Information Sciences 512 (2020) 1279–1292

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Recommendation system exploiting aspect-based opinion mining with deep learning method

Aminu Da'u[a,b,*], Naomie Salim[a], Idris Rabiu[a], Akram Osman[a]

[a] School of Computing, Faculty of Engineering, Univerti Teknologi Malaysia, Malaysia
[b] Hassan Usman Katsina Polytechnic, Katsina State, Nigeria

A R T I C L E I N F O

Article history:
Received 3 December 2018
Revised 10 October 2019
Accepted 21 October 2019
Available online 23 October 2019

Keywords:
Aspect Based Opinion Mining
Sentiment classification
Deep CNN
Tensor factorization
Recommendation systems
Ratings prediction

A B S T R A C T

With the developments of e-commerce websites, user textual review has become an important source of information for improving the performance of recommendation systems, as they contain fine-grained users' opinions that generally reflect their preference towards products. However, most of the classical recommender systems (RSs) often ignore such user opinions and therefore fail to precisely capture users' specific sentiments on products. Although a few of the approaches have attempted to utilize fine-grained users' opinions for enhancing the accuracy of recommendation systems to some extent, most of these methods basically rely on handcrafted and rule-based approaches that are generally known to be time-consuming and labour-intensive. As such, their application is limited in practice. Thus, to overcome the above problems, this paper proposes a recommendation system that utilizes aspect-based opinion mining (ABOM) based on the deep learning technique to improve the accuracy of the recommendation process. The proposed model consists of two parts: ABOM and rating prediction. In the first part, we use a multichannel deep convolutional neural network (MCNN) to better extract aspects and generate aspect-specific ratings by computing users' sentiment polarities on various aspects. In the second part, we integrate the aspect-specific ratings into a tensor factorization (TF) machine for the overall rating prediction. Experimental results using various datasets show that our proposed model achieves significant improvements compared with the baseline methods.

1. Introduction

With the explosive growth of information on the web, RSs play a vital role in addressing the issue of information overload, having been widely applied in many online services including social media and e-commerce websites. Collaborative filtering (CF) is the most widely used technique for RSs. The basic idea of this technique is that people who share similar behaviours in the past tend to have a similar preference in the future. Although CF methods have shown promising performance, one of their major challenges is the problem of data sparseness, which is characterized by the insufficient number of user ratings with a high number of items. This, however, affects the effectiveness of the recommendation systems.

With the recent advancement of e-commerce websites, it has been shown that user textual reviews, which contain rich information on different products can be utilized to alleviate the data sparsity problem, thereby enhancing the effectiveness of RSs. Generally, user reviews contain not only the user's comments on different aspects of products but also the user's

* Corresponding author at: School of Computing, Faculty of Engineering, Univerti Teknologi Malaysia, Malaysia.
E-mail addresses: dauaminu@graduate.utm.my (A. Da'u), Naomie@utm.my (N. Salim), oyoakram2@live.utm.my (A. Osman).

https://doi.org/10.1016/j.ins.2019.10.038
0020-0255/

fine-grained opinions towards various aspects of products. Essentially, these user opinions are very important as they reflect the user's preference towards products and consequently affect the accuracy of RSs. Unfortunately, most of the traditional RSs [1–3] often ignore such user's opinions for rating prediction, and consequently experience poor performance in the recommendation process.

Recently, a small number of works [4,5] have been introduced to utilize user's opinions to enhance the accuracy of RSs. For example, [4] used phrase-level opinion mining on user review texts to extract explicit product's aspects and the associated user's opinions to generate an explainable recommendation. A major limitation of this method is that it relies on the lexicon construction which is known to be time-consuming and labour-intensive. Yang et al. [5] introduced a unified framework that extracts aspects from textual reviews and integrates them into an extended CF filtering method for rating prediction. This approach uses a double propagation method that relies on a syntactic relation to expand opinion lexicons, and collectively extracts aspects and the associated user sentiments based on the dependency graph. One major challenge of this approach is that it heavily depends on the dependency parser that is prone to generate mistakes, especially when applied to online reviews. Thus, to overcome the above problems among others, in this research, we introduce an approach to exploit ABOM using the deep learning technique for improving the effectiveness of the recommendation system.

Our proposed work is inspired by the above approaches; however, it differs from them in several aspects. (1) Instead of using the lexicon or double propagation methods as used in prior methods, our proposed approach specifically explores using the deep learning technique to extract the product's aspects from reviews. Specifically, we employ a deep CNN model which has been shown to be effective in several *natural language processing* (NLP) tasks. The CNN is a nonlinear model that is easy to train and able to automatically capture the salient features from the review texts. The CNN model has been previously exploited for aspect extraction tasks [6,7]; however, unlike the traditional CNN-based methods, our CNN model is typically a multichannel convolutional neural network (MCNN) that exploits two different input layers, namely, the word embedding layer and part-of-speech (POS) tag embedding layer. The former is aimed at better learning of the semantic information of the document, while the latter is aimed at facilitating a better sequential labelling process. (2) In contrast to the above methods that basically focus on the rating prediction task based on user textual reviews, our proposed approach not only predicts the unknown ratings but also estimates and evaluates the quality of the extracted aspects that are utilized for the rating estimation.

The proposed method, named REAO (Recommender system Exploiting Opinion mining using deep learning methods), specifically consists of two major parts: aspect-based opinion mining (ABOM) and a rating prediction component. In the first component, we utilize a deep CNN model for aspect extraction from the review text and then apply an LDA technique [8] to transform the extracted aspect terms into latent factors. We finally compute the aspect-based ratings, forming a set of rating matrices with each matching to a particular aspect

In the second part, the generated aspect rating matrices are then integrated into an extended CF method for estimating the overall rating, which comprises the basis of the recommendation process. In this paper, as we deal with three different factors, namely, the user, item and aspect, and we particularly employ a tensor factorization (TF) technique to better capture the three sources of the information. To this end, we employ a CP-WOPT [9] model that has been previously used and shown to be very effective in dealing with sparse data for the RS.

We evaluated the model using real-world datasets and finally made comparisons with the baseline methods. The experimental results indicated that our proposed REAO approach performs better in comparison with the baseline models. Other contributions of this research are highlighted as follows:

- We propose an MCNN-based approach for ABOM exploiting two input channels, namely, word embedding and POS embedding channels.
- We propose an approach to incorporate the extracted aspect-based ratings into a three-dimensional TF technique for effective rating prediction.
- We carried out a series of extensive experiments to evaluate the performance of the proposed REAO approach in terms of both aspect extraction and rating prediction.

The rest of this paper is arranged as follows. Section 2 reviews the related existing methods. Sections 3 and 4 define the problem and discuss the overview of the proposed approach respectively. Sections 5 and 6 report the experiments and conclusion of the study accordingly.

2. Related work

This section reviews various research works related to this work. These include aspect extraction for opinion mining and review-based recommendation systems.

2.1. Aspect-based opinion mining (ABOM)

ABOM aims to extract aspects of products and classify the associated sentiment polarities of the user in the review [10]. In the past few years, several approaches have been proposed to study ABOM based on review texts. Most of the early approaches of aspect extraction are based on the frequency-based methods (Hu & Liu 2004), for which some constraints are used to identify the most frequently used nouns and noun phrases. The main drawback of this approach is that low-frequency words are often ignored. To overcome this problem, several other approaches have been proposed. These include rule-based approaches [11] that construct a set of rules to discover aspects, and topic-based approaches [12,13] that use statistical models such as LDA [8] to identify the topic distribution in a document. One critical issue with this approach is that it cannot capture the contextual information of documents, as they operate based on the bag-of-word process. Other approaches include supervised methods [14,15] that typically consider aspect extractions as a *sequence labelling* task. One of the limitations of the classical supervised approaches such as conditionals random fields (CRFs) [16] is that they are linear methods, and as such, they need a large number of datasets to effectively train the model.

