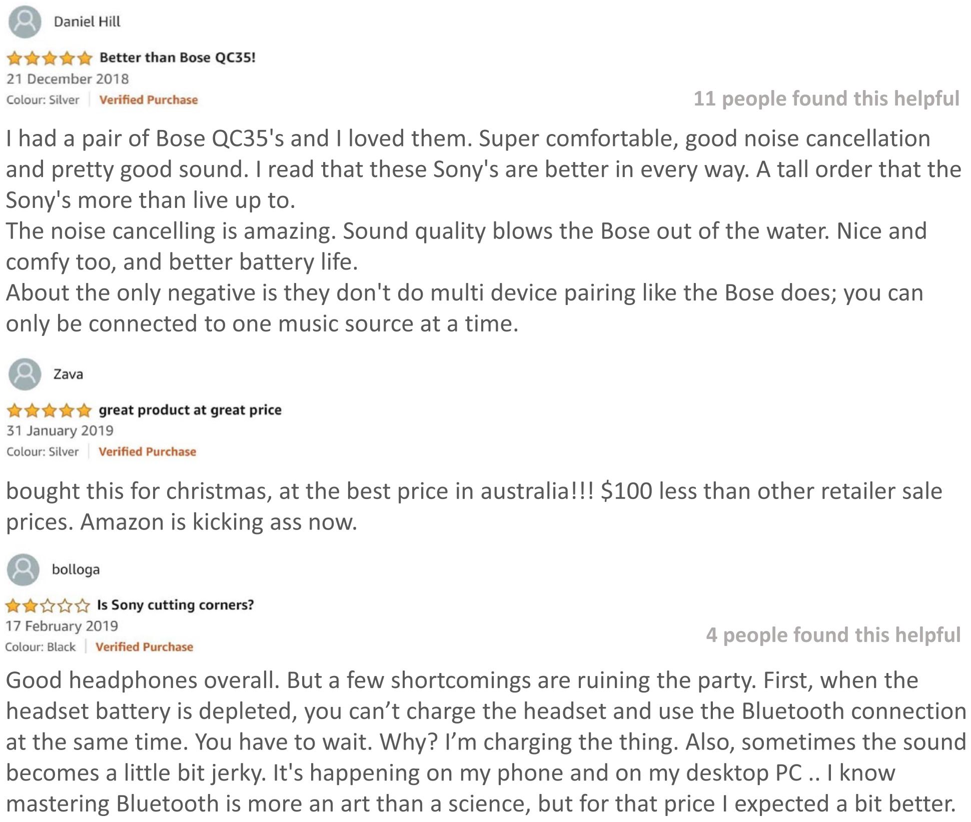
Fangzhao Wu
fangzwu@microsoft.com

Xing Xie
xing.xie@microsoft.com

Yongfeng Huang
yfhuang@tsinghua.edu.cn

[1] Tsinghua University, Beijing, China

[2] Microsoft Research Asia, Beijing, China



Daniel Hill

★★★★★ Better than Bose QC35!

21 December 2018

Colour: Silver | Verified Purchase

11 people found this helpful

I had a pair of Bose QC35's and I loved them. Super comfortable, good noise cancellation and pretty good sound. I read that these Sony's are better in every way. A tall order that the Sony's more than live up to.
The noise cancelling is amazing. Sound quality blows the Bose out of the water. Nice and comfy too, and better battery life.
About the only negative is they don't do multi device pairing like the Bose does; you can only be connected to one music source at a time.

Zava

★★★★★ great product at great price

31 January 2019

Colour: Silver | Verified Purchase

bought this for christmas, at the best price in australia!!! $100 less than other retailer sale prices. Amazon is kicking ass now.

bolloga

★★☆☆☆ Is Sony cutting corners?

17 February 2019

Colour: Black | Verified Purchase

4 people found this helpful

Good headphones overall. But a few shortcomings are ruining the party. First, when the headset battery is depleted, you can't charge the headset and use the Bluetooth connection at the same time. You have to wait. Why? I'm charging the thing. Also, sometimes the sound becomes a little bit jerky. It's happening on my phone and on my desktop PC .. I know mastering Bluetooth is more an art than a science, but for that price I expected a bit better.

Fig. 1 Example reviews for a type of earphone from Amazon. The first and second reviews are both five starts, but only the first one is considered as helpful by other consumers. The third one is a negative review, but it is accredited by four people and labeled as helpful

(RMR) Ling et al. (2014) harnessed the information of both ratings and reviews by aligning topic modeling to rating dimensions in a unified model.

In addition, neural methods incorporating reviews to learn more accurate user and item representations for recommendation have attracted huge attentions in recent years, such as DeepCoNN Zheng et al. (2017) and Transnets Catherine and Cohen (2017). They achieved promising results by automatically forming review representation using convolutional neural networks (CNN). Based on them, many other CNN based approaches were proposed (Seo et al. 2017a; Bao et al. 2014; Zheng et al. 2017). Similar with DeepCoNN, they mainly utilized rating scores and reviews, but made improvements upon DeepCoNN by adjusting network architecture. However, these existing review based recommendation methods usually rely on the content of reviews to learn user and item representations, and the helpfulness scores of these reviews voted by massive users are ignored. Thus, the quality of reviews (e.g., whether a review is reliable or helpful) is not considered in previous work.

Our work is motivated by following observations. First, different reviews have different informativeness in representing users and items. In Fig. 1, we show reviews about one type of Sony earphone collected from Amazon. The first and second reviews are all five stars, but the first one is more informative while the second one only mentioning the cheap price. Second, many reviews on e-commerce platforms are associated with their helpfulness information. For example, in Fig. 1, the first review is accredited by 11 users, indicating it as a highly helpful review. Also, the third review discloses one disadvantage of the earphone and is labeled by four people as helpful. Thus, helpful positive and negative reviews are highlighted in this user oriented way. Third, the helpfulness scores of reviews provided by massive users can provide important clues of review informativeness. For example, some e-commerce platforms such as Amazon usually display reviews owning the most helpfulness votes as important additional features of the products. Fourth, not all reviews have sufficient votes to compute the helpfulness score. In Fig. 1, the second review has no helpfulness vote from users. The same situation applies to massive reviews since the quantity of reviews far exceeds the number of users. Thus, it is not practical to utilize the helpfulness scores of all reviews as model inputs for review based recommendation.

