Expert Systems With Applications 140 (2020) 112871

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Weighted aspect-based opinion mining using deep learning for recommender system

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

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

ARTICLE INFO

Article history:
Received 2 March 2019
Revised 6 July 2019
Accepted 13 August 2019
Available online 14 August 2019

Keywords:
Aspect-based opinion mining
Convolutional neural network
Deep learning
Collaborative filtering
Recommender system, Rating prediction

ABSTRACT

The main goal of Aspect-Based Opinion Mining is to extract product's aspects and the associated user opinions from the user text review. Although this serves as vital source information for enhancing rating prediction performance, few studies have attempted to fully utilize it for better accuracy of recommendation systems. Most of these studies typically assign equal weights to all aspects in the opinion mining process, however, in practices; users tend to give different priority on different aspects of the product when reaching overall ratings. In addition, most of the existing methods typically rely on handcrafted, rule-based or double propagation methods in the opinion mining process which are known to be time-consuming and often inclined to errors. This could affect the reliability and performance of the recommender systems (RS). Therefore, in this paper, we propose a weighted **A**spect-based **O**pinion mining using **D**eep learning method for **R**ecommender system (**AODR**) that can extract product's aspects and the underlying weighted user opinions from the review text using a deep learning method and then fuse them into extended collaborative filtering (CF) technique for improving the RS. The proposed method is basically comprised of two components: (1) Aspect-based opinion mining module which aims to extract the product aspects from the review text to generate aspect rating matrix. (2) Recommendation generation component that uses tensor factorization (TF) technique to compute weighted aspect ratings and finally infer the overall rating prediction. We evaluate the proposed model in terms of both aspect extraction and recommendation performance. Experiment results on different datasets show that our **AODR** model achieves better results compared to the baselines.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

RS aims to tackle the problem of information overload in online e-commerce transactions and socian network platforms. CF is the most popular technique for RS which relies on the similarity among users/items based on their past behaviors. One major shortcoming of the CF methods is the issue of data sparsity, which is typically characterized by the low coverage of user ratings within the items owing to the relatively large number of users and items (Qiu, Gao, Cheng, & Guo, 2016). Recent works (Bao, Hui, & Zhang, 2014; Ling, Lyu, & King, 2014; McAuley & Leskovec, 2013) have shown that user textual reviews can be utilized to effectively alleviate the sparsity problem of CF methods. These opinions are typically very important in that they explain the user's preference that triggers the purchase decision on a product.

Recently, several methods have been introduced to exploit user's opinions for improving the predictive performance of RS (Wang, Liu, & Yu, 2012; Wu & Ester, 2015). However, most of these methods typically assign equal weights (importance) for all aspects while in real-life applications, aspects are differently treated by the user based on their importance. Intuitively, item rating is a weighted combination of various aspects and that the diverse preference of different users is determined by different weights put by different users. Moreover, most of these methods particularly rely on mechanical approaches such as rule-based and handcrafted methods which are known to be time-consuming and often error-prone in the estimation process.

In view of the above inspirations, this paper proposes a unified method to integrate aspect-based opinion mining and CF method for enhancing the predictive performance of the RS. The model comprises two different components: (1) Aspect-based opinion mining and (2) Recommendation generation. In the first component, we particularly design a Multichannel Convolutional

* Corresponding author at: School of Computing, Faculty of Engineering, Univerti Teknologi Malaysia, Skudai, Johor, 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.eswa.2019.112871
0957-4174/© 2019 Elsevier Ltd. All rights reserved.

Neural Network (MCNN) to deal with the aspect extraction task. Our MCNN model is an extension of the CNN model introduced in (Kim, 2014), however different from the traditional CNN model, our MCNN model specifically utilizes two different input layers, namely, word embedding and POS tag embedding layer (details of the MCNN model is given in Section 4).

In the second component, we specifically use a TF technique to first estimate the weighted aspect ratings and then infer the overall rating of the user on an item. The proposed AODR model is evaluated in terms of both rating prediction and item ranking using Amazon and Yelp datasets. The model is finally compared with the existing approaches and various experiments show better results of the proposed AODR approach compared to the baselines. In the following we highlight the major contributions of the paper:

- We introduce a deep learning based method for better extracting the product's aspects and the underlying user opinions from the user textual reviews for RS.
- We introduce an aspect weighted rating estimation method based on the TF technique to alleviate data sparseness and reduce the number of model parameters for enhancing the predictive performance of the RS.
- We extensively carry out different experiments to assess the effectiveness of our proposed model in terms of both aspect extraction and recommendation performance (rating prediction and item ranking tasks).