With the recent successes of the artificial neural network, several efforts have been made to exploit the deep neural network for the ABOM [6,7,17,18]. For instance, Irsoy and Cardie [17] used recurrent neural networks (RNNs) to better extract multifaceted aspects of products based on the user textual review. The authors demonstrate the power of the RNN-based methods over the CRF-based approaches for aspect extraction. For further improvements, Pengfei et al. [19] introduced a similar method by using more complex variants of the RNNs based on the fine-grained word embeddings for aspect detection. Poria et al. [6] used a multilayer CNN architecture that applies Amazon and Google word embeddings with additional features to better facilitate the sequence labelling process. Specifically, the model integrates syntactic and linguistic features as additional features to further improve the accuracy of the model. A similar approach has been proposed by Xu et al. [7] for the aspect extraction task. The authors exploited a simple CNN architecture based on the IOB labelling scheme. The model uses a double embeddings procedure exploiting both domain-specific and domain-independents embedding for better aspect extraction. Different from the aforementioned methods, in this work, we specifically use the multichannel convolutional network (MCNN) framework that uses word embeddings and POS tags embeddings as 'the inputs to the networks for better learning of semantic information of words.

The second aspect of the ABOM is to determine the sentiment polarities on the constituent aspects of a product based on the reviews. The mainstream methodss for sentiment classification were generally based on either supervised methods such as supports vector machines (SVM) and neural networks [20,21] or lexicon-based methods [12]. In this paper, we specifically employ the approach utilized by Wang and Chen [22] to compute the user's sentiments on different aspects of the products.

2.2. Review-based recommender systems

Another aspect related to this research is review-based RSs. In the past few years, several research works attempted to exploit user textual reviews for enhancing the accuracy of RSs [1,2,23–25]. One of the early works that utilized review texts for RSs uses manually designed ontologies to generates free texts [23]. One major limitation of this method is that it is time-consuming and domain-dependent, as such its application is limited in practice. To incorporate latent topics with latent factors for improving the accuracy of recommendation, some topic-based methods [1,2,24] have been proposed. McAuley and Leskovec [1] uses an LDA-like technique [8] to integrate latent topics with the latent factors for the recommendation system. Zhang and Wang [24] integrated latent topics with latent factors for better representation of users and items. Another method in [2] has been proposed to exploit topic modelling to better learn user/item features from the users' textual content for improving the performance of RSs. Diao et al. [25] introduced a unified framework to jointly estimate aspect ratings and the associated user's sentiment for better-personalized item recommendation. An approach introduced by Zhang et al. [4] uses a lexicon construction technique to better discover the user explicit features and incorporate them into a TF method for enhancing the accuracy of the recommendation system.

With the advancement of the artificial neural network, several researchers have explored using deep learning techniques for improving the recommendation systems [26–30]. These include methods that use denoising autoencoders [30], CNNs [26,28] and RNNs [27,29] models. Even though our proposed work deals with the deep learning technique, it should be noted that we only use the deep learning technique specifically for the task of ABOM. Our REAO method is also related to aspect-based recommendation systems. Recently, different methods have been proposed to exploit aspect extraction for enhancing the accuracy of recommendation processes. For instance, Snyder et al. [31] exploits an aspect's dependency for rating prediction based on the good grief method. Diao et al. [32] proposed a method to jointly extract a product's aspects and infer user numerical ratings on products. Wang et al. [33] proposed a latent and aspect rating analysis (LARA) method to incorporate the user's preference into the CF process. Specifically, the model takes a set of review texts with the aspect's features and numerical ratings as input and generates latent aspects for the rating prediction. This method has been extended by introducing a generative method for latent aspect rating analysis that can work without aspect keywords [34].

The most closely related method to our proposed approach is the work of [4] which exploits a lexicon construction method for opinion mining. Specifically, the authors use phrase-level opinion mining to extract explicit features and the associated opinion words for improving the performances of personalized recommendation. However, this method typically relies on lexicon generation which is generally labour-intensive and often depends on manual parameter fine-tuning. Our proposed method is also related to the work of [5] which jointly performs the ABOM task and rating predictions using the double propagation method. The model utilized lexicon expansion for the double propagation technique to better discover aspects based on the reviews. However, this method is limited as it heavily relies on the dependency parser. Furthermore, the performance of the model generally depends on the accuracy of the grammar. Different from the above methods, in our approach, we introduce an ABOM method based on the multichannel CNN model to better extract the aspects of the product and the associated opinion words for a more accurate recommendation.

Table 1
Notations.

Notation	Description
$\boldsymbol{r}_{ij}$	The user overall ratings
r_{ijk}	The aspect of specific ratings
$\boldsymbol{D}ij$	The user review written on the item
R^k	Aspect rating matrix
$\mathcal{R}$	*Tensor of size* $I \times J \times K$

3. Problem definition

Suppose there is a set of products $P = \{ p_1, p_2 \ldots\ldots\ldots p_n \}$ in a review $\boldsymbol{D}ij$ written by a set of users $\boldsymbol{U} = \{ u_1, u_2 \ldots\ldots\ldots u_m \}$. Let R be the overall rating matrix of the size $I \times J$, where the entry r_{ij} represents the overall ratings of the user u_i towards product p_j. Assuming there are k aspects $A = \{ a_{1,} a_{2,} \ldots\ldots\ldots a_{k,} \}$ and the associated k aspect rating matrices $R^1, R^2 \ldots\ldots R^k$, with one for each aspect. In the following, we define the research problems that we endeavour to solve in this study.

- *Aspects extractions*: The main purpose of the aspect extraction is to extract the product's aspects mentioned in the review texts. An aspect is a set of words that describes a rating factor in a textual document. Examples of these aspects include: "screen", "battery" and "performance" for the laptop domain. It is assumed that there are k aspects contained in the reviews and given as $a_{1,} a_{2,} \ldots\ldots\ldots a_{k,}$
- *Aspect-based rating*: An aspect-based rating of a user about a product is a numerical rating showing the user's opinions towards the product. Let $A = \{ a_{1,} a_{2,} \ldots\ldots\ldots a_{k,} \}$ be a k aspect in a text review; then, the aspect-based rating of a user u_i on product p_j can be represented as a k-dimensional vector r_{ijk}.
  Similar to the overall rating R, for each aspect, there is an aspect-based rating matrix that can be represented as $R^1, R^2 \ldots\ldots R^k$, where r_{ijk} can be written as the entry of the aspect rating matrix R^k, which denotes the ratings on a product P_j for the aspect a_k by a user u_i.
- *Overall rating prediction*: Our goal here is to estimate the overall ratings r_{ij} for the product P_j not yet rated by the user u_i. We define a user's overall rating as a numerical rating reflecting the user's general opinions on a product.

Considering the fact that users' opinions on the various aspects generally reflect their overall ratings on the product, we incorporate the overalls ratings matrix R and the k aspect rating matrices $R^1, R^2 \ldots\ldots R^k$ into a 3-dimensional tensor $\mathcal{R}$ factorization method where $I \times J \times K$ is the size of $\mathcal{R}$. Table 1 shows some important notations used in this paper.

