Computer Science and Information Systems 17(3):849–865 https://doi.org/10.2298/CSIS190925024B

A Recommendations Model with Multiaspect Awareness and Hierarchical User-Product Attention Mechanisms

Zhongqin Bi[1], Shuming Dou[2], Zhe Liu[3], and Yongbin Li[4]

[1] College of Computer Science and Technology, Shanghai University of Electric Power, Shanghai, China
zqbi@shiep.edu.cn
[2] College of Computer Science and Technology, Shanghai University of Electric Power, Shanghai, China
d2313776747@163.com
[3] State Grid Shanghai Electric Power Research Institute, Shanghai, China
liuzheacyy@163.com
[4] College of Computer Science and Technology, Shanghai University of Electric Power, Shanghai, China
lybin40000@163.com

Abstract. Neural network methods have been trained to satisfactorily learn user/product representations from textual reviews. A representation can be considered as a multiaspect attention weight vector. However, in several existing methods, it is assumed that the user representation remains unchanged even when the user interacts with products having diverse characteristics, which leads to inaccurate recommendations. To overcome this limitation, this paper proposes a novel model to capture the varying attention of a user for different products by using a multilayer attention framework. First, two individual hierarchical attention networks are used to encode the users and products to learn the user preferences and product characteristics from review texts. Then, we design an attention network to reflect the adaptive change in the user preferences for each aspect of the targeted product in terms of the rating and review. The results of experiments performed on three public datasets demonstrate that the proposed model notably outperforms the other state-of-the-art baselines, thereby validating the effectiveness of the proposed approach.

Keywords: recommendation, long short-term memory, attention mechanism, deep learning, rating prediction.

1. Introduction

The explosive growth in the amount of available digital information and the increasing number of Internet users has created a potential challenge pertaining to information overload, which hinders users from gaining timely access to their products of interest on the Internet. To alleviate the inconvenience caused by such information overload, recommendation systems are often employed. Recommendation systems do not require users to provide explicit information; instead, the users' interests are modeled by analyzing the users' historical behavior. Thus, recommendation systems can proactively recommend information that meets the users' interests and needs. In this manner, recommendation systems not only relieve the pressure of customer information overload but also help the platform optimize services and generate greater commercial benefits.

Recommendation algorithms have been widely integrated into many business application systems, such as the Netflix online video recommendation system or the Amazon online shopping mall. In the early stages of development, recommendation systems were based on demography [26], which was effective at providing real-time responses, although its reliability and interpretability were low. As data regarding users accumulates, it is more desirable to provide recommendations based on the users' interaction records. Recommender systems are generally based on two mechanisms: collaborative filtering (CF) [28] and content-based recommendation [4]. These methods can be used to realize enhanced recommendation performance and interpretability; however, both approaches have limitations due to data sparsity and cold start. To overcome the limitations associated with using any one specific recommendation method, hybrid recommendation systems have been developed [24, 32, 36] that combine several recommendation technologies to obtain a better recommendation effect.

In this paper, we propose a multiaspect awareness model with a hierarchical user-product attention (AHUPA) framework to provide recommendations. The proposed approach can adaptively learn both user preferences and product characteristics and can accurately capture the varying attention that a user gives to each aspect of different products. Compared with traditional collaborative filtering recommendation algorithms, this model possesses several unique advantages. First, a review encoder and a user/product encoder are constructed to learn better review representations and user/product representations from words and reviews, respectively. The attention network can capture the key words and reviews, making the extracted representations more reliable. Second, considering the significance of the overall ratings, we combine the user-product attention vectors learned from the encoders by using the user and product embeddings based on the IDs as the final representations for deep learning. Finally, to determine the attentive interaction of the users' and products' final latent factors corresponding to various aspects, we design a rating encoder with adaptive multiaspect awareness for each user-product pair. To sum up, the main contributions of this work are as follows.

- We propose a multiaspect awareness model for recommendations. With hierarchical user-product attention mechanisms, respectively, two representations are generated, which are concatenated for further representation learning.
- We introduce an attention network to capture the varying attention vectors of each specific user-product pair.
- We conduct comprehensive experiments on three public datasets to comparatively evaluate and demonstrate the effectiveness of the proposed approach.

The remainder of this paper is organized as follows. Section 2 describes the related works, Section 3 presents the preliminary definitions, and Section 4 describes the AHUPA model in detail. Experiments involving the training of the AHUPA model are presented and the advantages of the AHUPA model compared to other state-of-the-art methods are discussed in Section 5. Section 6 presents conclusions and describes future research directions.

2. Related Work

Learning accurate user and product representations is highly important for recommender systems. Many existing recommendation methods learn user and product representations based on the ratings that users give to products. Matrix factorization (MF) [11, 19, 27, 29] has become the most popular collaborative filtering approach[28]. Depending on the user-product rating matrix, the user interest and product characteristics are expressed as potential factor vectors in the public potential space, while the interaction rating between users and products is expressed in terms of the inner product. However, a rating only indicates a user's overall preference for a product, and the interpretability of the indication is low. For example, a user might assign a high score to a slim computer, but this information may not be supported by the overall rating. Consequently, MF methods cannot achieve fine-grained modeling of user preferences on the various product aspects, leading to unexplainable recommendations. Moreover, because most of the entries are missing, a rating matrix is usually extremely sparse, which leads to cold start problems and low recommendation accuracy [9, 10, 34].