The remainder of this paper is arranged as follows: we present the related work in Section 2. Sections 3 and 4 defines the research problem and overviews the proposed model respectively. Sections 5 and 6 presents a series of experiments and conclusion of the paper respectively.

2. Related work

In this section, we review different studies particularly related to this research which can principally be categorized into two directions, namely, aspect-based opinion mining/sentiment analysis and review based RS.

2.1. Aspect-based opinion mining

Aspect-based opinion mining/sentiment analysis basically deals with the extracting of aspects and the underlying sentiment polarity classification from the review text of a given item (Wu & Ester, 2015). For the past years, several methods have been introduced to investigate Aspect-based opinion mining on user text reviews. These include unsupervised methods such as rule-based (Guang & Bing, 2009) and topic modeling (Jia, Zhang, Lu, & Wang, 2014). Although these methods have gained many achievements, one of the shortcomings of these approaches is that they ignore the contextual information of words which has been shown very important in the Natural Language Processing (NLP) tasks. Other methods include supervised approaches (Lafferty, Mccallum, Pereira, & Pereira, 2001) which typically consider aspect extraction as a sequence labeling task. One major drawback of the traditional supervised methods such as Conditional Random Field (CRF) (Shu, Xu, & Liu, 2017) is that they are linear models as such they require relatively larger datasets for effective training.

With the recent achievements of deep learning, several attempts have been made to utilize deep learning techniques for aspect-based opinion mining (Irsoy & Cardie, 2014; Liu, Joty, & Meng, 2015; Poria, Cambria, & Gelbukh, 2016; Xu, Liu, Shu, & Yu, 2018). For example, Irsoy (Irsoy & Cardie, 2014) used a recurrent neural network (RNN) model for the aspect extraction and demonstrated the superior performance of the model over the CRF based methods. For better improvement, an approach introduced in Liu et al. (2015) utilized a more advanced variant of the RNN model for the aspect extraction. A multilayer CNN based method was proposed by Poria (Poria et al., 2016) for aspect extraction. The authors used additional syntactic and linguistic features for better performance of the model. Another CNN based model was proposed in Xu et al. (2018) using a double embedding architecture that exploits domain dependent and domain independent word embeddings for aspect extraction.

Different from the above approaches, this paper introduces a deep MCNN method for extracting product's aspects using two different input channels: word embedding (Mikolov, Yih, & Zweig, 2013) for capturing the semantic information of the words, and Part of Speech (POS) tag embedding for improving the sequential labeling of the aspects.

The second task of aspect-based opinion mining is to classify the polarity scores associated with each aspect in a sentence. Generally, the approaches for classifying the user sentiment polarity can be achieved using either supervised approaches such as Support Vector Machine (SVM), Neural Network (Devi, Kumar, & Prasad, 2016; Yoon & Kim, 2017) or lexicon based approaches (Titov & McDonald, 2008). In this paper, we specifically adopt the strategy used in Wang and Chen (2015) for computing the sentiment polarity scores associated with the different aspects of products in the user textual reviews.

2.2. Review-based recommender system

Over many years, several works have been introduced to exploit user text reviews for improving the performance of RS (Diao et al., 2014; Jakob & Ag, 2009; Ling et al., 2014; McAuley & Leskovec, 2013). One of the earliest work to exploit reviews for RS typically uses manually designed ontologies to generate free text (Jakob & Ag, 2009). However, this method is generally not suitable for integrated RSs as it is time-consuming and domain independent. To integrate fine-grained information for improving predictive performance, some topic modeling methods (Ling et al., 2014; McAuley & Leskovec, 2013) have been introduced. For example, McAuley (McAuley & Leskovec, 2013) used topic modeling to combine latent topics with the latent factor for rating prediction. A method has been proposed in Ling et al. (2014) to use topic modeling for learning features from user reviews together with factorization machine for rating prediction. Diao et al. (2014) proposed a joint modeling method to simultaneously model aspect ratings and user sentiments for improving the overall rating prediction. Another method proposed in Zhang et al. (2014) integrated explicit product features into a TF machine for RS. One major limitation of all the above methods is that they mostly ignore the user-specific sentiment on the different aspects of the product.