4. Overview of the proposed method

In this section, we present details of our proposed REAO model which exploits the aspect extraction method using a deep learning method for building the RS. Fig. 1 depicts the whole process of the proposed model which consists of two different components. First, we apply a deep learning technique for the aspect extraction from review texts using an MCNN model

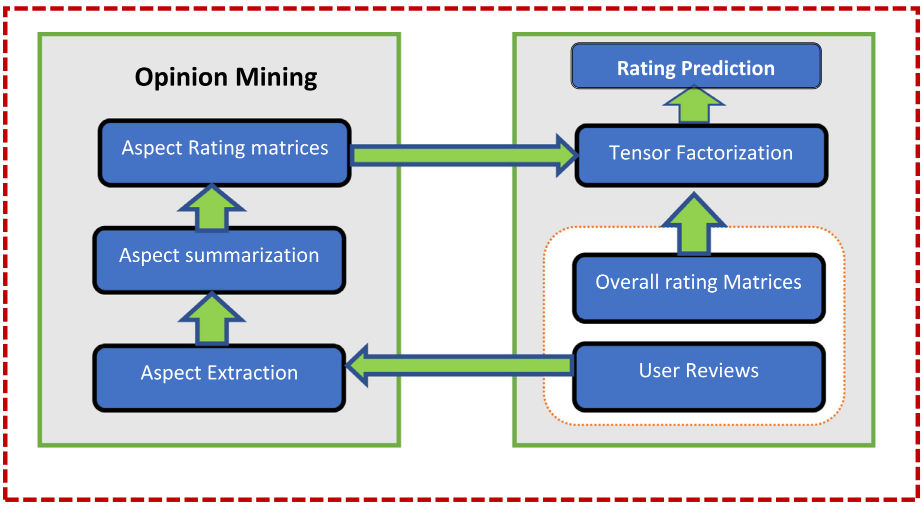

Fig. 1. An illustration of the proposed REAO method.

and then use an LDA model to generate a cluster of latent aspects. We then apply a lexicon method for computing the user sentiments on the aspects to generate aspect rating matrices. Second, we incorporate the aspect-based rating matrices together with the overall ratings into a tensor factorization method for rating estimation. We describe the details of the method in the following subsection.

4.1. Aspect extraction

Our goal here is to discover aspects and the associated opinion words and then estimate user-specific sentiments on individual aspects based on the reviews. To this end, we first develop an MCNN model for better extraction of aspects in the review. Then, the LDA model is applied for the opinion summarization, and finally, a lexicons method is employed to compute the associated user's sentiment (opinion) scores. The details of the procedure are described as follows.

4.1.1. The MCNN

Our proposed multichannel CNN method is an extended version of the CNN architecture proposed by [35]. It typically comprises input channels, convolution layers, max-pooling layers, and a fully connected layer. In the input layers, we specifically use word embeddings and POS tag embedding channels. For the word embedding channel, we used pre-trained word embeddings [36] for better learning of the semantics information of words. Formally, a text of $n - length$ can be represented as: $|X|_1^n = \{\, x_1 \ldots\ldots,\ x_n\},\ X \in R^K$.

For the POS tags embeddings, we use the one-hot-vector. Specifically, we transform each tag into a k-dimensional vector. Following [37], we utilize a *Stanford Tagger* with a set of 45 tags. This can formally be given as: $|S|_1^n = \{\, s_1 \ldots\ldots,\ s_n\},\ S \in R^{45}$.

We then apply convolution operations to extract the salient features using two different filter sizes for POS tags and word2vec features given as **P** and ***Z*** respectively. Let $w_p \in R^{hk}$ be a filter for matrix ***P*** and $w_z \in R^{hx45}$ be a filter for matrix **Z**, where h is the height of the filter. Then, a convolution generates features by:

$$C_i = f(w \cdot x_{i+h} + b) \tag{1}$$

where f and b represent a nonlinear function and a bias term respectively. This is applied to each window, $[x_{1:h},\ x_{2:h+1},\ \ldots x_{n-h:\,n,}\ ]$. With $w_p \in R^{n-k+1}$ and $w_z \in R^{n-k+1}$, for the ***P** and **Z***, respectively, we obtain the features map for matrix **P** which is given as:

$$c_p = \left[c_1^p, c_1^p \ldots . c_{n-h+1}^p\right] \tag{2}$$

Then, to generate the feature map for matrix **Z**, we have:

$$c_z = \left[c_1^z, c_1^z \ldots . c_{n-h+1}^z\right] \tag{3}$$

Essentially, different POS tags and semantic features can be extracted using several filters. We then apply a pooling operation to take the maximum elements in each generated feature. This can be represented as:

$< ct > \check{C} < ot >_p = \max[c_1^p, c_1^p \ldots . c_{n-h+1}^p]$ and $\check{C}_z = \max[c_1^z, c_1^z \ldots . c_{n-h+1}^p]$ for matrix ***P*** and matrix ***Z*** respectively.

The final features are obtained using the max-pooling operation by concatenating the POS and semantic feature.

This can be represented as $C = \ \check{C}_p \oplus \check{C}_s$. Here, $\oplus$ is the concatenationsoperator. As several filters are used for the *POS* and *semantic* features, the final feature is obtained as:

$$C = \check{C}_p^1 \oplus \ldots.. \oplus \check{C}_p^n \oplus \check{C}_s^m \oplus \ldots.. \oplus \check{C}_s^m \tag{4}$$

where ***n*** and ***m*** are the filters for *semantic* and *POS* features respectively.

Finally, for the model output, we specifically use a softmax function for generating the probability distributions over given aspects.

4.1.2. Aspect summarization

In the real-word situation, several aspects and opinion words are contained in the user textual review; however, many of these aspects may carry the same meaning. For example, aspects terms such as , *services, performance*, and *performances* can all represent the *service* aspect. Therefore, for aggregating the user sentiment polarities for these aspects there is a need for mapping the extracted aspects into latent aspects. To accomplish this, we employ latent Dirichlet allocation [8] which has been utilized in the previous related studies [38]. The inputsto the LDA method is the collection of reviews that include the aspects terms, and the outputsis the set of aspects, of which each aspect is comprised of a set of aspectsterms. Due to the characteristic of LDA, an aspect term can belong to several different groups. In essence, the number of aspects can be estimated experimentally.

4.1.3. Aspect-based ratings

This subsection presents the approach to compute the aspect-based rating matrices $R^1, R^2 \ldots. R^k$ based on the aspects and the associated opinion terms extracted from the previous method. To achieve this, we first calculate the sentiment scores for the aspects in the reviews and then take the ratio of the opinion words' polarities. Similar to [22], we adopt a lexicon-based approach using *Sent Wordnet* [39]. In this approach, each aspect-based rating is estimated based on the opinion terms

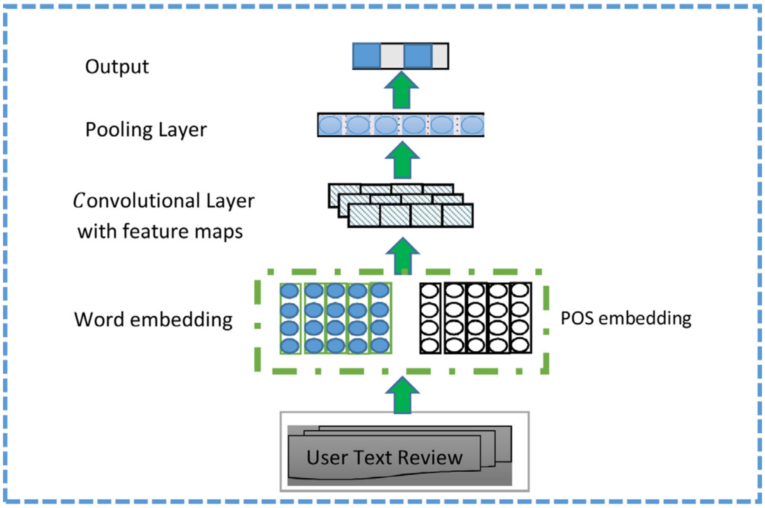

Fig. 2. An overview of the MCNN model for the aspect extraction.