In this paper, we propose a neural **Helpfulness-aware review recommendation** (**HARR**) model. Our model consists of three components, i.e., *a review encoder*, *a user/item encoder* and *a rating predictor*. We use the review encoder to generate review representations. The review encoder forms review representations in a hierarchical way. Motivated by Yang et al. (2016), our model first forms sentence representations from words and then learns review representations from sentences. We use attention mechanism to selectively pertain important words and sentence embeddings. Since many reviews don't have helpfulness scores voted by users, we propose a helpfulness predictor based on the review encoder. The module utilizes the output of review encoder to predict the helpfulness scores of reviews. After review representation and its helpfulness score is generated, we send them as input for the user/item encoder. All reviews written by a user are used to form user representation. For a particular item, we take all reviews written towards them by different people. Similarly, the user/item encoder also utilizes attention mechanism to balance reviews according to their helpfulness scores. To meet this end, we propose an attention network with query vector, which takes the helpfulness score as the attention weight of this review. The final user/item representation is the weighted combination of reviews vectors. Finally, simple dot product is used to calculate similarity between a user vector and an item vector to obtain the final rating score. We conduct extensive experiments on four benchmark datasets. The results show our approach can effectively improve the performance of recommendation while at the same time producing applicable review helpfulness score.

2 Related work

2.1 Review based recommender systems

Introducing online reviews to improve rating prediction has been extensively studied and justified in many pioneering works (McAuley and Leskovec 2013; Kim et al. 2016; Almahairi et al. 2015; Tan et al. 2016; Ling et al. 2014). Both non-neural collaborative filtering and deep neural networks have been adopted to capture the semantic meaning of reviews. They are effective in mitigating the cold-start issue and enriching user or item representations (Park and Chu 2009; Zhou et al. 2011). Among non-neural methods, classic Matrix Factorization (MF) methods such as Hidden Factors as Topics (HFT) employed topic modeling technique to capture latent feature from user reviews and calculated similarities between matrices. It was enhanced by TopicMF to a model using simultaneously user and item reviews (Bao et al. 2014). Additionally, TopicMF extracts information using non-negative matrix factorization (NMF). Upon them, Ratings Meet Reviews (RMR) Ling et al. (2014) harnessed the information of both ratings and reviews by aligning topic modeling to rating dimensions in a unified model. However, these MF based methods involved laborious feature engineering which reduced their practicality. Moreover, the review representations they form lack context information, thus cannot capture correlated sentence meaning.

The recent shift towards deep learning models is prominent, which learns reviews in a passage context. Among all neural approaches, DeepCoNN (Zheng et al. 2017) first validated the competitive ability of CNN in review information extraction. They designed two paralleled CNN networks to encode user behaviors and item properties from reviews, on the top of which a shared layer was introduced to couple the two in a way similar with factorization machine (FM). It outperformed traditional MF models tremendously. Many subsequent research followed it, using CNN as encoder, but with model architecture in variation. Transnets (Catherine and Cohen 2017) extended DeepCoNN by introducing interaction between source network and target network, minimizing the loss between the above two to enforce the model to learn accurate user-item pair representation. They both reached superior results on benchmark Amazon dataset but they were not with flaws. Most importantly, the two methods were noise disturbing, lacking dynamic variation between review helpfulness. To solve the problem, ATRank (Seo et al. 2017b) was introduced to deal with user preference diversity in a bottom level, forming fine-grained sentence representations by adding extra word-level attention mechanism. D-ATT (Seo et al. 2017a) extended the work by applying dual local and global attention, encoding both local user-item properties and overall review context meanings. Afterwards, (Tay et al. 2018) proposed a Multi-pointer Co-attention networks. The multi-pointer model was differential with a gumbel–softmax based pointer mechanism. Chen et al. (2018c) proposed Neural Attentional Rating Regression (NARRE), it used the same attention mechanism and conducted experiment to prove the attention weight indeed selected informative words.

The above models only took user and item relation into account while ignoring whether a review is helpful in user or item modeling. As a result, they were still easily misled by noise and redundant messages. We argue that different reviews should be aligned varied importance when forming user or item representation. Meanwhile, The helpfulness score voted by users on E-commerce platform can serve this end. The only problem of introducing them into the network is that not all reviews have helpfulness scores. Thus our method is proposed to predict helpfulness scores and takes it as inputs for product recommendation task. In architecture, our model is fundamentally different with all the other methods, setting review helpfulness prediction as preliminary goal first, then using the helpfulness score outputs to generate more accurate user/item representations. In implementation, the hierarchical attention network we use is capable of forming both comprehensive passage-level and fine-grained sentence-level representation. In motivation, our model innovatively introduces an approach to perform product recommendation and product review helpfulness prediction.