To address these issues, several researchers [5, 12, 23] have combined auxiliary information with ratings to improve the recommendation performance. As supplementary information for ratings, reviews contain rich semantics that can be mined to identify user preferences and product characteristics [39]. Different approaches have been developed to utilize textual reviews for recommendations. such as HFT[23], TriRank [21], EFM [40], and LDA [3]. These topic-based methods integrate topic models in their frameworks to automatically identify aspect information in reviews for better user and product modeling. Specifically, EFM, TriRank and LDA have explicitly claimed that they can provide explanations for recommendations. Although those methods have achieved better performances than have MF methods using only ratings, the features are simple extractions of words or phrases from the texts, which changes the integrity of the reviews and may distort their original meanings. Therefore, these shallow feature extraction methods cannot accurately model user and product representations.

In recent years, with the help of the powerful feature extraction ability of deep learning models, some studies have also tried to combine different neural network structures with collaborative filtering to learn feature interactions. In [15], He et al. presented a neural collaborative filtering (NCF) framework to learn the nonlinear interactions between users and items. Later, researchers attempted to incorporate critical information into the NCF model to improve the recommendation performance. Zheng et al. [41] proposed a deep cooperative neural network (DeepCoNN) that uses convolutional neural networks to learn the user and product representations from reviews; this model led to significantly improved recommendations. Chen et al. [6] proposed a neural attentional regression model with review-level explanations (NARRE), which took into account the effectiveness of the comments while predicting the score and explains the recommendation results at the level of comment information. Zhang et al. [38] developed a neural user-item coupling model (CoupledCF) to realize joint learning in terms of explicit and implicit coupling and used the review information to enhance the user-product coupling relationships. These methods first train unsupervised neural networks to obtain word embedded representations and then integrate deep neural networks with rating information to achieve good recommendation performance. However, they overlook the fact that the user's attention preferences regarding product aspects vary for different products, which may result in weak recommendations. User preferences can be understood as multiple aspects of the attention vector, which changes according to different products. Consequently, users have different preferences for the functions of different products of the same category; for example, in the computer category, in addition to focusing on basic features, a user will expect an expensive computer to have higher resolution and rendering capabilities than a cheap computer.

Attention mechanisms have achieved immense success in speech recognition, computer vision, natural language processing and exploration of the hierarchical attention mechanism [1, 2, 22, 31, 35, 37]. The key idea of soft attention is to learn to assign attentive weights (normalized by summing to 1) for a set of features: higher (lower) weights indicate that the corresponding features are informative (less informative) for the end task, which learns and determines the focus areas and enables the model to focus on the most effective information with limited resources. In the field of recommendation systems, researchers proposed the deep knowledge aware network (DKN) [33] and introduced an attention network to learn user representations based on the users' news-click records. Chen et al. [6] modeled the usefulness of reviews by using review-level attention to enhance the learning of both the user and product representations. In this paper, we propose a multiaspect awareness attention model that learns the user preference weights over various product aspects and then guides the interaction between user preferences and product characteristics for different aspects.

3. Preliminaries

3.1. Long Short-term Memory

The long short-term memory (LSTM) [16] model is a sequential convolutional network derived from a recurrent neural network (RNN) that solves the gradient disappearance and gradient explosion problems endemic to RNNs by utilizing a cell state and a gate mechanism. LSTM is widely used in long-sequence text modeling tasks due to its excellent performance when learning and memorizing long sequential data.

Three gate structures are used to control the state flow in the LSTM cell. In each time step t, given an input sequence vector $\boldsymbol{x}_t$, the current cell state $\boldsymbol{c}_t$ and hidden state $\boldsymbol{h}_t$ can be updated with the previous cell state $\boldsymbol{c}_{t-1}$ and hidden state $\boldsymbol{h}_{t-1}$ as follows:

$$\begin{bmatrix} \boldsymbol{i}_t \\ \boldsymbol{f}_t \\ \boldsymbol{o}_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \end{bmatrix} \left(\boldsymbol{W} \left[\boldsymbol{h}_{t-1}; \boldsymbol{x}_t \right] + \boldsymbol{b} \right), \tag{1}$$

$$\hat{\boldsymbol{c}}_t = \tanh \left(\boldsymbol{W}_c \left[\boldsymbol{h}_{t-1}; \boldsymbol{x}_t \right] + \boldsymbol{b}_c \right), \tag{2}$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \hat{\boldsymbol{c}}_t, \tag{3}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot \tanh(\boldsymbol{c}_t), \tag{4}$$

where $\boldsymbol{i}_t$, $\boldsymbol{f}_t$ and $\boldsymbol{o}_t$ denote the input gate, forget gate and output gate in the gate mechanism, respectively; σ denotes the logistic sigmoid function; $\tanh$ is the hyperbolic tangent activation function; and $\odot$ represents elementwise multiplication. Intuitively, the input gate $\boldsymbol{i}_t$ determines the amount of information the current $\boldsymbol{x}_t$ reserves for $\boldsymbol{c}_t$; the forget gate $\boldsymbol{f}_t$ determines the amount of the previous cell state $\boldsymbol{c}_{t-1}$ saved to the current $\boldsymbol{c}_t$; and the output gate $\boldsymbol{o}_t$ determines the amount of $\boldsymbol{c}_t$ passed to the output $\boldsymbol{h}_t$ of the current state.

In practice, because the output at the current moment is related not only to the previous state but also to the future state, the prediction must be performed while considering the combination of the forward and backward inputs. A more common approach is to apply a bidirectional LSTM to simulate the text semantics from the forward and backward inputs. For the sequence vectors $[\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_T]$, the forward and backward LSTMs read the sequences from $\boldsymbol{x}_1$ to $\boldsymbol{x}_T$ and from $\boldsymbol{x}_T$ to $\boldsymbol{x}_1$, respectively. Subsequently, the forward hidden state $\overrightarrow{\boldsymbol{h}_t}$ and backward hidden state $\overleftarrow{\boldsymbol{h}_t}$ are connected to represent the current hidden layer state $\mathbf{h}_t$, i.e., $\boldsymbol{h}_t = \left[\overrightarrow{\boldsymbol{h}_t} ; \overleftarrow{\boldsymbol{h}_t}\right]$, which learns the whole sequence information corresponding to the target word w_{it}.