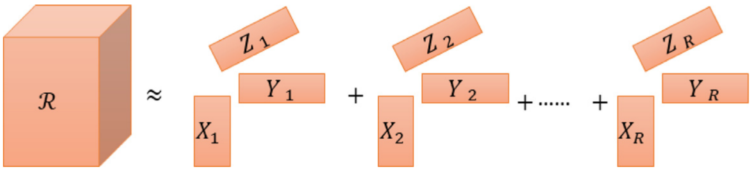

Fig. 3. Illustration of the CP decomposition.

associated with the aspects. As explained earlier, aspects are typically nouns or noun phrases while opinion words are typically adjectives. Given an aspects a_k in a review D_{ij}, we estimate the aspect-based ratings as follows:

$$r_{ijk} = \frac{\sum_{w \epsilon W_k}(D_{ij})OP(w)}{\left|W_k(D_{ij})\right|} \tag{5}$$

where W_k represents the sets of words contained in the reviews D_{ij} that are associated with the aspects a_k, and $OP(w)$ denotes the words' polarity scores based on the sent WordNet. As the aspect rating values express the attitude of the user on the product for the aspect, we normalize the aspect rating values so that they fall into the same range as the overall ratings (usually ranging from 1–5).

4.2. Overall rating prediction

As stated earlier, the ultimate goal of the RS is to estimate the overall rating r_{ij} for a product $\boldsymbol{p_j}$ that is not yet rated by the user $\boldsymbol{u_{i,}}$. Given the k aspect-specific ratings $R^1, R^2 \ldots. R^k$ (as computed in Section 4.1 above), we integrate the aspect rating matrix $R^1, R^2 \ldots. R^k$ and the overall rating matrix R into a 3rd order TF machine to predict the overall rating $r_{ij,}$.

Many different TF models such as HOSVD [5] and CANDECOMP/PARAFAC(CP) [9] can be used for computing TF. In this research, we specifically apply CP-WOPT [9], which is a variant of the CP model used in [9] and has the ability to better decompose a high-order tensor to a sum of rank-one tensors in a scalable manner. An illustration of the CP decomposition for tensor $\mathcal{R}$ is shown in Fig. 2. Formally, CP decomposition can be defined by factor matrices X, Y, Z of sizes $I \times R, J \times R, K \times R$, respectively, such that:

$$r_{ijk} = \sum_{r=1}^{R} x_{ir} y_{jr} z_{kr} \tag{6}$$

For all $i = 1, 2 \ldots, \ I, \ \ j = 1, 2 \ldots, \ J, \ and \ k = 1, 2 \ldots K.$
where r_{ijk} and $\boldsymbol{R}$ is the entry and the rank of tensor $\mathcal{R}$ respectively. $I \times R, \ J \times R, \ K \times R$ are the sizes of the $\boldsymbol{X}, \boldsymbol{Y}, and\, \boldsymbol{Z}$ respectively. Fig 3 depicts the CP decomposition for the tensor $\mathcal{R}$.

For the sake of better understanding of the CP-WOPT tensor method, we first describe some notations. Let $\mathcal{A}$ and $\mathcal{B}$ be two third-order tensors of equal size $I_1 \ \times \ I_2 \ \times I_3$, then their elementwise product represented as $\mathcal{A}$ *B is defined as:

$$(\mathcal{A} * \mathcal{B})_{i_1 i_2 i_3} = a_{i_1 i_2 i_3} b_{i_1 i_2 i_3}, \forall 1 \le i_n \le \ I_N$$

Additionally, let $I_1 \times I_2 \times I_3$ be the sizes of a tensor $\mathcal{A}$, then its norms can be given as:

$$\|\mathcal{A}\| = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \sum_{i_3=1}^{I_3} x^2{}_{i_1 i_2 i_3}}. \tag{7}$$

With the three matrices **A, B, C** of size $I_1 \times R, I_2 \times R, I_3 \times R$, ***A, B, C*** defines an $I_1 \times I_2 \times I_3$ tensor whose elements are given by:

$$([\![\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!])_{i_1 i_2 i_3} = \sum_{r=1}^{R} A_{i_1 r} B_{i_2 r} C_{i_3 r} \tag{8}$$

Finally, given a tensor $\mathcal{A}$ of size $I_1 \times I_2 \times I_3$, $\mathcal{A}$ can be given as a matricization. The mode-n matricization of $\mathcal{A}$ can be given as $\boldsymbol{A}_{(n)}$. Tensor element $\mathcal{A}_{i_1 i_2 i_3}$ maps to matrix $\boldsymbol{A}_{(n)}$ element (i_n, j) where

$$j = 1 + \sum_{\substack{k=1 \\ k \neq n}}^{3} (i_k - 1) J_k \tag{9}$$

with

$$J_k = \begin{cases} 1, & if\ k = 1\ or\ if\ k = 2,\ and\ n = 1. \\ \prod\limits_{\substack{m=1 \\ m \neq n}}^{k-1} I_m & otherwise \end{cases}$$

CP decomposition can be considered as a weighted least square problem [9] that considers only the observed entries and minimizes the following objective function:

$$fw(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) = \frac{1}{2} \sum_{i=1}^{J} \sum_{j=1}^{J} \sum_{k=1}^{K} \left\{ w_{ijk} \left(r_{ijk} - \sum_{r=1}^{R} x_{ir}\, y_{jr}\, z_{kr} \right) \right\}^2 \tag{10}$$

where w that has the same size as $\mathcal{R}$ represents a nonnegative tensor given as:

$$w_{ijk} = \begin{cases} 1\ if\ \mathrm{r_{ijk}}\ is\ known \\ 0\ if\ \mathrm{r_{ijk}}\ is\ missing \end{cases}$$

$$For\ all\ i = 1, 2 \ldots, I,\ j = 1, 2 \ldots, J,\ and\ K = 1, 2 \ldots .K.$$

The factor matrices ***X***, ***Y***, and ***Z*** can be learned by minimizing Eq. (10).

With the notations described earlier, Eq (10) can be replaced as:

$$fw(\mathbf{X},\ \mathbf{Y},\ \mathbf{Z}) = \frac{1}{2} \parallel \mathcal{W} * (\mathcal{R} - [\![\mathbf{X}, \mathbf{Y}, \mathbf{Z}]\!]) \|^2 \tag{11}$$

Assuming that $\mathcal{S} = \mathcal{W} * \mathcal{R}$ and $\mathcal{T} = \mathcal{W} * \|\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}\|$, then Eq. (11) can be rewritten as

$$fw(\mathbf{X},\ \mathbf{Y},\ \mathbf{Z}) = \frac{1}{2} \|\mathcal{S} - \mathcal{T}\|^2 \tag{12}$$

Essentially, the tensor S can be precomputed since W and $\mathcal{R}$ do not vary during the iteration.

The gradient of the objective function can be determined by estimating the partial derivative of *fw* w.r.t each element of the factor matrices. In matrix notation, the partial derivatives of *fw* w.r.t each element of factor matrices can be given as:

$$\begin{aligned} \frac{\delta fw}{\delta \boldsymbol{X}} &= (\boldsymbol{T}_{(1)} - \boldsymbol{S}_{(1)})(\mathbf{Z} \odot \boldsymbol{Y}) \\ \frac{\delta fw}{\delta \boldsymbol{y}} &= (\boldsymbol{T}_{(2)} - \boldsymbol{S}_{(2)})(\mathbf{Z} \odot \boldsymbol{X}) \\ \frac{\delta fw}{\delta \boldsymbol{z}} &= (\boldsymbol{T}_{(3)} - \boldsymbol{S}_{(3)})(\mathbf{Y} \odot \boldsymbol{X}) \end{aligned} \tag{13}$$

$\boldsymbol{S}_{(n)}$ *and* $\boldsymbol{T}_{(n)}$ can be computed from Eq. (9) where $\odot$ represents the Khatri-Rao product and is given for **Z** and **Y:**

$$\boldsymbol{Z} \odot \boldsymbol{Y} = \boldsymbol{z_1} \otimes \boldsymbol{y_1}\ \boldsymbol{z_2} \otimes \boldsymbol{y_2} \ldots\ldots \boldsymbol{z_R} \otimes \boldsymbol{y_R} \tag{14}$$

where $\otimes$ represents the vector Kronecker product.