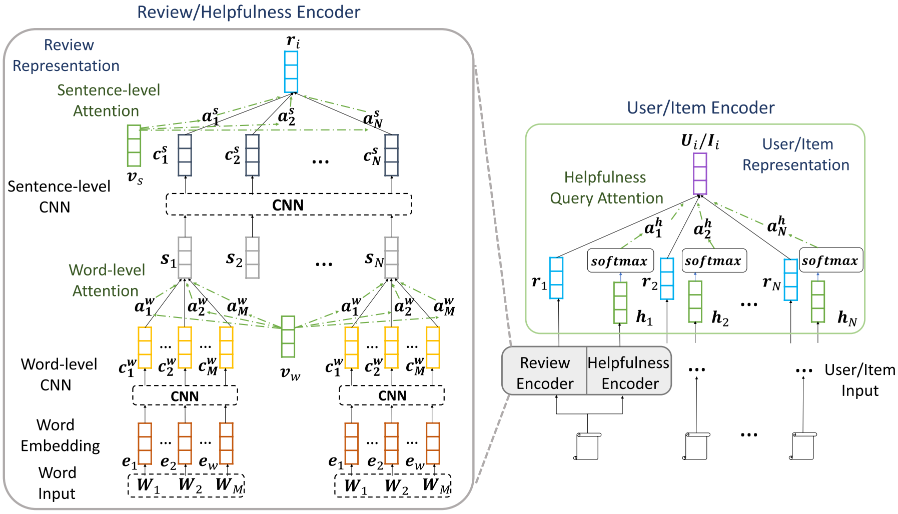


Fig. 2 The architecture of our *HARR* approach

2.2 Review helpfulness prediction

Initially, traditional machine learning classifier such as support vector machine (SVM) was used to predict review helpfulness based on a list of features, e.g., structural, lexical, syntactic and meta-data features (Kim et al. 2006; Ngo-Ye and Sinha 2014; Krishnamoorthy 2015; Qazi et al. 2016). Recently, neural based methods improve the task a great deal by using convolutional encoder to form representation from review texts (Kim 2014; Zhang et al. 2015; Chen et al. 2018a). They also boosted efficiency by cross-domain learning, adding discriminated factors in loss function (Chen et al. 2018b; Liu et al. 2017).

Different from their works, we build a hierarchical predictor based on review texts. What should be clarified is that we do not set helpfulness prediction as the ultimate goal of our network, but an intermediate output, serving as query vector for user/item representations. The hierarchical helpfulness predictor shares the same architecture with the hierarchical review encoder model, but with independent parameters.

3 Our approach

In this section, we introduce our HARR approach in detail. It uses a hierarchical attention network to form review representation and generates both helpfulness and rating scores. HARR network is composed of three modules, i.e., a *review/helpfulness encoder*, a *user/item encoder* and a *rating predictor*. The three sub-modules form the network from bottom to top. We will introduce them accordingly as follows.

3.1 Review/helpfulness encoder

The architectural overview of this module is shown in the left side of Fig. 2. It is used to learn the latent representation of one review. As is shown in Fig. 2, a sentence-level encoder and passage-level encoder with similar layers are deployed in this module. After generating review-level representation, the model predicts the helpfulness score of each review.

3.1.1 Sentence-level encoder

The *sentence-level encoder* consists of three layers. The first layer is word embedding. It converts a sequence of words into a sequence of low-dimensional dense vectors which contain semantic information of these words. Denote a sentence s containing M words as $[w_1, w_2, ..., w_M]$, it is transformed to high dimensional vectors $[\mathbf{e_1}, \mathbf{e_2}, ..., \mathbf{e_M}]$ via the embedding matrix $\mathbf{E} \in \mathbb{R}^{V \times D}$. 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 word-level convolutional neural network (CNN). Since review texts are usually noisy and the informative sections are several key words combined together rather than a complete sentence, it is not worth the price to use RNN to encode sequence information. For example, through the expression “La La Land is driving me crazy!”, “La La Land” is recognized as the name of a movie rather than a place name. Therefore, 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:

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

where $\mathbf{U}_w \in \mathbb{R}^{N_w \times (2K_w+1)}$ and $\mathbf{b_w} \in \mathbb{R}^{N_w}$. $\mathbf{w}_{(i-K_w):(i+K_w)}$ is the combination of sentence representation vectors from position $i - K_w$ to position $i + K_w$ and g is the activation function which is ReLU (Glorot et al. 2011) in our approach. The contextual representation of the i_{th} word is the concatenation of outputs of multiple filters at position i, which is denoted as $\mathbf{c}_i \in \mathbb{R}^{N_w}$, where N_w is the number of filters and $2K_w + 1$ is the window size.

The third layer of *sentence-level encoder* is an attention network. Different words in the same review sentence may have different importance for users' rating. For example, in the sentence "I find the earphone extremely light and comfortable", the word "light" and "comfortable" may be more informative than "find" in inferring selling point of the earphone. In addition, the same word may have different informativeness in other review sentences. For instance, the same word "light" in the sentence "Both light and dark color laptops look fashionable" is less informative. Hence, to automatically select useful words of a sentence in different contexts, our model use a word-level additive attention network. The attention weight α_i^w of the contextual representation $\mathbf{c}_i^w$ for the i_{th} word in sentence s is computed as follows:

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

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

where $\mathbf{v}_w \in \mathbb{R}^{N_w}$ and $b_w \in \mathbb{R}$ are the parameters of the attention network. α_i^w indicates the relative importance of the i_{th} word evaluated by the attention network. The final attention layer output is the summation of contextual word representation weighted by their attention weight:

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

3.1.2 Passage-level encoder

Reviews are noisy, some sentences express the functional feature of products while the others are only vent of emotions, thus the difference across sentences in the same review should be captured. For instance, in the review "I had a pair of Bose and I loved them. Super comfortable, good noise cancellation and pretty good sound.", only the second sentence is indicative for the earphone's good quality. To meet this end, we apply a two-layer passage-level encoder to take the sentence relatedness and their varied importance into consideration. A CNN layer is utilized as the first layer to encode sentences in the same review. In the above review, expressions "comfortable", "noise cancellation" describe "a pair of Bose", so encoding the two neighbour sequences is essential to form more complete review representation. Denote a review $\mathbf{r}$ contains N sentences $[\mathbf{s}_1, \mathbf{s}_2, ...\mathbf{s}_N]$, the contextual representation of review $\mathbf{r}_i$ as $\mathbf{c}_i^s$, which is computed as follows:

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

where $\mathbf{U}_s \in \mathbb{R}^{N_s \times (2K_s+1)}$ and $b_s \in \mathbb{R}^{N_s}$. $\mathbf{s}_{(i-K_s):(i+K_s)}$ is the combination of sentence representation vectors from position $i - K_s$ to position $i + K_s$, and g is the ReLU activation function. N_s is the number of filters and $2K_s + 1$ is the window size.