With the recent achievements of representation learning, many deep learning-based RSs have been proposed. These include a method that uses CNN model. For example, Zheng, Noroozi, and Yu (2017) exploited the deep learning method for RS. The authors used two parallel CNN models for modeling user and item representation exploiting user and item review respectively. This method was later improved by Catherine and Cohen (2017) by introducing an additional layer to better learn user/item representation. Some studies used RNNs model for feature representation learning to build RS. For example, Lu, Smyth, Dong, and Smyth (2018) introduced a coevolutionary latent model for capturing the coevolution of the user/item latent features. Jing and Smola (2017) used LSTM models for multitasking learning to simultaneously predict the returning times of user and recommend-items. Da'u and Salim (2019) exploited RNN model with attention for sentiment aware recommendation. The authors used a semi-supervised topic modeling to extract product's aspects from the user text review and the associated sentiment and finally integrated them into an LSTM model using interactive attention for recommender system. AutoEncoder (AE) model has been shown to be powerful in feature representation learning. As such several authors have utilized the model to better learn the item and user representation for improving the performance of the RS. Bai, Fan, Tan, and Zhang (2017), exploited stacked denoising AE (SDAE) to better learn feature representation for RS. Similar method was introduced in X. Li & She (2017) to better learn probabilistic latent variables to easily incorporate side information from multimedia source for RS.

Although our work is directly related to the deep learning methods, however, it should be noted that we exploit deep learning method specifically for the task of aspect-based opinion mining. Our proposed approach is also related to aspect-based RSs. Recently, some few approaches have been introduced to utilize the product's aspects for rating prediction. For example, Snyder and Barzilay (2007) modeled aspect dependencies for aspect-based rating prediction using a good grief algorithm. Diao et al. (2014) proposed an approach for simultaneous aspect extraction and rating prediction. Wang, Lu, and Zhai (2010) introduced a Latent and Aspect Rating Analysis (LARA) model which specifically estimates each latent rating on the product's aspects. This model was further improved by a generative method for LARA that does not require aspect keywords (Wang, Lu, & Zhai, 2011). Bauman, Liu, and Tuzhilin (2017) used the latent factor strategy for modeling the relationship between the aspects, user and products rating prediction.

The most closely related work to our proposed model in terms of rating prediction is the work of Zhang et al. (2014) which simultaneously exploit opinion mining and CF for RS. The authors apply phrase-level sentiment analysis for extracting user's explicit factors and opinion terms for improving the performance of rating prediction. However, this approach is purely based on lexicon construction which is often time-consuming. Our work is also related to a method introduced by Yang, Yu, Liu, Nie, and Wang (2016), which simultaneously consider opinion mining and rating prediction for CF. The authors typically used double propagation method to extract aspect and opinion terms for rating prediction.

Different from these methods, our proposed method typically uses aspect extraction method which is particularly based on deep learning technique and instead of directly using the extracted opinions from the text review, we additionally use a TF method for estimating the aspect weighted ratings and finally infer the rating prediction for RS.

3. Problem formulation

Assuming there is a set of items denoted as $\mathbf{P} = \{p_1,\ p_2\ldots\ \ldots\ \ldots p_n\}$ and a set of users denoted as $\mathbf{V}= \{v_1,\ v_2\ldots\ \ldots\ \ldots v_n\}$. Let $\mathbf{R}$ of size $I \times J$ represents a user-item overall matrix, where the entry r_{ij}represents the rating of user v_i on an item p_j. Intuitively, only a subset of products is reviewed by a user in the textual review. We use a matrix $\mathbf{M} = [m_{ij}]_{I \times J}$ of indicator variables m_{ij} to denote whether r_{ij} is observed ($m_{ij} = 1$) or not ($m_{ij} = 0$). The research problems we aim to address in this paper are defined as follows:

Aspect extraction: The goal of this task is to extract the product's aspects reviewed by a user in a user textual review. Aspects are a set of words that describe a rating factor in a textual document. We assume there are aspects mentioned in the review given as $a_1, a_2, \ldots\ \ldots\ \ldots a_{K,}$