Having obtained the gradient, we can now apply any first-order optimization method [40] to obtain the factor matrices ***X***, ***Y***, ***Z***. The predicted values of the rating that a user u_i will give for a product p_j can be represented as:

$$\hat{r}_{ij} = \sum_{r=1}^{R} x_{ir} y_{jr} z_{kr} \tag{15}$$

Table 2
SemEval datasets indicating the number of aspects and sentences. D1 and D2 represent the SemEval-2014 restaurant and SemEval-2014 laptop domain respectively.

Dataset	D1		D2	
	#Aspect	#Sentence	#Aspect	# Sentence
Train	3693	3045	2358	3041
Test	1134	800	654	800

5. Experimental study

To evaluate the performances of the proposed approach, we conducted series of experiments which aimsto answersthe following questions: (1) What is thenperformancesof the proposed REAO approach in terms of aspectsextractions and rating prediction? (2) What is the performancesof the REAO model in the condition of the cold start issue? (3) What are the impacts of varying the value of the parameter K?

5.1. Aspect extractions tasks

To validate the effectiveness of the proposed model in terms of aspect extraction, we evaluate the model using SemEval2014 datasets [41] which have been widely used in related works [6,42]. The datasets consist of review sentences with aspects from the restaurant and laptop domains. Table 2 shows the statistics of the datasets. The datasets are preprocessed in order to obtain clean texts. Specifically, all the texts are converted into lower case and split into separate sentences. All the stop words and the words with special and alphanumeric characters are removed to ensure the texts are free from noises.

5.1.1. MCNN settings

For the MCNN model settings, following the settings used in [35]; specifically, for each dataset we apply the following strategies: we used rectified linear units as the activation function for the network and three filter sizes of $(3, 4, 5)$, with 100 feature maps for each filter. A dropout rate of 0.5 and the L2 constraint of 3 are used. The model is trained through gradient descent over shuffled mini-batches with the Adadelta update rule. These values were selected based on grid search using the traditional 5-fold cross-validation.

5.1.2. Evaluation

To validate the impact of the MCNN method for the aspect extraction, we apply various settings to our MCNN model based on input channels and the feature accordingly:

- **MOD-1**: Here, the word embedding input layer is randomly initialized while the input layer using the POS tags is not used during the model training.
- **MOD-2**: Similar to the above settings, in this case, the input channel containing the POS tags is not used during the training. However, a pre-trained word2vec is used to initialize the word embedding input channel and is optimized during the training.
- **MOD-3**: In this case, all the input channels (word embeddings and POS tags embeddings) are considered for the training. In particular, we utilize the Google word2vec in the first channel and POS tags in the other channel.

These three versions of the MCNN settings are further compared with the following state-of-the-art approaches for aspect extraction:

- *DLIREC* [42]: This is the winning system in the SemEval-2014 challenges which uses semantic and lexical features for improving the accuracy of the model.
- $CNN + LP$ [6]: This is an aspect extraction approach which uses s deep multilayer CNN model with linguistic patterns for the aspect extraction.

For better evaluation of our aspect extraction model, we particularly apply the $F1$ score metrics that are calculated in terms of recall and precision [6,42]. Due to space limitations, we omit the detailed expression of the evaluation metrics.

5.1.3. Results analysis

Table 3 shows the performance of the aspects extractions task compared with the baseline methods. The results of the different variants of the model are reported in terms of the $F1$ score accuracy, and the best result is bolded in each case. It can be observed from Table 3 that the best performing version of our approach (Mod-3) significantly outperforms the best performing baseline method (CNN+P) with the improvement of 1.44% and 2. 33% on the D1 and D2 datasets, respectively, and the statistical tests show that the improvements are significant at the confidence levels of 0.05. As seen from Table 3,

Table 3
F1 score performance in terms of aspects extractions task.

Model	DLIREC	CNN+Pl	Mod-1	Mod-2	Mod-3
D1	74 0.55%	82 0.32%	81.55%	82.48%	**83.86%**
D2	84 0.01%	87 0.17%	85.76%	86.87%	**88.43%**

different variants of our MCNN settings have different performance. For example, Mod-3 outperforms all the other variants in each case while Mod-1 shows relatively worse performance. This better performance of the Mod-3 can be attributed to the use of two input channels exploiting pre-trained word embeddings and POS tag embeddings, compared to Mod-1 and Mod-2 in which only one input channel is utilized during the training. This indicates the impact of using the MCNN model exploiting pre-trained embedding and POS tag embeddings.

5.2. Rating prediction

For assessing the rating prediction performances of our proposed approach REAO model, we particularly use two publicly accessible datasets: The Amazon and Yelp review datasets, which are described as follows:

Amazon – This is the collection of datasets compiled by McAuley and Leskovec [43][1] from the Amazon reviews. The dataset consists of different products categories. This is the largest known dataset used for the RS evaluation. Because the original datasets are very large, we utilize five different sub-datasets for our experiments: *Musicals instrument* (*MI*), *Automotive* (*Auto*), *Pets supplies* (*Pet*), *V.games and Instant videos* (*IV*). Specifically, we utilize the 5-core category where each user or item has at least 5 interactions.

Yelp[2] – This dataset is from an online reviews platform which comprises reviews of various businesses in various metropolitan regions across 4 countries. As the datasets are originally too large, we preprocessed them to obtain a set with at least 5 ratings for each user.

To provide clean datasets, we preprocess the datasets as follows: we first remove all reviews with plain text, the reviews of unknown users and the reviews with only ratings from the datasets to ensure that the datasets do not contain an incomplete record.

Following the works in [43,44] we randomly split the datasets into a training (80%), test (10%) and validation (10%) set to fine-tune the hyper parameters. Then, for each dataset, the experimental results of the five folds are averaged to report the performance. The summary of the datasets is shown in Table 4.

5.2.1. Baselines

For better evaluation of our REAO approach in terms of rating prediction, the following baseline methods are used for comparison:

- MF (matrix factorization) [45]: This is a state-of-the-art CF method. It typically uses a rating matrix as input for rating prediction through estimation of two low ranked matrices.
- SVD++ model [46]: This is an extension of the MF method that utilizes both explicit and implicit feedback for improving the recommendation performance. It is widely used as a baseline for both explicit and implicit feedback-based CF methods.
- MTER (multitask explainable recommendation) [47]: This is a TF-based model that utilizes opinionated contents for the rating prediction task. It uses a lexicon-based method for modelling user opinions and joint TF for an explainable recommendation.
- Transnets [28]: This is a deep learning-based method for rating prediction. It uses user/item reviews as input for learning user/item representations.

We compare these four baselines with the best performing variant of our approach based on the three versions of the MCNN method as described in Section 4. For a fair comparison, for all the basslines, we directly utilized the original codes provided by the respective authors and tuned the parameters as described in their original papers [28,46,47].

5.2.2. Evaluation

To assess the rating prediction performance of our REAO approach, we use root *mean squared error* (RMSE) and *mean absolute error* (MAE) which is given as follows:

$$RMSE = \sqrt{\frac{\sum_{(u,i)\in T}\left(\breve{R}_{u,i} - R_{u,i}\right)^2}{|T|}} \tag{17}$$

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

[2] http://www.yelp.com/datasetchallenge/.

Table 4
Summary of the datasets for evaluating the rating prediction task.