The second layer of our *passage-level encoder* is a passage-level additive attention network. As stated above, the attention layer is used to attend to select the most informative sentences in a review. The attention weight α_i^s of contextual sentence representation c_i^s in review r is computed as follows:

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

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

where $\mathbf{v}_s \in \mathbb{R}^{N_s}$ and $b_s \in \mathbb{R}$ are the parameters of the attention network. α_i^s indicates the relative importance of the i_{th} sentence evaluated by the attention network. The final attention layer output is the summation of contextual sentence representation weighted by their attention weight, which is computed as:

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

3.1.3 Helpfulness prediction

For each review r_i, we predict the helpfulness score of it in a straight-forward way, defining it as a binary classification task (namely, *helpful* or *unhelpful*). We use a softmax layer to compute the probabilities of r_i being a helpful review, which is formulated as follows:

$$\mathbf{h}_i = softmax(\mathbf{W}^T \times \mathbf{r}_i + \mathbf{b}_h), \tag{9}$$

where $\mathbf{h}_i$ is the predicted label, $\mathbf{W} \in \mathbb{R}^{N_h}$ and $\mathbf{b}_h \in \mathbb{R}^C$ are parameters of helpfulness prediction layer and C is the number of categories, which is two in our approach.

During model training, crossentropy is used as the loss function, and the overall objective function is formulated as follows:

$$\mathcal{L} = -\sum_{q=1}^{Q} \sum_{c=1}^{C} h_{q,c} \log \hat{h}_{q,c}, \tag{10}$$

where $\hat{h}_{q,c}$ and $h_{q,c}$ are the predicted and gold probability of the q_{th} review in the c_{th} category. Q is the number of reviews to train.

In prediction stage, the label with the largest score in $\mathbf{h}_i$ determines whether a review is helpful or not.

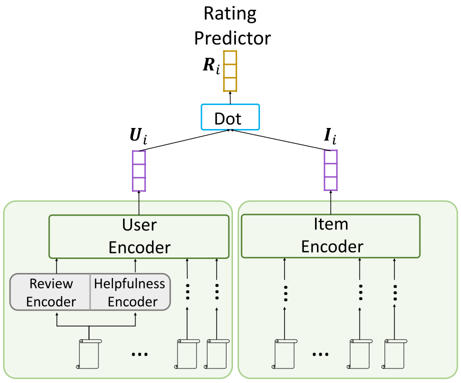


Fig. 3 The overview of our HARR approach

3.2 User/item encoder

Motivated by the observation that review helpfulness scores contain rich information about the validness of the review text, we design a mechanism to form user/item representation using their helpfulness score as attention weights. For instance, in Fig. 1, the first review is more detailed than the second, and earns more helpful scores accordingly, thus should be given higher weight. The user/item encoder forms the top level module of our HARR method, which is illustrated in Fig. 3. After generating helpfulness score vector $\mathbf{h}_i$, we pass it to a fully connected layer and then utilize it as the query vector in both user- and item-level attention networks. Denote a user $\mathbf{u}$ has P reviews $[\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_P]$. The attention weight α_i^h of the i_{th} review $\mathbf{r}_i$ from user $\mathbf{u}$ is calculated as:

$$\mathbf{h'}_i = \mathbf{W}_t \times \mathbf{h}_i + \mathbf{b}_t, \tag{11}$$

$$a_i^h = \mathbf{h'}_i^T \tanh(\mathbf{v}_h^T \times \mathbf{r}_i + b_h), \tag{12}$$

$$\alpha_i^h = \frac{\exp(a_i^h)}{\Sigma_{j=1}^P \exp(a_j^h)}, \tag{13}$$

where $\mathbf{W}_t$ and $\mathbf{b}_t$ are parameters of the fully connected layer, and $\mathbf{h'}_i$ is the transformed helpfulness vector. $\mathbf{v}_h \in \mathbb{R}^{N_s}$ and $b_h \in \mathbb{R}$ are parameters of the query attention network. The final query attention output is the summation of all review representations weighted by their attention weights, which is denoted as:

$$\mathbf{u} = \sum_{i=1}^{P} \alpha_i^h \mathbf{r}_i. \tag{14}$$

For item $\mathbf{i}$, the calculation process is the same with user $\mathbf{u}$, thus we omit it for brief.

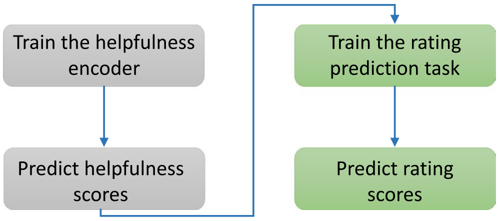


Fig. 4 The training pipeline of HARR. The network parameters of the two tasks are independent

3.3 Rating prediction and model training

In our HARR approach, the rating score of a user-item pair is predicted based on the representations of user $\mathbf{u}$ and item $\mathbf{i}$ as follows:

$$\hat{y} = g(\mathbf{U}_r^T(\mathbf{u} \odot \mathbf{i}) + b_r), \tag{15}$$

where $\odot$ is item-wise dot product, g is the ReLU activation function and $\mathbf{U}_r$ and b_r are parameters in the rating prediction layer. In the model training stage, we optimize parameters by minimizing the difference between gold scores and predicted scores. Mean squared error (MSE) is used as the loss function:

$$\mathcal{L} = \frac{1}{N_p} \sum_{i=1}^{N_p} (\hat{y}_i - y_i)^2, \tag{16}$$

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

In the network training process, we design a pipeline to train our network for the two tasks (e.g., helpfulness prediction and rating prediction), which is illustrated in Fig. 4. In our implementation, the helpfulness encoder and review encoder in Fig. 3 share the same model architecture, but with independent parameters. We first train the helpfulness encoder on a limited review corpus with fully labelled helpfulness scores. Then we utilize the helpfulness encoder to predict helpfulness scores for the entire review corpus. Finally, the predicted helpfulness scores are employed as query attention for the subsequent rating prediction task.