Aspect-based ratings: Aspect rating is a numerical rating showing the user's preference on a specific aspect of an item. Let A= $\{a_1, a_2, \ldots\ \ldots\ \ldots a_{K,}\}$ be a K aspect in a review text, then the aspect level ratings of user v_i and $p_j$can beerepresented as a K-dimensional vectors$\boldsymbol{r}_{ij}$, where r_{ijk} represents a numerical rating on an aspect $a_{k,}$. For each aspects, there is an aspects rating matrix that can be represented as $\boldsymbol{R}^1$, $\boldsymbol{R}^2\ldots\ \ldots\ \boldsymbol{R}^K$, and r_{ijk} can be given as the entry of aspect rating matrix $\mathbf{R}^{\mathrm{k}}$, which indicates the rating on the product $p_j$for the aspect $a_{k,}$ by the user v_i.

Table 1
Notations.

Notation	Description
$\mathbf{P} = \{p_1,\ p_2\ldots\ \ldots\ \ldots p_n\}$	The set-of-users
$\mathbf{V} = \{v_1,\ v_2\ldots\ \ldots\ \ldots v_n\}$.	The set-of-items
$\boldsymbol{r}_{ij}$	The user-overall-ratings
$\hat{r}_{ij}$	The predicted ratings
m_{ij}	The reviews written by the user.s
A= $\{a_1, a_2, \ldots\ \ldots\ \ldots a_{K,}\}$	The sets of aspects
r_{ijk}	The aspect-specific-ratings
$\boldsymbol{w}_{ij}$	An aspect weight-vector
R^k	Aspect rating matrix
$\bar{R}^k$	Aspect weight
$\mathcal{R}$	Tensor of size $I \times J \times K$

Aspect weight: Let $\boldsymbol{w}_{ij}$ be an aspects weights vectors of length-K, where w_{ijk} is a numerical rating of aspect a_k. Ifs an aspectsa_k of items$j$ is not numerically rated or reviewed by a user i, we have $w_{ijk} = 0$. We use a tensor $\mathcal{L} = [I_{ijk}]_{I \times J \times K}$ of indicator variables $I_{ijk} \in \{0,\ 1\}$ to represent whether r_{ijk} is known($I_{ijk} = 1$) or not known ($I_{ijk} = 0$).With the input of the aspect rating matrices $\boldsymbol{R}^1, \boldsymbol{R}^2\ldots\ \ldots\ \boldsymbol{R}^K$ and the overall rating $\mathbf{R}$, we can learn the weight placed on each aspect $a_{k,}$ of the item p_j by the user v_i. Those weightss can be integrated into the aspects-rating-matrices to generate weighted aspects-rating-matrices $\bar{\boldsymbol{R}}^1$,$\bar{\boldsymbol{R}}^2$......$\bar{\boldsymbol{R}}^K$.

Overall Rating Prediction: Following the work in Wang et al. (2012), we can aggregate the weighted aspect rating matrices and thee overall ratings into a three-dimensional array, 3rd – tensor $\mathcal{R}$. Where $I \times J \times K$ is the size of $\mathcal{R}$. Our goal here is to predict the overalls ratings $\boldsymbol{r}_{ij}$ for the item not yet ratedd by the users. Some important notations are presented in Table 1.

4. Overview of the model

This section presents the detail of our proposed AODR approach which exploits aspect-based opinion mining and TF technique for RS. Fig. 1 illustrates the entire AODR framework. It is comprised of two main components. First, we design an aspect-based opinion mining method using deep learning technique from the user textual review and then use a lexicon approach to estimate the user sentiment scores on each aspect of the product for generating aspect specific rating matrices. Second, we integrate the generated aspect rating matrices along with the user explicit ratings into a TF technique to compute aspect weighted ratings and finally infer the overall rating prediction forming the basis for the item recommendation. Details of the methods are described in the following subsections:

4.1. Aspect-based opinion mining

Aspects and opinion terms are typically made up of nouns, noun phrases or adjective words which are contained in text reviews being analyzed. Our goal here is to extract the aspect and the opinion terms and then compute the user sentiment polarities associated with the extracted aspects. To achieve that, we introduced a multichannel convolutional neural network (MCNN) architecture specifically for the aspect extraction task and then applied a Latent Dirichlet Allocation (LDA) technique to cluster the generated aspects. We finally used a lexicon method to estimate the aspect ratings. Detail of the MCNN architecture and other steps are presented in the following subsections.