Datasets	Users	Items	Reviews	Sparsity
MI	1498	1022	11,206	0.9927
Auto	2857	18,925	20,812	0.9996
IV	5109	1596	37,132	0.9954
Pet	17,975	9203	165,672	0.9990
V. games	23,313	11,322	236,144	0.9991
Yelp	174,251	65,200	1709,170	0.9998

Table 5
Comparisons results of the proposed REAO approach in terms of the rating predictions with the baseline methods. The best performing model is highlighted in bold. All of the reported improvements are statistically significant at $p<0.05$. $\Delta_{MF}, \Delta_{SV}, \Delta_{MT}$ and Δ_{TR} are the relative improvements (%) of REAO over the MF, SVD++, MTER and Transnet models respectively.

Datasets	Metric	MF	SVD++	MTER	Transnet	REAO	Improvement (%)			
							Δ_{MF}	Δ_{SV}	Δ_{MT}	Δ_{TR}
MI	RMSE	1.004	0.902	0.912	0.975	**0.802**	20.10	11.09	12.07	17.72
	MAE	0.768	0.683	0.682	0.802	**0.632**	17.69	7.42	7.30	21.17
Auto	RMSE	1.010	0.895	0.873	0.906	**0.814**	19.45	9.05	6.79	10.16
	MAE	0.760	0.654	**0.593**	0.690	0.598	21.39	8.63	−0.78	13.36
IV	RMSE	1.201	1.036	0.981	1.150	**0.974**	18.96	6.01	0.80	15.38
	MAE	0.997	0.793	0.791	0.863	**0.784**	21.39	1.14	0.87	9.16
Pet	RMSE	1.233	1.185	0.981	1.132	**0.972**	21.17	17.99	0.89	14.13
	MAE	1.019	0.898	0.798	0.847	**0.784**	23.06	12.73	1.80	7.49
V. Games	RMSE	1.200	1.162	1.051	1.103	**1.027**	14.40	11.61	2.27	6.89
	MAE	0.976	0.972	0.883	0.923	**0.817**	16.27	15.91	7.42	11.45
Yelp	RMSE	1.321	1.297	1.189	1.171	**1.131**	14.41	12.81	4.89	3.44
2017	MAE	1.131	1.037	0.972	0.999	**0.941**	16.75	9.25	3.19	5.79
Average	RMSE	1.162	1.080	0.988	1.073	0.953	18.08	11.43	3.49	11.29
	MAE	0.942	0.840	0.787	0.854	0.759	19.43	9.18	3.30	11.40

$$MAE = \frac{\sum_{(u,i)\in T} \left| \breve{R}_{u,i} - R_{u,i} \right|}{|T|} \tag{18}$$

where $\breve{R}_{u,i}$ and $R_{u,\,i}$ are the predicted and the actual ratings of user i and item j respectively. T is the testing test. Generally, the smaller values of the *RMSE and MAE* indicate better performance of the model.

5.2.3. Results and discussion

Table 4 shows the results of our REAO approach compared to other baselines in terms of RMSE and MAE. It should be noted that only the results from the best performing variant (MOD 3) of the MCNN model are used for comparison with the baseline methods. All the reported results are the average of 5-fold cross-validation. The best prediction accuracy of each sub-dataset is bolded.

It can be seen from Table 4 that our proposed REAO method significantly outperforms the baseline methods in most of the datasets and the improvement is statisticall*y significant at p* < 0.05. This shows the effectiveness of our proposed REAO method in terms of rating prediction compared to the baseline methods.

Several observations can be made from Table 5. First, one can see that all the models that utilize review text in addition to the numerical ratings (i.e., SVD++, MTER, Transnets, and REAO) apparently outperform the MF model, which merely relies on the user's ratings for the rating prediction. This confirms the earlier finding by[43,44] which shows the benefits of utilizing textual reviews for better performance of RS. It can be observed that the MTER model that typically utilizes user opinionated data as additional information outperforms all the other baseline models including SVD++ and Transnets. This indicates the benefit of utilizing the additional source of information in addition to the user and item latent factors for recommendation systems.

However, despite the good performance of the MTER approach compared with the other baselines, our proposed REAO model outperforms the MTER model with gains of 3.4%, and 2.61% on average in terms of RMSE and MAE respectively. Compared to MF and SVD ++, our proposed REAO model achieves significant gains of 18.08%, 19.43% and 11.43%, and 9.18% on average in terms of RMSE and MAE respectively. More importantly, our proposed approach outperforms the recently introduced Transnets model, which is also a deep learning-based method, with gains of 11.29% and 11.40% on average in terms of the RMSEs and MAEs respectively.

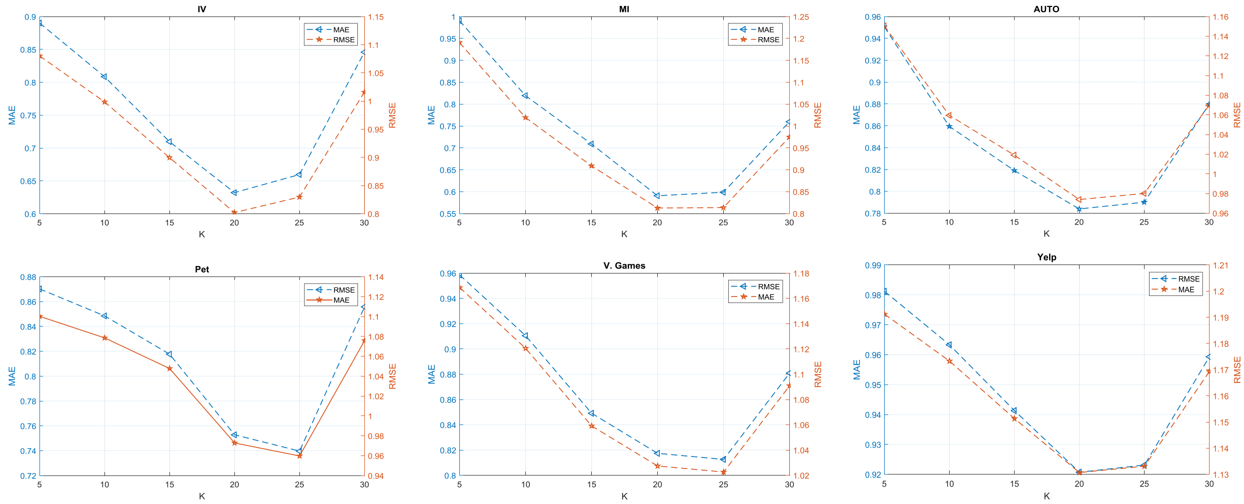

Fig. 4. Performance of the REAO model in terms of RMSE and MAE based on different values of K.

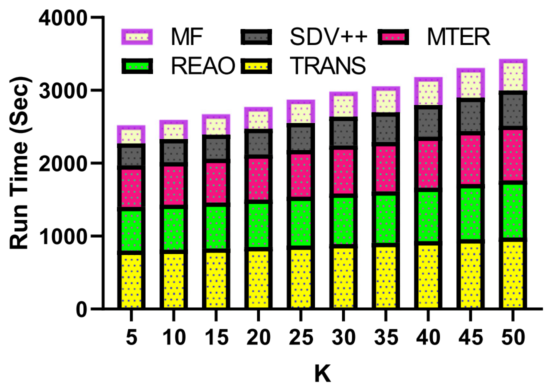

Fig. 5. The run time of the proposed model compared with the baseline methods.