3.2. Attention Mechanism

The attention mechanism refers to human attention and modes of thought. Humans can rapidly screen valuable information from a large amount of content. The attention mechanism uses an attention calculation method to extract the weight of the key information, and it is widely used in various deep learning tasks, such as natural language processing, image classification and speech recognition. In particular, in the field of text processing, the attention mechanism can model the hierarchical semantics of words, sentences, and documents.

For the input information vectors $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n]$, the additive model is used to score the attention to select the important information in $\boldsymbol{X}$. The soft selection mechanism is adopted to capture the weight distribution of the input information as follows:

$$e_i = \boldsymbol{v}^\top \tanh\left(\boldsymbol{W}\boldsymbol{x}_i + \boldsymbol{b}_i\right), \tag{5}$$

$$\alpha_i = \frac{\exp\left(e_i\right)}{\sum_{j=1}^{N} \exp\left(e_j\right)}, \tag{6}$$

where the probability vector of α_i is termed the distribution of attention and $\boldsymbol{W}$ and $\boldsymbol{v}$ are learnable network parameters. The input information is aggregated by weighted averaging as follows:

$$\boldsymbol{s} = \sum_{i=1}^{N} \alpha_i \boldsymbol{x}_i. \tag{7}$$

4. Methods

In this section, we describe the proposed AHUPA model in detail. The framework of AHUPA is illustrated in Figure 1. The proposed model involves three major modules: a review encoder, which learns the representations of the reviews from words; the user/product encoder, which learns the representations of the users and products from reviews; and the rating encoder, which captures the user's attention weight regarding various product aspects. Next, we introduce each module in detail.

4.1. Review Encoder

Given a review with the words $[w_{i1,} w_{i2,} ..., w_{it}]$, $t \in [1, T]$, we first embed the reviews to the vector sequence $[\boldsymbol{e}_{i1,} \boldsymbol{e}_{i2,} ..., \boldsymbol{e}_{it}]$ through an embedding matrix $\boldsymbol{E} \in R^{\mathrm{V} \times \mathrm{D}}$, where

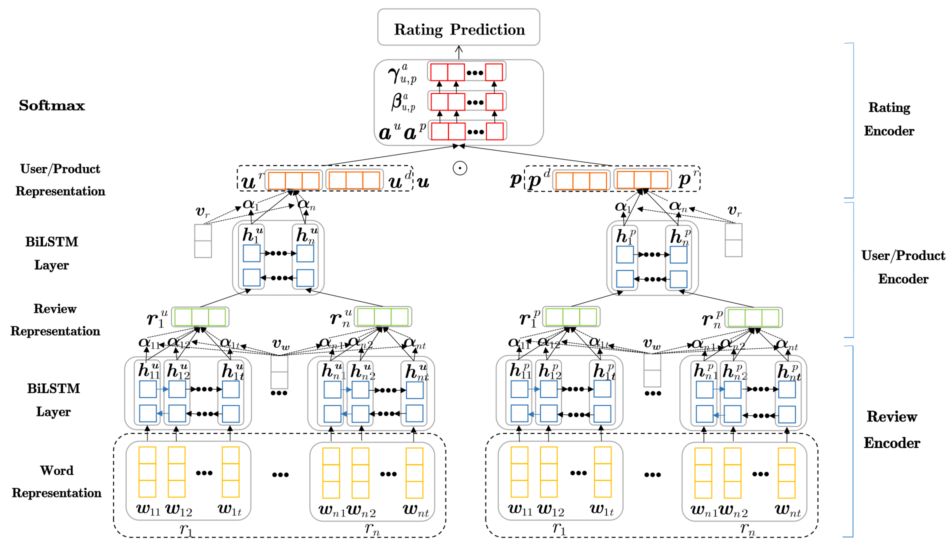


Fig. 1. Architecture of the proposed AHUPA model

V and D represent the vocabulary size and word embedding dimension, respectively. We obtain the word embedding $\boldsymbol{E}$ by training an unsupervised word2vec [27] model fine-tuned during training. We use a bidirectional LSTM to extract the contextual information in both directions for the words; specifically, this LSTM contains both a forward and backward LSTM that read a review in forward and reverse order, respectively:

$$\overrightarrow{\boldsymbol{h}_{it}} = \overrightarrow{\text{LSTM}}\left(\boldsymbol{e}_{it}\right), t \in [1, T], \tag{8}$$

$$\overleftarrow{\boldsymbol{h}_{it}} = \overleftarrow{\text{LSTM}}\left(\boldsymbol{e}_{it}\right), t \in [T, 1]. \tag{9}$$

We obtain the word output representation $\boldsymbol{h}_{it}$ by concatenating the forward hidden state $\overrightarrow{\boldsymbol{h}_{it}}$ and the backward hidden state $\overleftarrow{\boldsymbol{h}_{it}}$, i.e., $\boldsymbol{h}_{it} = \left[\overrightarrow{\boldsymbol{h}_{it}}; \overleftarrow{\boldsymbol{h}_{it}}\right]$, which learns the review information around the target word w_{it}.