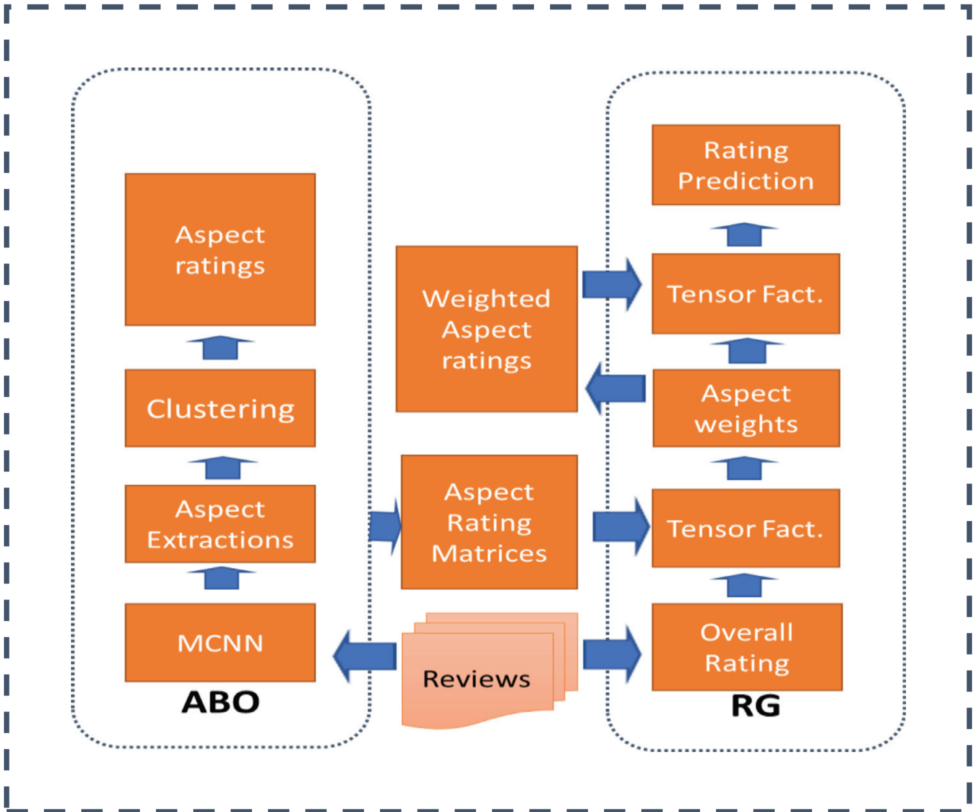

Fig. 1. General framework of AODR comprising Aspect-Based Opinion mining (**ABO**) and Recommendation Generation (**RG**) components respectively.

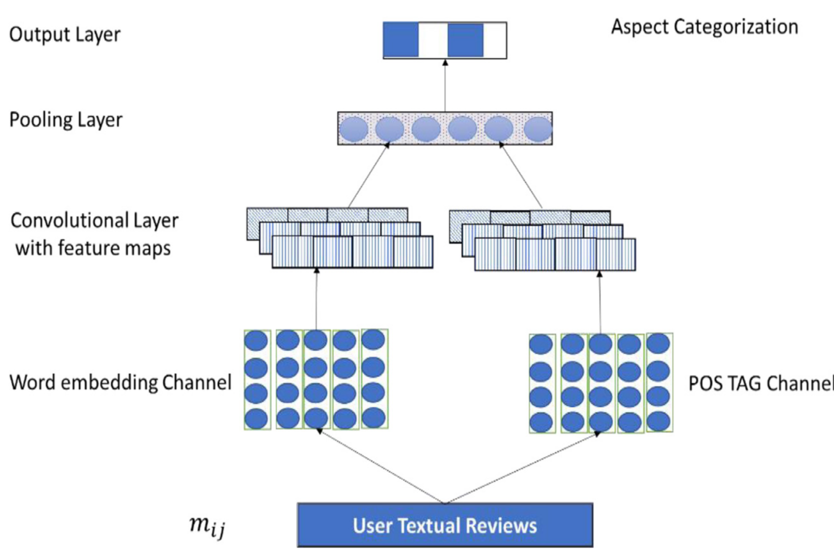

Fig. 2. The MCNN architecture for Aspect extraction with the word embedding and POS embedding channel.