4 Experiments

4.1 Datasets and experimental settings

We conduct experiments on four benchmark datasets in different domains from the Amazon Product Review corpus.[1]

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

Table 1 Statistics of our dataset

Dataset	# Users	# Items	# Reviews	# Helpfulness
Movies	123,960	50,052	1,697,533	213,606
Electronics	192,403	63,001	1,689,188	107,502
CDs	75,258	64,443	1,097,592	164,635
Toys	19,412	11,924	167,597	8091

Helpfulness denotes the number of valid reviews with more than 10 votes, which we use to train for helpfulness prediction task

Table 2 Classification results of the auxiliary helpfulness prediction task

Dataset	Precision	Recall	F1 score
Movies	0.9338	0.8655	0.8984
Electronics	0.8884	0.8564	0.8721
CDs	0.8728	0.9075	0.8898
Toys	0.8689	0.9056	0.8869

High F1 scores on all datasets ensure the accuracy of helpfulness score inputs in the rating prediction task

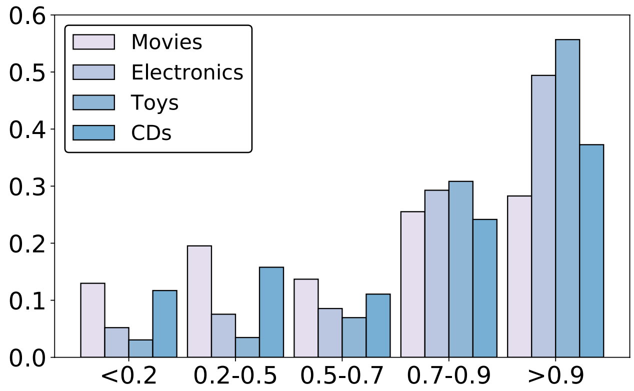


Fig. 5 Distributions of Helpfulness scores

i.e., *Movies and TV*, *Electronics*, *CDs and Vinyl* and *Toys and Games*. The detailed statistics of the four dataset are summarized in Table 1. We only pertain items and users which have at least 5 reviews. Among these reviews, only ones labeling by at least 10 people as helpful or unhelpful are regarded as training samples for helpfulness prediction task. For helpfulness level partition, we use the widely adopted "*a in b*" definition (Singh et al. 2017; Krishnamoorthy 2015; Liu et al. 2008; Tang et al. 2013). E.g., the helpfulness score is the percentage of people who vote for "helpful" against the total number of votes, which is in the range [0, 1]. Additionally, we analyze the helpfulness score distribution of these reviews in Fig. 5 and discover that the distribution varies significantly on different datasets. For instance, the distribution of the helpfulness scores in the "Movies" dataset is relatively uniform, while there is a high portion of helpful reviews in the "Toys" dataset. To help our model better distinguish between helpful and less helpful reviews, we select only the extreme helpful (*helpfulness score* > 0.9) and unhelpful (*helpfulness score* < 0.2) reviews in training time. We then train a binary model to predict helpfulness score on a complete dataset for the main rating prediction task.

In our experiments, the dimension of word embeddings was set to 300. We used the pretrained Google embedding (Mikolov et al. 2013) to initialize the word embedding matrix. Hyperparameters were selected according to performances on validation set. The word-level CNN had 200 filters and the window size was 3. The sentence-level CNN had 100 filters with window size of 3. We empirically found performance bound for maximum sentence length, review length, user/item length is 40 per sentence, 15 per review, 25 per user and 50 per item respectively. We applied dropout strategy (Srivastava et al. 2014) after all convolutional and dense layer 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. We independently repeated each experiment for 5 times and reported the average performance, using the standard Root Mean Square Error (RMSE) as the evaluation metric.

4.2 Performance evaluation

First, we report binary classification scores on the helpful and unhelpful categories in Table 2. It should be concluded that we reach high F1 scores on both categories, which ensures the accuracy of our subsequent task of rating prediction.

Then we will evaluate the performance of our approach by comparing it with several baseline methods. The methods to be compared include: (1) **PMF** (Mnih and Salakhutdinov 2008), probabilistic matrix factorization; (2) **NMF** (Lee and Seung 2001), non-negative matrix factorization; (3) **HFT** (McAuley and Leskovec 2013), hidden factor as topic; (4) **DeepCoNN** (Zheng et al. 2017), deep cooperative neural networks; (5) **D-ATT** (Seo et al. 2017a), dual attention CNN model; It designs a local and global attention mechanism to form review representations respectively; (6) **TransNets** (Catherine and Cohen 2017), an improved method of DeepCoNN by regularizing the representation layer to be similar with actual target review; (7) **MPCN** (Tay et al. 2018), multi-pointer co-attention. It operates with a gumbel-softmax based pointer mechanism to select the most important reviews; (8) **NARRE** (Chen et al. 2018c), neural attentional rating regression with review-level explanations; (9) **HARR-help**, a variant of our approach without incorporating review helpfulness information. In this implementation, we pertain

Table 3 RMSE scores of different methods on different datasets

	Movies	Electronics	CDs	Toys
PMF Mnih and Salakhutdinov (2008)	1.3000	1.4007	1.1696	1.3076
NMF Lee and Seung (2001)	1.2989	1.3544	1.2253	1.0399
HFT McAuley and Leskovec (2013)	1.2578	1.3141	1.0379	1.1688
DeepCoNN Zheng et al. (2017)	1.1435	1.1713	1.0223	1.0281
D-ATT Seo et al. (2017a)	1.0895	1.1696	1.0001	0.9910
TransNets Catherine and Cohen (2017)	1.0817	1.1683	1.0051	0.9869
MPCN Tay et al. (2018)	1.0696	1.1619	1.0025	0.9864
NARRE Chen et al. (2018c)	0.9877	1.0853	0.9308	0.8769
HARR-help	0.9805	1.0480	0.9183	0.8699
HARR	0.9735	0.9671	0.8915	0.8517

Lower RMSE score means better performance

the hierarchical CNN architecture with attention to form user and item representations from reviews.