From the product perspective, not all words equally reflect the product's characteristics. Hence, we introduce a word-level attention mechanism to help the models select the important words based on the contextual information and build more informative word representations. The attention weight of the t-th word in the i-th review is computed as follows:

$$u_{it} = \boldsymbol{v}_w^T \tanh\left(\boldsymbol{W}_w \boldsymbol{h}_{it} + \boldsymbol{b}_w\right), \tag{10}$$

$$\alpha_{it} = \frac{\exp\left(u_{it}\right)}{\sum_{t=1}^{T} \exp\left(u_{it}\right)}. \tag{11}$$

where $\boldsymbol{v}_w \in R^{N_w}$ and $\boldsymbol{b}_w \in R$ denote the attention network parameters, and α_{it} represents the relative importance of the t-th word in the attention network evaluation. The final representation of the i-th review is a weighted sum of the word level representations in a view as follows:

$$\boldsymbol{r}_i = \sum_{t=1}^{T} \alpha_{it} \boldsymbol{h}_{it}. \tag{12}$$

4.2. User/Product Encoder

We build the representations of the users or products based on the representations of their reviews in a similar manner. We first use a bidirectional LSTM to encode the reviews:

$$\overrightarrow{\boldsymbol{h}_i} = \overrightarrow{\text{LSTM}}\left(\boldsymbol{r}_i\right), i \in [1, L], \tag{13}$$

$$\overleftarrow{\boldsymbol{h}_i} = \overleftarrow{\text{LSTM}}\left(\boldsymbol{r}_i\right), i \in [L, 1], \tag{14}$$

where L is the number of reviews of a user or product. We concatenate $\overrightarrow{\boldsymbol{h}_i}$ and $\overleftarrow{\boldsymbol{h}_i}$ to obtain an output representation of the i-th review, i.e., $\boldsymbol{h}_i = \left[\overrightarrow{\boldsymbol{h}_i}\,;\,\overleftarrow{\boldsymbol{h}_i}\right]$. $\boldsymbol{h}_i$ summarizes the neighbor reviews around the i-th review.

Different reviews contribute differently to the modeling of a user or product. Therefore, at the review level, we use an attention framework to help the proposed model distinguish informative reviews from less informative reviews, thereby generating representations of users or products that are better at obtaining user preferences and product characteristics. The attention weight of the i-th review is computed as follows:

$$u_i = \boldsymbol{v}_r^T \tanh\left(\boldsymbol{W}_r \boldsymbol{h}_i + \boldsymbol{b}_r\right), \tag{15}$$

$$\alpha_i = \frac{\exp\left(u_i\right)}{\sum_{i=1}^{L} \exp\left(u_i\right)}, \tag{16}$$

where $\boldsymbol{v}_r \in R^{N_s}$ and $\boldsymbol{b}_r \in R$ are the attention network parameters, and $\boldsymbol{W}_r$ represents the weight matrices. The final representation of the user or product is the summation of the contextual representations of the reviews weighted by their attention weights:

$$\boldsymbol{u} = \sum_{i=1}^{L} \alpha_i \boldsymbol{h}_i. \tag{17}$$

4.3. Rating Encoder

In this section, we describe the attention network used to determine the user's attention weight for different aspects of the product.

Although a large amount of valuable information can be learned from reviews, several of the underlying characteristics of the users and products can be inferred from the ratings instead of from the reviews. Therefore, the module input is divided into two parts. In one part, the review encoder and the user/product encoder obtain the user preferences or product characteristics from the reviews. The other part converts the user and product IDs into a dense vector through the embedding layer. Next, the ID-based embedded features and review-based features of the users and products are used as the input to the next layer.

Feature fusion can be conducted to improve model learning. In previous works, different fusion methods have had different effects on model performances. In this work, we select the connection method to perform feature fusion. The final user representation is a combination of the user preference $\boldsymbol{u}^r$ and user density vector $\boldsymbol{u}^d$, as follows:

$$\boldsymbol{u} = \left[\boldsymbol{u}^r; \boldsymbol{u}^d\right]. \tag{18}$$

Similarly, the final product representation is a combination of the product preference and product density vector:

$$\boldsymbol{p} = \left[\boldsymbol{p}^r; \boldsymbol{p}^d\right]. \tag{19}$$

Moreover, to further improve the model's learning ability, we add a fully connected layer after the fusion step and adopt the nonlinear ReLU activation function [13].

The core of the rating encoder is to determine the specific attention weight of user u regarding one aspect k of product p, $k \in [0, K]$, where K is the dimension of the latent vectors. Although the user preferences and product characteristics can be clearly reflected by the reviews, certain latent characteristics can be obtained by observing the rating patterns. Thus, we rely on reviews and ratings to capture users' preferences for various aspects of a product. Subsequently, the attention weight based on reviews is computed as follows:

$$\boldsymbol{r} = \left[\boldsymbol{u}^r; \boldsymbol{p}^r\right], \tag{20}$$

$$\boldsymbol{\gamma}_{u,p} = \boldsymbol{v}^T \mathrm{Re}LU\left(\boldsymbol{W}_a \boldsymbol{r} + \boldsymbol{b}_a\right), \tag{21}$$

$$\gamma_{u,p,k}^a = \frac{\exp\left(\gamma_{u,p,k}\right)}{\sum_{j=1}^{K} \exp\left(\gamma_{u,p,k}\right)}. \tag{22}$$

where $\boldsymbol{W}_a$ and $\boldsymbol{b}_a$ denote the weight matrix and bias vector, respectively. The attention weight $\beta_{u,p,k}^a$ based on ratings is computed in a similar manner.