4.1.1. MCNN for aspect extraction

Fig. 2 shows an overview of our MCNN architecture. The model is an extension of the CNN architecture introduced in Kim (2014). The model comprises two CNN channels: word embedding and POS embedding channel. For the word embedding, the main idea is to learn the semantic and contextual information of texts. To achieve this, we apply a pre-trained word2vec (Mikolov et al., 2013) which is trained using a large corpus of Google news based on the CBOW (continues bag of word model) architecture.

Formally, each word in the sentence is mapped into a low dimensional vector by a lookup layer transformation leading to a matrix $X \in R^{n \times k}$. For the POS Tag embedding channel, the main idea is to better facilitate the sequential tagging process based on the POS Tag. Following Jebbara and Cimiano (2016), we specifically utilize a Stanford POS Tagger which is encoded as a 45-dimensional vector. This can formally be given as $w_z \in R^{n \times 45}$.

4.1.1.1. Convolution layer. The main idea of the convolutional layer is to extract the most salient features from the user textual reviews. This layer specifically generates local features by using two different filter sizes for the word embedding and POS tag channel accordingly. Let $w_x \in R^{h \times k}$ is a filter and h is the height of the filter to the matrix X for the word embedding channel. Formally, the features generated by a convolutional operation can be given as:

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

where f is a nonlinear function and b corresponds to a bias term. By sliding the window on all the possible words in X, we obtain a feature map given as:

$$c_x = \left[c_1^x, c_2^x \ldots c_{n-h+1}^x\right] \tag{2}$$

with $c_x \in R^{n-h+1}$. Similarly, for the POS tag embedding channel, we apply different filter $w_z \in R^{h \times l}$ and obtain a feature map given as:

$$c_z = \left[c_1^z, c_2^z \ldots c_{n-h+1}^z\right] \tag{3}$$

with $c_z \in R^{n-h+1}$. In practice, we can apply different filters to generate different semantic and POS tag features respectively.

4.1.1.2. Pooling-layer. Next, pooling operation is then applied to capture the maximum elements of the contents. Formally, this can be represented as:

$$\check{C}_x = \max(c_x) \quad \text{and} \quad \check{C}_z = \max(c_z) \tag{4}$$

After the pooling operation, the final features are then obtained by concatenation of the semantic and POS features using a filter. This is formally represented as $C = \check{C}_x \oplus \check{C}_z$. where $\oplus$ is the concatenation operator. As we use different features for the POS and semantic features, we get the final feature, which can be given as:

$$C = \check{C}_x^1 \oplus \quad \ldots.. \oplus \check{C}_x^n \oplus \check{C}_z^1 \oplus \quad \ldots.. \oplus \check{C}_z^m \tag{5}$$

where $\boldsymbol{n}$and $\boldsymbol{m}$ are the filters for semantic and POS features respectively.

4.1.1.3. Output layer. Finally, the softmax function is applied for generating the output labels. Here, we particularly consider the aspect extraction as a sequence labeling process. Formally the output can be obtained as:

$$O = w \cdot (c \circ r) + b \tag{6}$$

where O *is a* masking operator and $r = R^{n+m}$ is a sample drawn from *bernoulli* distribution.

4.1.2. Aspect clustering

This subsection describes how the product's aspects and the associated opinion words can be summarized in the form of clusters which can be used for estimating the aspect-based ratings. In practice, several aspect terms are mentioned in user text reviews; however, many of these aspect terms may refer to the same aspect group. For instance, aspect terms like *performance* and performances, *service* and services can all depict the *service* aspect. Thus, to aggregate the sentiments for these various aspects, we need to map the extracted aspect terms into latent aspects. To achieve this, we exploit the LDA procedure which has been used in the previous related works (Ma, Lei, Zhao, & Qian, 2017). The input to the LDA methods is the collection of reviews including the aspect terms, and the output is the aspects sets, each aspect is composed of a set of aspect terms. LDA allows that one aspect can belong to many various clusters. In practice, the number of aspects can be determined experimentally. Due to the space limitation, details of the LDA model can be found in Blei, Ng, and Jordan (2003).