In practice, we train all MF based models till convergence. For interaction only models, the embedding size is set to 50. For text involving ones, we use the same Google embedding as our method. Also, CNN filters and window sizes of all the convolutional structure are set to be the same with our method. For D-ATT, the global attention layer uses filter sizes of [2, 3, 4]. We use two transform layers in the TransNets model. In MPCN model, the number of pointers p is fixed as three. Since some baseline methods don't model relative review importance, we assign a special delimiter token to separate reviews within a user/item document for DeepCoNN, TransNet and D-ATT. If FM is used, the number of factors is set to 10.

Table 3 reports the results of our experiments. Comparing our HARR method with other baselines, several observations can be made.

First, neural methods beat traditional matrix factorization (MF) methods significantly. Several reasons may account for this. To begin with, non-neural methods like MF or latent dirichlet allocation (LDA) only capture statistical distribution, even with reviews as input (HFT). They extract features rather than semantic meanings. On the contrary, neural based methods mostly use CNN as encoder, which have been proved to be effective in extracting local text meaning (Conneau et al. 2016; Le et al. 2018).

Second, methods using attention mechanism generally perform better than other neural methods. This is intuitive since online reviews are noisy and repetitive, using attention can help select the most informative information for recommendation. However, different implementation of attention matters in model performance. D-ATT uses both local and global attention to form review representations from words. MPCN improves performance by utilizing multi-pointer attention to model relevance between different reviews. For NARRE, it enables an automatic choice of review importance, thus performing better.

Third, compared to other neural models which also employ attention to select important information, our HARR-help approach achieves the best results. Among other neural baselines with attention, some methods concatenate all sentences in a review (e.g., D-ATT, MPCN, NARRE), ignoring relative importance between review sentences. Meanwhile, some others merge all reviews into a long document to form user or item representations (e.g., D-ATT, MPCN), which also neglect that some reviews are redundant or even misleading. Different with them, our approach learns representations from different levels of reviews in a hierarchical manner. Inside each level, we use attention to aggregate useful information and form representations for the upper level. Thus, our approach learns accurate review representations from word-level, sentence-level and review-level.

Fourth, our HARR approach performs better than HARR-help and consistently reaches the best result on all datasets. By using the helpfulness scores of reviews as query attention, we explicitly introduce helpfulness information from online user evaluation. Therefore, HARR can select helpful reviews in a user-guided way and provide better recommendation.

4.3 Analysis on helpfulness query attention

From Fig. 6, it can be discovered that using helpfulness scores as query attention vectors to select the most helpful review indeed boosts model performances. This is because using helpfulness scores as query vectors enables a dynamic user or item representations. In some cases, a review may mention the price or the quality of a product, but the description is general and ambiguous. Take the sentence "the quality of this earphone did not meet my expectation" as an example, it fails to introduce much information into our model. In our approach, the weight of a review is assigned according to its quality, thus preventing our model from being misled by less informative reviews. The improvement

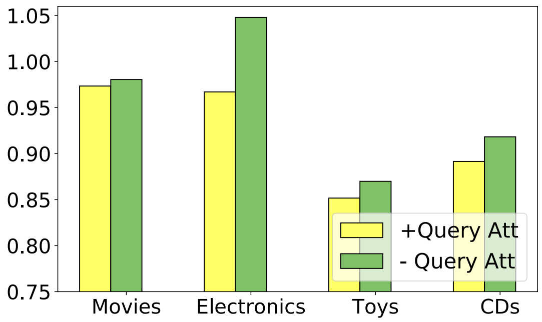


Fig. 6 The effectiveness of utilizing review helpfulness scores as attention queries

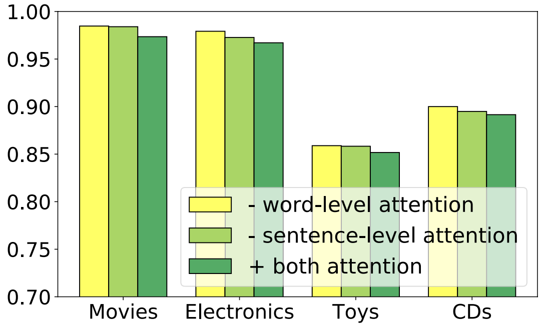
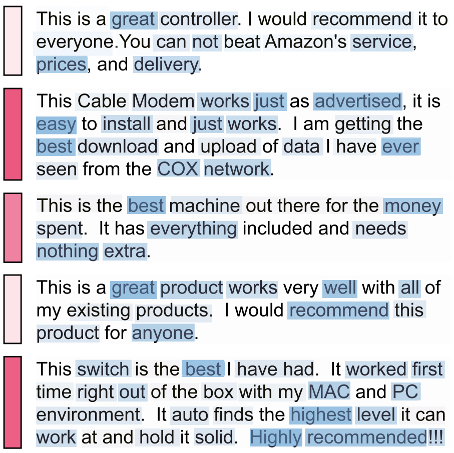


Fig. 7 Effectiveness of word- and sentence-level attention

on Electronics is the largest, which is in line with the fact that this dataset is more noisy than others. On Movies, the effectiveness is not much salient, since reviews of movies are subjective and vary between people, modeling their helpfulness is more challenged.

4.4 Analysis on hierarchical attention

To ascertain that each form of attention mechanism in our network works, we examine the relative contributions they make by removing certain layer each time and reporting performances of the remaining model. Figure 7 showcases the results when we remove word-level or sentence-level attention. According to Fig. 7, the word-level attention is effective in improving model performance. Since different words have different importance in user/item representation, highlighting keywords in a sentence for recommendation is an essential way to boost performance. For example, Keywords and short expressions mentioning price, quality and appearance can be automatically given higher weights in the model. From Fig. 7, we can also observe that the sentence-level attention is helpful in selecting important sentence when constructing review representations. In a review, a user may not directly point out the advantages or disadvantages of a product in each sentence, so modeling sentence importance is fundamental. What's more, though both attentions increase rating accuracy, conclusion can be reached that word-level attention is more essential than sentence-level attention.