To further investigate the efficiency of our proposed model, we also carry out other experiments with respect to the running time and make comparisons with the baselines. Fig. 5 shows the running time for our proposed method compared with the baselines. It can be observed from Fig. 4 that although our method shows higher computation time compared with some of the baselinesmethods, namely, MF, SVD++, and MTER, when, compared to the Transnets model, which also uses the deep learning-based method, our model records relatively smaller running time. Thus, we can conclude that the empirical running time of our method is relatively acceptable compared to the related deep learning-based method.

As seen from Fig. 5, the MF and SVD++ models record relatively smaller running time compared with the rest of the baselines including all the TF-based methods (i.e., MTER and REAO). This is attributed to the fact that tensor factorization-based methods typically use high parameterization and update high-order dimensions that include user, item and other additional source data, unlike the matrix MF-based methods that generally use relatively fewer parameters and update only two latent factors, usually user and item factors during the training.

The main advantage of our REAO approach compared to the baseline method is that our approach particularly takes into consideration the user's specific sentiment on various products' aspects, and it typically uses the deep learning method in the ABOM process which consequently contributes to the significant improvements of our model.

5.2.4. Sensitivity analysis

As stated earlier, we utilized the LDA procedure for generating K clusters of aspects. Therefore, to evaluate the impacts of the various values of K on the performance of the proposed model, we carry out different experiments by using different values of *K* that range from 5 to 30 with a step of 5 and calculate the RMSE and MAE results. Fig 4 indicates the performances of our model in terms of RMSE and MAE based on the different values of K for all the datasets.

It can be observed from the curves in Fig. 4 that the best value of K is within the range of 20 and 25 in most of the cases. It is important to mention that as the value of K is increasing, the one-one relationship of the factors and the aspects finally break; thus, increasing the value of K cannot always provide further enhancement. Therefore, in our experiment, we

Table 6
Top 5 aspects clusters.

tuners	software	mouthpiece	Mute	Microphone
guitar	program	cream	Stylus	Mics
capo	drivers	fog	Strings	Stand
guitars	interface	reeds	Cartage	Wireless
picks	windows	harmonica	Stick	Microphones

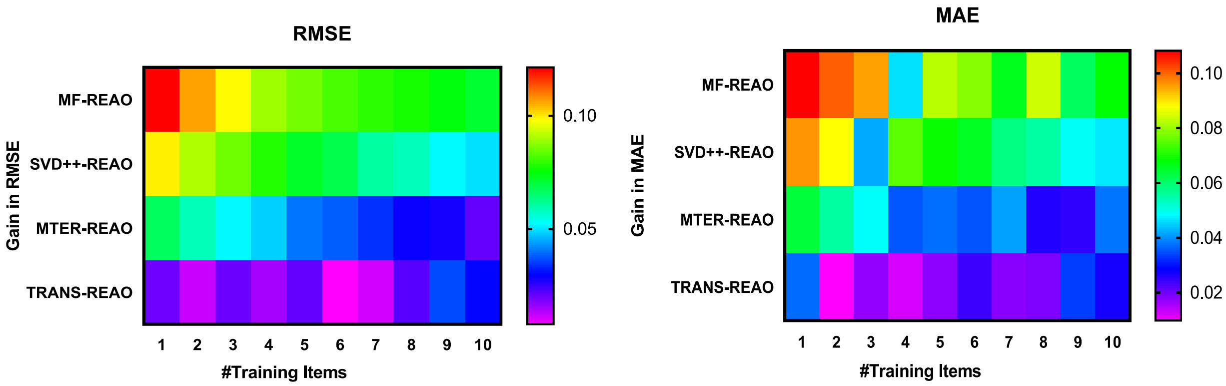

Fig. 6. Performance on the cold start problem.

select 20 as the K value. Table 6 shows the generated clusters by the LDA technique when the number of the clusters is set to 5.

5.2.5. Cold start problem

As indicated in Table 4, the datasets are typically sparse; however, it is very difficult to provide an effective rating prediction in sparse rating data. This sparseness generally leads to *the cold start problem*, which is one of the main problems in the rating prediction. Given limited ratings data, essentially, items/users are modelled only with biased terms. By utilizing user textual reviews in modelling user/item representation, the problem of cold start can be alleviated by our REAO approach considerably, since rich information about the user opinions is contained in the text reviews.

To demonstrate the efficiency of our proposed method in alleviating the issue of the cold start, following the works in [43,44], we present the performances of a subset of users based on their ratings. Specifically, we choose users with 1 *to* 10 ratings in the training sets and compute the averages of the *RMSE* and *MAE* for those users in our experiments. Fig. 6 shows the gain in *RMSE* and *MAE* values against the number of ratings by the users in the training sets. Here, the gain in *RMSE* and *MAE* is given as the average *RMSE* and *MAE* of the baselines minus that of our proposed approach respectively. A positive value shows the better performances of our proposed method in the *cold start condition* compared to the baseline methods. Here, we report the experimental results on the *musical instrument* (MI) datasets.

It can be shown that our method outperforms all the other baseline methods including the HFT model that also utilizes text review. This clearly shows that our model is more effective in using review text particularly because of the user preference and sentiment that are derived from the individual aspect and used along with the user ratings.

6. Conclusion

In this paper, we presented an RS model based on the ABOM approach. We first demonstrated how a deep learning-based technique can be used for aspect extraction based on the user textual reviews. Then, we showed how the extracted aspects can be utilized for computing aspect-based ratings, which can finally be integrated into a tensor factorization machine for enhancing the accuracy of the recommendation system.

The proposed approach is essentially made up of two parts, namely, ABOM and the rating prediction. After the aspect extraction process, we then used an LDA procedure to cluster the extracted aspects and the underlying opinion words into latent aspects. We then applied a lexicon approach to computing the aspect-based ratings based on the user's opinions on each constituent aspect in the reviews. To better handle the integration of the user's opinions on different aspects alongside the overall ratings of the user, a TF method with three dimensions is used, thereby computing the representation of the underlying factors.

The main impression of our proposed approach is that users' opinions and preferences on various aspects of a product can essentially be used to reflect their overall ratings on the item, and users' opinions are very important feedback for ensuring an accurate performance of RSs. We carried out a series of experiments on different datasets. The experimental results showed that our REAO approach performs better compared to the baseline approaches. As a future direction, we think that using a neural *attention network* for the ABOM task and incorporating more source information is a worthwhile endeavour to explore further improvement of the RS accuracy.

Declaration of competing interest

The authors hereby declare that they have no any conflict of interest whatsoever.