We set $\boldsymbol{a}^u$ and $\boldsymbol{a}^p$ as the feature vectors of the user and product through the feature fusion, respectively. In addition, two attention factors are introduced to the interaction between the user and product that allow the user to adjust the degree of attention to different products according to the attention weight distributions $\boldsymbol{\gamma}_{u,p}^a$ and $\boldsymbol{\beta}_{u,p}^a$. The output of the attention interaction fusion can then be used as a rating feature as follows:

$$\boldsymbol{z}_{u,p}^a = \boldsymbol{\gamma}_{u,p}^a \odot \boldsymbol{\beta}_{u,p}^a, \tag{23}$$

$$\boldsymbol{F} = \boldsymbol{z}_{u,p}^a \odot \left(\boldsymbol{a}^u \odot \boldsymbol{a}^p\right), \tag{24}$$

where $\boldsymbol{z}_{u,p}^a \in R^K$ denotes a user's attention vector to a product, and $\odot$ is the elementwise product. We assume that the user's attention is distributed in k dimensions. According to the relation $f_k = z_{u,p,k}^a \cdot a_k^u \cdot a_k^p$, an attentional weight $z_{u,p,k}^a$ adjusts the interaction depth of the user-product pair under a factor k, which indicates that $z_{u,p,k}^a$ reflects the importance of factor k in the relationship between the user and the product.

In recommendation systems, the system recommends products based on the user's predicted rating of a product. In the proposed model, a linear layer and a softmax layer are used to project the score representation $\boldsymbol{F}$ into a score distribution of N classes:

$$q = \mathrm{softmax}\left(\boldsymbol{W}\boldsymbol{F} + \boldsymbol{b}\right). \tag{25}$$

Finally, we use cross entropy as the loss function to optimize the model parameters and minimize the difference between the actual and predicted rating:

$$L = -\sum_{d \in T} \sum_{n=1}^{N} q_n^g (\boldsymbol{F}) \cdot \log (q_n (\boldsymbol{F})) . \tag{26}$$

where q_n^g is the probability of the rating score n, with the ground truth being 1 and the other values being 0. T represents the training set.

5. Experiments and Evaluation

5.1. Experimental Settings

We evaluated the effectiveness of the proposed model on three publicly accessible subsets from the Amazon collection[5] [14], i.e., **Patio, Lawn and Garden**, **Video Games**, and **CDs and Vinyl**. The dataset statistics are listed in Table 1. Similar to the approach reported in [14], we preprocessed all the datasets by retaining only users and products with at least 5 reviews. Moreover, we removed the punctuation, stopwords, and infrequent symbols (those appearing less than 10 times) from each user/product review. The ratings range in these datasets is [1, 5].

Table 1. Statistics of the evaluation datasets

Datasets	#users	#products	#reviews
Patio, Lawn and Garden	1,686	962	13,272
Video Games	24,303	10,672	231,780
CDs and Vinyl	75,258	64,443	1,097,592

We pretrained a word2vec [25] model to initialize the word embedding matrix $\boldsymbol{E}$. We randomly initialized all the weight parameters to conform to the uniform distribution $U(-0.01{,}0.01)$. To speed up training, we limited each user and product to at most 10 reviews, where every review has no more than 60 words. We used the Adam optimizer [17] as the optimization algorithm and set the initial learning rate to 0.005; the dropout technique [30] was used to prevent overfitting, and the dropout ratio was 0.5. We randomly selected 80% of the user-product pairs in each dataset for training, 10% for validation, and 10% for testing. The optimal parameters were selected based on the model's performance on the validation set. Root mean square error (RMSE) was used to evaluate model performances. The RMSE is defined as follows:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(y^{\hat{(i)}} - y^{(i)} \right)^2}, \tag{27}$$

where $\boldsymbol{N}$ is the number of samples, $y^{\hat{(i)}}$ and $y^{(i)}$ are the predicted rating score and the gold rating score respectively of the i-th user-product pair.

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

5.2. Baselines

We compared the proposed AHUPA model with several baseline methods in terms of rating predictions. The methods used for comparison included the following:

- **PMF** [27]: Probabilistic matrix factorization that provide recommendations based on ratings via matrix factorization.
- **SVD++** [18]: A recommendation method based on a rating matrix via an SVD and similarities between items.
- **HFT** [23]: HFT models ratings and review texts using MF and LDA, respectively. Then, an exponential transform function is used to associate the latent topics and latent factors for rating prediction.
- **A^{3}NCF** [8]: A benchmark deep neural network that combines ratings with reviews to predict ratings. This model introduces a topic model to extract the user preferences and product characteristics from the review text.
- **CoupledCF** [38]: A coupled learning model that combines ratings and reviews to provide recommendations based on non-IID values. A doc2vec [20] framework is trained to obtain the user or product characteristics based on the reviews.
- **NARRE** [6]: A neural attention regression model with review level interpretation that uses attention mechanisms to simulate the informality of the recommended comments.

5.3. Model Comparisons

Table 2. Results obtained using different methods on different datasets. RMSE (lower values are better) is the evaluation metric

Models	Patio, Lawn and Garden	Video Games	CDs and Vinyl
PMF	1.1329	1.2052	1.1040
SVD++	0.9185	0.9543	0.8949
HFT	0.9073	0.9475	0.8920
A^{3}NCF	0.8855	0.9402	0.8871
CoupledCF	0.8864	0.9345	0.8827
NARRE	0.8725	0.9249	0.8792
AHUPA	0.8676	0.9181	0.8699

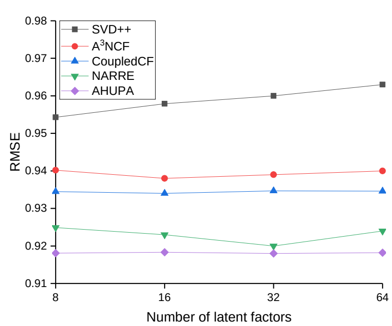
The rating prediction results of the different methods are presented in table 2. In addition, Figure 2 shows the performances of the partially adopted methods under different numbers of predictive factors. Considering space limitations, we show only the results on the two relatively larger datasets. For the MF methods (PMF and SVD++), the number of predictive factors is equal to the number of latent factors. Due to the weak performance of PMF, it is omitted in Figure 2 to better highlight the performance differences among the other methods. Based on these findings, the following observations can be made.