This is a great controller. I would recommend it to everyone.You can not beat Amazon's service, prices, and delivery.

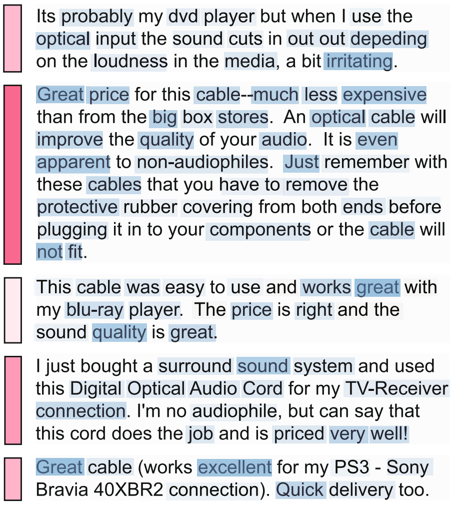
This Cable Modem works just as advertised, it is easy to install and just works. I am getting the best download and upload of data I have ever seen from the COX network.

This is the best machine out there for the money spent. It has everything included and needs nothing extra.

This is a great product works very well with all of my existing products. I would recommend this product for anyone.

This switch is the best I have had. It worked first time right out of the box with my MAC and PC environment. It auto finds the highest level it can work at and hold it solid. Highly recommended!!!

Fig. 8 Visualization of the review- and word-level attention weights in a user example. Red bars and blue blocks represent review- and word- level attention weights accordingly. Darker colors represent higher attention weights (color figure online)


Its probably my dvd player but when I use the optical input the sound cuts in out out depeding on the loudness in the media, a bit irritating.

Great price for this cable--much less expensive than from the big box stores. An optical cable will improve the quality of your audio. It is even apparent to non-audiophiles. Just remember with these cables that you have to remove the protective rubber covering from both ends before plugging it in to your components or the cable will not fit.

This cable was easy to use and works great with my blu-ray player. The price is right and the sound quality is great.

I just bought a surround sound system and used this Digital Optical Audio Cord for my TV-Receiver connection. I'm no audiophile, but can say that this cord does the job and is priced very well!

Great cable (works excellent for my PS3 - Sony Bravia 40XBR2 connection). Quick delivery too.

Fig. 9 Visualization of the review- and word-level attention weights in an item example

4.5 Case study

In this section, we will conduct several case studies to visually explore the effectiveness of the personalized attention mechanism in our approach. We randomly select a sample user and a sample item and show their attention visualization results in Figs. 8 and 9. The helpfulness of each review evaluated by the helpfulness predictor was illustrated in the left side of the review text. In each review text area, we showcase the word-level attention weights learned by the review encoder. From these two figures, we can see that our approach can effectively select and attend to informative reviews. For example, the second review in Fig. 8 is assigned high attention weight by our approach since it reveals rich information of user preferences. However, the third review in Fig. 9 receives low attention weight since it contains limited information of items. Moreover, in each sentence, it can be observed that words indicating particular user preferences and item features are assigned with more weights. Thus, these results validate the effectiveness of our approach in selecting informative words and reviews to learn more accurate user/item representations.

5 Conclusion

Reviews written by customers reflect actual functionality of products in a simplified and user-oriented way, but suffer heavily on subjectivity and redundancy. In this paper, we propose a helpfulness-aware review based approach for product recommendation. Our model learns each review using a review encoder and forms user/item representations based on these reviews with a user/item encoder. Different forms of attention mechanism strengthen the representation ability of our model. In the review encoder, we apply a hierarchical attention network to selectively encode important words and sentences. The review representations are the outputs of the review encoder. In user/item encoder, we form each user/item representation using an attention network with query vector. Each query vector is the helpfulness score of the related review, which indicates whether a review is helpful for describing the product's quality or not. Taking into account that considerable reviews don't have helpfulness scores associated with them, we propose a helpfulness prediction model to address the problem. Extensive studies further confirm the effectiveness of our approach in accurate recommendation and capturing helpful reviews.

References

Almahairi, A., Kastner, K., Cho, K., Courville, A.: Learning distributed representations from reviews for collaborative filtering. In: RecSys, pp. 147–154 (2015)

Bao, Y., Fang, H., Zhang, J.: Topicmf: Simultaneously exploiting ratings and reviews for recommendation. In: AAAI (2014)

Bell, R., Koren, Y., Volinsky, C.: Modeling relationships at multiple scales to improve accuracy of large recommender systems. In: KDD, pp. 95–104 (2007)

Bhatt, A., Patel, A., Chheda, H., Gawande, K.: Amazon review classification and sentiment analysis. Int. J. Comput. Sci. Inf. Technol. **6**(6), 5107–5110 (2015)

Catherine, R., Cohen, W.: Transnets: Learning to transform for recommendation. In: RecSys, pp. 288–296 (2017)

Chen, C., Qiu, M., Yang, Y., Zhou, J., Huang, J., Li, X., Bao, F.: Review helpfulness prediction with embedding-gated cnn. arXiv preprint arXiv:1808.09896 (2018)

Chen, C., Yang, Y., Zhou, J., Li, X., Bao, F.S.: Cross-domain review helpfulness prediction based on convolutional neural networks with auxiliary domain discriminators. In: NAACL, pp. 602–607 (2018)

Chen, C., Zhang, M., Liu, Y., Ma, S.: Neural attentional rating regression with review-level explanations. In: WWW, pp. 1583–1592 (2018)

Cheng, P., Wang, S., Ma, J., Sun, J., Xiong, H.: Learning to recommend accurate and diverse items. In: WWW, pp. 183–192 (2017)

Conneau, A., Schwenk, H., Barrault, L., Lecun, Y.: Very deep convolutional networks for text classification. arXiv preprint arXiv:1606.01781 (2016)

Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: AISTATS, pp. 315–323 (2011)