References

[1] J. McAuley, J. Leskovec, Hidden factors and hidden topics: understanding rating dimensions with review text, in: Proceedings of the Seventh ACM Conference on Recommender Systems, 2013, pp. 165–172.
[2] G. Ling, M.R. Lyu, I. King, Ratings meet reviews, a combined approach to recommend, in: Proceedings of the Eighth ACM Conference on Recommender Systems RecSys ', 14, 2014, pp. 105–112.
[3] Y. Tan, M. Zhang, Y. Liu, S. Ma, Rating-boosted latent topics: understanding users and items with ratings and reviews, in: Proceedings of the IJCAI International Joint Conferences on Artificial Intelligence Organization, 2016, pp. 2640–2646. 2016-Janua.
[4] Y. Zhang, G. Lai, M. Zhang, Y. Zhang, Y. Liu, S. Ma, Explicit factor models for explainable recommendation based on phrase-level sentiment analysis, in: Proceedings of the Thirty-Seventh International ACM SIGIR Conference on Research & Development in Information Retrieval - SIGIR ', 14, 2014, pp. 83–92.
[5] C. Yang, X. Yu, Y. Liu, Y. Nie, Y. Wang, Collaborative filtering with weighted opinion aspects, Neurocomputing 210 (2016) 185–196.
[6] S. Poria, E. Cambria, A. Gelbukh, Aspect extraction for opinion mining with a deep convolutional neural network, Knowl. Based Syst. 108 (Sep. 2016) 42–49.
[7] H. Xu, B. Liu, L. Shu, P.S. Yu, Double embeddings and CNN-based sequence labeling for aspect extraction, in: Proceedings of the Fifty-Sixth Annual Meeting of the Association for Computational Linguistics, 2018, pp. 592–601.
[8] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent Dirichlet allocation, J. Mach. Learn. Res. 3 (4–5) (2003) 993–1022.
[9] E. Acar, D.M. Dunlavy, T.G. Kolda, M. Mørup, Scalable tensor factorizations for incomplete data, Chemom. Intell. Lab. Syst. 106 (1) (2011) 41–56.
[10] A. Da'u, N. Salim, Aspect extraction on user textual reviews using multi-channel convolutional neural network, PeerJ Comput. Sci. 5 (2019) e191.
[11] Q. Guang, L. Bing, Opinion word expansion and target extraction through double propagation, Comput. Linguist. 37 (1) (2009) 1–19.
[12] I. Titov, R. McDonald, A joint model of text and aspect ratings for sentiment summarization, in: Proceedings of the ACL08 HLT, 51, 2008, pp. 308–316.
[13] H. Wang, M. Ester, A sentiment-aligned topic model for product aspect rating prediction, in: Proc. 2014 Conf. Empir. Methods Nat. Lang. Process., 2014, pp. 1192–1202.
[14] W. Jin, H.H. Ho, A novel lexicalized HMM-based learning framework for web opinion mining, in: Proceedings of the Twenty-Sixth Annual International Conference on Machine Learning - ICML '09, 2009, pp. 1–8.
[15] J. Lafferty, A. Mccallum, F.C.N. Pereira, F. Pereira, Conditional random fields, in: Proceedings of the Eighteenth International Conference on Machine Learning 2001 (ICML 2001), 2001, pp. 282–289.
[16] L. Shu, H. Xu, B. Liu, Lifelong learning crf for supervised aspect extraction, in: Proceedings of the ACL Fifty-Fifth Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers), 2, 2017, pp. 148–154.
[17] O. Irsoy, C. Cardie, Opinion mining with deep recurrent neural networks, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 720–728.
[18] P. Liu, S. Joty, H. Meng, Fine-grained opinion mining with recurrent neural networks and word embeddings, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2015), 2015, pp. 1433–1443.
[19] L. Pengfei, J. Shafiq, M. Helen, Fine- grained opinion mining with recurrent neural net- works and word embeddings, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 1433–1443.
[20] D.V.N.V.N. Devi, C.K. Kumar, S. Prasad, A feature based approach for sentiment analysis by using support vector machine, in: Proceedings of the IEEE Sixth Interenational Conf. Adv. Comput., 2016, pp. 3–8.
[21] J. Yoon, H. Kim, Multi-Channel lexicon integrated cnn-bilstm models for sentiment analysis, in: Proceedings of the Conference on Computational Linguistics and Speech Processing, 2017, pp. 244–253.
[22] F. Wang, L. Chen, Review mining for estimating users' ratings and weights for product aspects, Web Intell. 13 (3) (2015) 137–152.
[23] N. Jakob, S. Ag, Beyond the stars exploiting free-text user reviews, in: Proceedings of the TSA09, 2009, pp. 57–64.
[24] W. Zhang, J. Wang, Integrating topic and latent factors for scalable personalized review-based rating prediction, IEEE Trans. Knowl. Data Eng. 28 (11) (2016) 3013–3027.
[25] Q. Diao, M. Qiu, C.-Y. Wu, A.J. Smola, J. Jiang, C. Wang, Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS), in: Proceedings of the Twentieth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '14, 2014, pp. 193–202.
[26] L. Zheng, V. Noroozi, and P.S. Yu, "Joint deep modeling of users and items using reviews for recommendation," pp. 1–10, 2017.
[27] A. Da'U, N. Salim, Sentiment-Aware deep recommender system with neural attention networks, IEEE Access 7 (2019) 45472–45484.
[28] R. Catherine, W. Cohen, TransNets: learning to transform for recommendation, in: Proceedings of the RecSys'17, 2017, pp. 288–296.
[29] B. Hidasi, M. Quadrana, A. Karatzoglou, D. Tikk, Parallel recurrent neural network architectures for feature-rich session-based recommendations, in: Proceedings of the Tenth ACM Conference on Recommender Systems RecSys 2016, 2016, pp. 241–248.
[30] B. Wang and M. Liu, Deep learning for aspect-based sentiment analysis. CS224N Proj., pp. 1–9, 2015.
[31] B. Snyder, B. Snyder, R. Barzilay, R. Barzilay, Multiple aspect ranking using the good grief algorithm, in: Proceedings of the NAACL HLT, 2007, pp. 300–307.
[32] Q. Diao, M. Qiu, C.-Y. Wu, A.J. Smola, J. Jiang, C. Wang, Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS), in: Proceedings of the Twentieth ACM SIGKDD International Knowledge Discover Data Mining - KDD '14, 2014, pp. 193–202.
[33] H. Wang, Y. Lu, C. Zhai, Latent aspect rating analysis on review text data : a rating regression approach, in: Proceedings of the KDD'10, 2010, pp. 1–10.
[34] H. Wang, Y. Lu, C.X. Zhai, Latent aspect rating analysis without aspect keyword supervision, in: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 138, 2011, pp. 618–626.
[35] Y. Kim, Convolutional neural networks for sentence classification, in: proceedings of the Emperical Methods in Natural Language Processing (EMNLP), 2014, pp. 23–31.
[36] T. Mikolov, W. Yih, G. Zweig, Linguistic regularities in continuous space word representations, in: Proceedings of the NAACL-HLT, 2013, pp. 746–751.
[37] S. Jebbara, P. Cimiano, Aspect-Based relational sentiment analysis using a stacked neural network architecture, in: Proceedings of the Artificial Intelligence, 2016, pp. 1–9. 29 August-2.
[38] X. Ma, X. Lei, G. Zhao, X. Qian, Rating prediction by exploring user ' s preference and sentiment, Multimed. Tools Appl. (2017) 1–20.
[39] S. Baccianella, A. Esuli, and F. Sebastiani, "SentiWordNet 3. 0 : an enhanced lexical resource for sentiment analysis and opinion mining," vol. 0, pp. 2200–2204, 2008.
[40] E. Acar, D.M. Dunlavy, T.G. Kolda, An optimization approach for fitting canonical tensor decompositions, Contract 25 (2009) no. February, p. n/a-n/a.
[41] M. Pontiki, J. Pavlopoulos, SemEval-2014 task 4: aspect based sentiment analysis, in: Proceedings of the Eighth International Workshop on Semantic Evaluation, 2014, pp. 27–35.
[42] Z. Toh, W. Wang, DLIREC: aspect term extraction and term polarity classification system, in: Proceedings of the Eighth International Workshop on Semantic Evaluation (SemEval 2014), 2015, pp. 235–240.

[43] J. McAuley and J. Leskovec, *Hidden factors and hidden topics: Understanding rating dimensions with review text*. 2013, pp. 165–172.
[44] L. Zheng, V. Noroozi, P.S. Yu, Joint deep modeling of users and items using reviews for recommendation, in: Proceedings of the Tenth ACM International Conference on Web Search Data Mining WSDM 2017, 2017, pp. 425–433.
[45] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Comput. (Long. Beach. Calif). 42 (8) (2009) 30–37.
[46] R.M. Bell, Y. Koren, Scalable collaborative filtering with jointly derived neighborhood interpolation weights, in: Proceedings of the IEEE International Conference on Data Mining, ICDM, 2007, pp. 43–52.
[47] N. Wang, H. Wang, Y. Jia, and Y. Yin, "Explainable recommendation via multi-task learning in opinionated text data," 2018.