First, the methods that incorporate reviews (i.e., HFT, CoupledCF, NARRE, A^{3}NCF, AHUPA) exhibit a better performance than do the collaborative filtering models (PMF, SVD++) that consider only the rating matrix as the input. This is not surprising, because review information is complementary to ratings and contains rich information about user preferences and product characteristics. Therefore, better-quality modeling increases the accuracy when learning user preferences and product features and leads to a better rating prediction result.

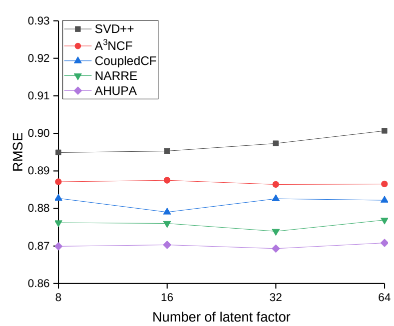
Second, compared to the HFT model, which utilizes traditional matrix factorization techniques, the methods that utilize deep learning technology (i.e., CoupledCF, NARRE, AHUPA) demonstrate superior performances among the methods that exploit reviews. This phenomenon likely occurs because deep learning can model users and products in a nonlinear [15] way. Previous works [37] have also demonstrated that methods based on neural networks capture semantic meaning better than topic models such as LDA [3] when analyzing textual information.

Third, as shown in Table 2, our method AHUPA achieves the best performance over all the datasets, significantly outperforming all the baseline methods. Although review information is useful in recommendation, the performance can vary depending on how the review information is utilized. Our model integrates features extracted from reviews via an LSTM and—more importantly—applies attention networks to learn user and product representations instead of relying on a simple concatenation. Moreover, we propose an attention for utilizing text and rating information to capture the user's attention regarding different aspects of the product. This attention design allows each user preference to be modeled at a finer granularity. As the results show, this approach can lead to a better performance.

Finally, as the number of latent factors increases, AHUPA achieves the best performances on both datasets in Figure 2. Compared to SVD++, review-based methods are relatively stable, because they can already achieve relatively good performances even when the number of latent factors is small. This result demonstrates the benefits of considering review information in preference modeling. In addition, AHUPA outperforms NARRE on all the latent factors, which shows the benefits of capturing the attention weights for each user-product pair in the rating encoder while considering review helpfulness.



(a) Video Games



(b) CDs and Vinyl

Fig. 2. Performance of all competitors w.r.t. the number of latent factors on two larger datasets.

5.4. Model Analysis: Effectiveness of Multilayer Attentive Learning

To examine the effectiveness of the multilayer attention in the proposed model, we compare three model variants by removing one type of attention each time to evaluate its contribution to the performance. The results are shown in Fig. 3.

As Fig. 3 shows, the rating attention mechanism effectively improves model performance, which validates our idea that users pay different attention to different products. This result also indicates that the proposed rating attention effectively captures the weight changes that represent users' attention to different products. Moreover, word-level attention is also useful. This may be because recognizing and highlighting the important words using the word-level attention network helps in learning more informative review representations. In addition, review-level attention is also useful in our AHUPA approach because different reviews are informative to different degrees for both the users and the product representations. Differentiating between informative and less informative reviews can help the model learn more accurate representations of users and items. Finally, combining all three levels of attention further improves the performance of the proposed approach, which validates the effectiveness of the attention mechanism in the proposed approach.

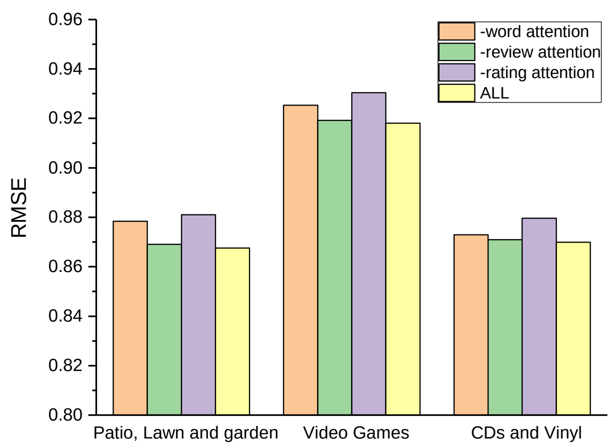


Fig. 3. Effectiveness of different attention layers

5.5. Model Analysis: Effectiveness of Attention Weights Based on Reviews and Ratings

To investigate the effects of the attention weight $\gamma^a_{u,p}$ and $\boldsymbol{\beta}^a_{u,p}$ for each user-product pair in the rating encoder, we implemented two separate network models by removing $\gamma^a_{u,p}$ or $\boldsymbol{\beta}^a_{u,p}$ from the attentive interaction. Fig. 4 shows the performance of a single network model with review-based ($\gamma^a_{u,p}$) or rating-based ($\boldsymbol{\beta}^a_{u,p}$) attention information.

The results show that the review information is more effective than the rating information in enhancing the recommendation performance because reviews provide more valuable information than do ratings with regard to learning the changes in the users' attention weights for each user-product pair—especially in larger datasets. Moreover, the proposed model exhibits the highest performance, which validates the rationality of calculating attention weights based on reviews and ratings. The results also indicate that the proposed rating encoder can capture the attention weights of users pertaining to different aspects of each product.