He, X., Chen, T., Kan, M.Y., Chen, X.: Trirank: Review-aware explainable recommendation by modeling aspects. In: CIKM, pp. 1661–1670 (2015)

Kim, D., Park, C., Oh, J., Lee, S., Yu, H.: Convolutional matrix factorization for document context-aware recommendation. In: RecSys, pp. 233–240 (2016)

Kim, S.M., Pantel, P., Chklovski, T., Pennacchiotti, M.: Automatically assessing review helpfulness. In: EMNLP, pp. 423–430 (2006)

Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)

Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: KDD, pp. 426–434 (2008)

Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **8**, 30–37 (2009)

Krishnamoorthy, S.: Linguistic features for review helpfulness prediction. Expert Syst. Appl. **42**(7), 3751–3759 (2015)

Le, H.T., Cerisara, C., Denis, A.: Do convolutional networks need to be deep for text classification? In: AAAI (2018)

Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. In: NIPS, pp. 556–562 (2001)

Ling, G., Lyu, M.R., King, I.: Ratings meet reviews, a combined approach to recommend. In: RecSys, pp. 105–112 (2014)

Liu, P., Qiu, X., Huang, X.: Adversarial multi-task learning for text classification. arXiv preprint arXiv:1704.05742 (2017)

Liu, Y., Huang, X., An, A., Yu, X.: Modeling and predicting the helpfulness of online reviews. In: ICDM, pp. 443–452 (2008)

Luo, X., Zhou, M., Li, S., You, Z., Xia, Y., Zhu, Q.: A nonnegative latent factor model for large-scale sparse matrices in recommender systems via alternating direction method. TNNLS **27**(3), 579–592 (2015)

McAuley, J., Leskovec, J.: Hidden factors and hidden topics: understanding rating dimensions with review text. In: RecSys, pp. 165–172 (2013)

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS, pp. 3111–3119 (2013)

Mnih, A., Salakhutdinov, R.R.: Probabilistic matrix factorization. In: NIPS, pp. 1257–1264 (2008)

Mudambi, S.M., Schuff, D.: What makes a helpful review? A study of customer reviews on amazon. com. MIS Q. **34**(1), 185–200 (2010)

Ngo-Ye, T.L., Sinha, A.P.: The influence of reviewer engagement characteristics on online review helpfulness: a text regression model. Decis. Support Syst. **61**, 47–58 (2014)

Park, S.T., Chu, W.: Pairwise preference regression for cold-start recommendation. In: Proceedings of the third ACM conference on recommender systems, pp. 21–28. ACM (2009)

Qazi, A., Syed, K.B.S., Raj, R.G., Cambria, E., Tahir, M., Alghazzawi, D.: A concept-level approach to the analysis of online review helpfulness. Comput. Hum. Behav. **58**, 75–81 (2016)

Seo, S., Huang, J., Yang, H., Liu, Y.: Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In: RecSys, pp. 297–305 (2017)

Seo, S., Huang, J., Yang, H., Liu, Y.: Representation learning of users and items for review rating prediction using attention-based convolutional neural network. In: MLRec (2017)

Singh, J.P., Irani, S., Rana, N.P., Dwivedi, Y.K., Saumya, S., Roy, P.K.: Predicting the "helpfulness" of online consumer reviews. J. Bus. Res. **70**, 346–355 (2017)

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)

Tan, Y., Zhang, M., Liu, Y., Ma, S.: Rating-boosted latent topics: Understanding users and items with ratings and reviews. IJCAI **16**, 2640–2646 (2016)

Tang, J., Gao, H., Hu, X., Liu, H.: Context-aware review helpfulness rating prediction. In: RecSys, pp. 1–8 (2013)

Tay, Y., Luu, A.T., Hui, S.C.: Multi-pointer co-attention networks for recommendation. In: KDD, pp. 2309–2318 (2018)

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: Hierarchical attention networks for document classification. In: ACL, pp. 1480–1489 (2016)

Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification. In: Advances in neural information processing systems, pp. 649–657 (2015)

Zheng, L., Noroozi, V., Yu, P.S.: Joint deep modeling of users and items using reviews for recommendation. In: WSDM, pp. 425–434 (2017)

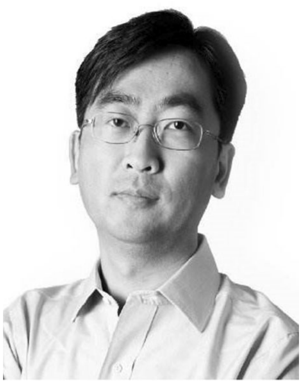
Zhou, K., Yang, S.H., Zha, H.: Functional matrix factorizations for cold-start recommendation. In: SIGIR, pp. 315–324 (2011)

Suyu Ge is a junior in the department of Electronic Engineering from Tsinghua University, Beijing, China. Her current research interests include text mining, and recommender systems.

Tao Qi is a senior in the department of Electronic Engineering from Tsinghua University, Beijing, China. His current research interests include text mining, information extraction and recommender systems.

Chuhan Wu is a doctoral student in the department of Electronic Engineering from Tsinghua University, Beijing, China. His current research interests include sentiment analysis, text mining and information extraction.

Fangzhao Wu is an associate researcher at Microsoft Research Asia. He received the B.E. degree in Electronic Engineering from Tsinghua University in 2012. He received the PhD degree in the Department of Electronic Engineering from Tsinghua University in 2017. His research interests include machine learning, text mining and social network analysis.



Dr. Xing Xie is currently a senior principal research manager at Microsoft Research Asia, and a guest Ph.D. advisor at the University of Science and Technology of China. He received his B.S. and Ph.D. degrees in Computer Science from the University of Science and Technology of China in 1996 and 2001, respectively. He joined Microsoft Research Asia in July 2001, working on data mining, social computing and ubiquitous computing.

Yongfeng Huang is a Professor in the Department of Electronic Engineering, Tsinghua University, China. He received the PhD degree in computer science and engineering from Huazhong University of Science and Technology in 2000. His research interests include next-generation Internet and Web data mining.