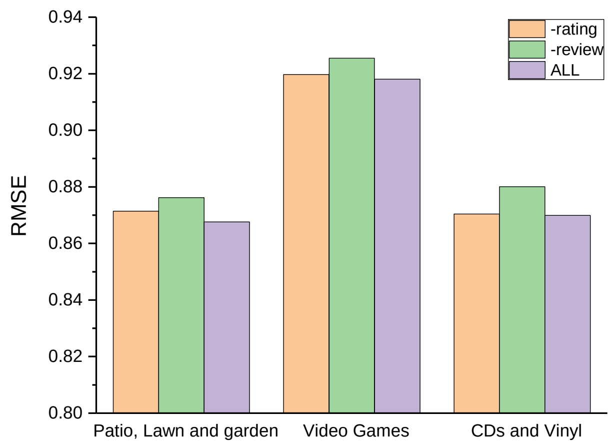


Fig. 4. Effectiveness of reviews and ratings

5.6. Visualization of Attention Weights

We conducted several case studies to verify the observations and demonstrate the effectiveness of the word- and review-level attention networks. First, we visualized the attention weights of the word-level attention networks, as shown in Fig. 5. According to Fig. 5, the word attention network effectively selects important words. For example, the words "high" and "quality" are assigned higher attention weights than are the words "hoses" and "clean". This is because "high" and "quality" express more information than the other review words in terms of the user and product representations.

Second, the model effectively selects the important reviews by using the review level attention network. For example, the first review shown in Fig. 6(a) is assigned a high attention weight because it is highly informative for modeling the user preferences, while the second review shown in Fig. 6(a) is assigned a low attention weight because it contains limited information about the users. In addition, as shown in Fig. 6(b), the second review is assigned a higher attention weight than the fourth review because it is more valuable for modeling the product characteristics. These results validate that the proposed approach effectively evaluates the varying importances of words and reviews to highlight user preferences and product characteristics.

This is a high quality 8 ply hose. I have had good luck with Gilmour hoses. A good choice in hoses.
This is difficult to clean.
Boyfriend got this for xmas and he loved it. We haven't used it yet but he looks forward to it.
Cleans easily and the hummers seem to like it.
The seed has a lot of filler. Not sure it's worth the money.

Fig. 5. Visualization of the word-level attention weights in randomly selected reviews from the Patio, Lawn and Garden dataset. A darker color corresponds to a higher attention weight

Highly recommended as a great PS2 back catalog game.
This one takes the small unit control scheme and adds some new elements. It's ok, but not as good as Dawn of War.
A great Xbox game and a must have for any Dungeons and Dragons fan that also likes videogames. It's best when played with others as the classes complement each other and the most fun is had by enjoying the experience with someone else. The cutscenes and dialogue are sometimes clunky, but otherwise a good time.
A great JRPG that's a lot different than the norm. It's a tough find at a good price, however.

(a) User

It works perfectly! Nothing is wrong with it.
Works great. Cheap price and way more memory than on the Nintendo make. Really great for you retro fans out there.
These things work really great and can be combined for even longer cable lengths but be warned that the rubber shielding seems to slip around the plug head. Not a big deal, but could lead to eventual short.
This thing was DOA. It wasn't even worth returning for the money I spent on it. Don't waste your money on this card.

(b) Product

Fig. 6. Visualization of review attention weights for a randomly selected user and product in the CDs and Vinyl dataset. A darker color corresponds to a higher attention weight

6. Conclusions

This paper proposes a novel multiaspect awareness model based on hierarchical user-product attention to realize rating prediction. We use a review encoder to learn the review representations from words and a user/product encoder to learn the user/product representations from the reviews. In addition, a rating encoder is introduced in AHUPA that utilizes the features extracted from reviews and ratings; this encoder can clarify the attention that a user pays to each aspect of the targeted product. Specifically, this attention design fully reflects the changes in the user's attention for the same aspect of different products. Experiments performed on three real-world datasets from Amazon demonstrate that the proposed approach effectively improves the recommended performance and outperforms many baseline methods.

In the future, we aim to more extensively examine user-product attention interactions to explore the different product aspects that interest users and develop improved embedded learning methods for extracting user and product characteristics by adopting better language models. Moreover, we aim to consider a larger amount of ancillary information, such as labels and attributes [7], that can better express the user interest and product characteristics.

Acknowledgments. This work is supported by the National Nature Science Foundation of China (No. 61972357, No. 61672337).

References

1. Ahmed, K., Keskar, N.S., Socher, R.: Weighted transformer network for machine translation. ArXiv abs/1711.02132 (2018)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. CoRR abs/1409.0473 (2014)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. 3, 993–1022 (2003)
4. Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, Á.: Recommender systems survey. Knowl. Based Syst. 46, 109–132 (2013)
5. Catherine, R., Cohen, W.W.: Transnets: Learning to transform for recommendation. CoRR abs/1704.02298 (2017), http://arxiv.org/abs/1704.02298
6. Chen, C., Zhang, M., Liu, Y., Ma, S.: Neural attentional rating regression with review-level explanations. In: WWW (2018)
7. Cheng, H.T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., Anderson, G., Corrado, G.S., Chai, W., Ispir, M., Anil, R., Haque, Z., Hong, L., Jain, V., Liu, X., Shah, H.: Wide deep learning for recommender systems. In: DLRS 2016 (2016)
8. Cheng, Z., Ding, Y., He, X., Zhu, L., Song, X., Kankanhalli, M.S.: $\mathbf{A^3NCF}$: An adaptive aspect attention model for rating prediction. In: IJCAI (2018)
9. Cheng, Z., Ding, Y., Zhu, L., Kankanhalli, M.S.: Aspect-aware latent factor model: Rating prediction with ratings and reviews. ArXiv abs/1802.07938 (2018)
10. Gao, H., Liu, C., Li, Y., Yang, X.: V2vr: Reliable hybrid-network-oriented v2v data transmission and routing considering rsus and connectivity probability. IEEE Transactions on Intelligent Transportation Systems pp. 1–14 (2020)
11. Gao, H., Xu, Y., Yin, Y., Zhang, W., Li, R., Wang, X.: Context-aware qos prediction with neural collaborative filtering for internet-of-things services. IEEE Internet of Things Journal 7(5), 4532–4542 (2020)

12. Gao, H., Kuang, L., Yin, Y., Guo, B., Dou, K.: Mining consuming behaviors with temporal evolution for personalized recommendation in mobile marketing apps. Mobile Networks and Applications pp. 1–16 (06 2020)
13. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: AISTATS (2011)
14. He, R., McAuley, J.J.: Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. ArXiv abs/1602.01585 (2016)
15. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. Proceedings of the 26th International Conference on World Wide Web (2017)
16. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation 9, 1735–1780 (1997)
17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR abs/1412.6980 (2014)
18. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: KDD (2008)
19. Koren, Y., Bell, R.M., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer 42 (2009)
20. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. ArXiv abs/1405.4053 (2014)
21. Ling, G., Lyu, M.R., King, I.: Ratings meet reviews, a combined approach to recommend. In: RecSys '14 (2014)
22. Ma, X., Gao, H., Xu, H., Bian, M.: An iot-based task scheduling optimization scheme considering the deadline and cost-aware scientific workflow for cloud computing. EURASIP Journal on Wireless Communications and Networking 2019 (12 2019)
23. McAuley, J.J., Leskovec, J.: Hidden factors and hidden topics: understanding rating dimensions with review text. In: RecSys '13 (2013)
24. Melville, P., Mooney, R.J., Nagarajan, R.: Content-boosted collaborative filtering for improved recommendations. In: AAAI/IAAI (2002)
25. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS (2013)
26. Pazzani, M.J.: A framework for collaborative, content-based and demographic filtering. ARTIFICIAL INTELLIGENCE REVIEW 13, 393–408 (1999)
27. Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. In: NIPS (2007)
28. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web. pp. 285–295. WWW01, Association for Computing Machinery, New York, NY, USA (2001), https://doi.org/10.1145/371920.372071
29. Seemann, P.: Matrix factorization techniques for recommender systems (2014)
30. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15, 1929–1958 (2014)
31. Sukhbaatar, S., Szlam, A., Weston, J., Fergus, R.: End-to-end memory networks. In: NIPS (2015)
32. Wang, C., Blei, D.M.: Collaborative topic modeling for recommending scientific articles. In: KDD (2011)
33. Wang, H., Zhang, F., Xie, X., Guo, M.: Dkn: Deep knowledge-aware network for news recommendation. In: WWW (2018)
34. Wang, X., He, X., Feng, F., Nie, L., Chua, T.S.: Tem: Tree-enhanced embedding model for explainable recommendation. In: WWW (2018)
35. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A.C., Salakhutdinov, R., Zemel, R.S., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: ICML (2015)
36. Yang, X., Zhou, S., Cao, M.: An approach to alleviate the sparsity problem of hybrid collaborative filtering based recommendations: The product-attribute perspective from user reviews. Mobile Networks and Applications (04 2019)

37. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A.J., Hovy, E.H.: Hierarchical attention networks for document classification. In: HLT-NAACL (2016)
38. Zhang, Q., Wang, L., Meng, X., Xu, K., Hu, J.: A generic framework for learning explicit and implicit user-item couplings in recommendation. IEEE Access 7, 123944–123958 (2019)
39. Zhang, Y., Ai, Q., Chen, X., Croft, W.B.: Joint representation learning for top-n recommendation with heterogeneous information sources. In: CIKM '17 (2017)
40. Zhang, Y., Lai, G., Zhang, M., Zhang, Y., Liu, Y., Ma, S.: Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval (2014)
41. Zheng, L., Noroozi, V., Yu, P.S.: Joint deep modeling of users and items using reviews for recommendation. CoRR abs/1701.04783 (2017)

ZHONGQIN BI received his Ph.D. degree in system analysis and integration from East China Normal University, China in 2009. He is currently assistant professor in College of Computer Science and Technology, Shanghai University of Electric Power. Dr. Bi has published more than 30 articles in refereed journals and conference proceedings. His main research interests are cloud computing, data process and quality control in smart grid.

SHUMING DOU is currently working toward a master's degree in the College of Computer Science and Technology, Shanghai University of Electric Power, Shanghai, China. His research interests include recommendation system, data process, and natural language processing.

ZHE LIU received his Master degree in electrical engineering from North China Electric Power University, China in 2016. He is currently engineer in State Grid Shanghai Electric Power Research Institute. He has published nearly 20 articles in refereed journals and conference proceedings. His main research interests are energy Internet and power quality.

YONGBIN LI received his MA. Sc in Computer Application Technology from Shanghai Jiaotong University, China in 2008. He is currently assistant professor in College of Computer Science and Technology, Shanghai University of Electric Power. MA. Li has published more than 10 articles in refereed journals and conference proceedings. His main research interests are data process and software engineering and he has won the third prize of Anhui provincial science and technology progress.

Received: September 25, 2019; Accepted: July 1, 2020.