4.1.3. Aspect-based rating

When aspect terms and the associated opinion words are extracted and clustered, one can easily estimate the relevant polarities of each aspect of a particular item. Thus, to estimate the aspect specific rating matrices $\boldsymbol{R}^1$, $\boldsymbol{R}^2$……$\boldsymbol{R}^K$, similar to the work of Wang, Chen., Wang, and Chen (2012), we first, compute the sentiment polarity scores for each aspect in the review and then take the ratio of the word polarity. To this end, we employ a lexicon based method using Senti Wordnet Baccianella, Esuli, and Sebastiani (2008). In this method, each aspect rating is computed based on the opinion word associated with the aspect. Formally, given an aspect a_k in a review D_{ij} we compute the aspect rating as follows:

$$r_{ijk} = \frac{\sum_{w \in W_k}(D_{ij})\ OP\ (w)}{\left|W_k(D_{ij})\ \right|} \tag{7}$$

where W_k corresponds to the set of words in the review D_{ij} that is associated with the aspect a_k and $OP(w)$ represents the words polarity score based on the SentiWordnet.

4.2. Aspect weight estimation

To compute the aspect weights, we use a three dimensional TF, $\mathcal{W}$ which is the best way to capture the interaction between a three-dimensional arrangement of our data corresponding to the user, item and aspect. In this way, each element in the tensor match to a parameter w_{ijk}. To this end, we specifically adopt CANDECOMP/PARAFAC(CP) model (Acar, Dunlavy, Kolda, & Mørup, 2011) which has been widely used in the literature (Li et al., n.d.; Wang et al., 2012) and proved effective in dealing with a high order tensor decomposition (Acar et al., 2011). The tensor $\mathcal{W}$ can be decomposed as follows:

$$\mathcal{W} \approx \sum_{r=1}^{R} \boldsymbol{x}_r \circ \boldsymbol{y}_r \circ \boldsymbol{z}_r \tag{8}$$

where R and the operator $\circ$ denotes the number of rank-one components and the vector outer product respectively. $\boldsymbol{x}_r$, $\boldsymbol{y}_r$ and $\boldsymbol{z}_r$ are the column vectors in the corresponding factor matrixes **X, Y,** and **Z.**$I \times R$, $J \times R$ and $K \times R$ are the sizes of **X, Y,** and **Z** respectively. Element-wise, Eq. (8) can be replaced as:

$$\breve{w}_{ijk} = (\boldsymbol{x}_r, \boldsymbol{y}_r,\ \boldsymbol{z}_r) = \sum_{r=1}^{R} x_{ir} \cdot y_{jr} \cdot z_{kr}, \tag{9}$$

where each row $\boldsymbol{x}_r$, $\boldsymbol{y}_r$ and $\boldsymbol{z}_r$ *of* these matrices represent the user, item and aspect weight latent factors. The predicted ratings $\hat{r}_{ij}$ based on the model can be estimated from the aspect ratings and weight vector as follows:

$$\hat{r}_{ij} = \breve{w}_{ij}^T \boldsymbol{r}_{ij} = \sum_{k=1}^{K} \breve{w}_{ijk} \cdot r_{ijk} \tag{10}$$

To compute the optimized values of **X, Y** and **Z** parameters w.r.t prediction error, the objective function f is minimized:

$$f = \frac{1}{2} \sum_{i=1}^{I} \sum_{j=1}^{J} m_{ij} \left(r_{ij} - \hat{r}_{ij}\right)^2 \tag{11}$$

Subjects to the following constraints:

$$g_{ijk} \equiv -w_{ijk} \leq 0, \tag{12}$$

$$h_{ij} \equiv \sum_{k=1}^{K} I_{ijk} \cdot w_{ijk} - 1 = 0 \tag{13}$$

For all $i = 1, \ldots, I$, $j = 1, \ldots, J$, and $k = 1, 2\ldots\ldots, K$,
where g_{ijk} *and* h_{ij} are shorthands to be used in the sequel. Following the *PHR* method (Rockafellar, 1973), the

$$\frac{\partial\phi}{\partial x}=\sum_{j=1}^{J}m_{ij}\cdot e_{ij}\cdot\left(\sum_{k=1}^{K}l_{ijk}\cdot r_{ijk}\cdot z_k\right)\otimes y_j+\sum_{j=1}^{J}m_{ij}\cdot(v_{ij}+\rho\cdot h_{ij})\cdot\left(\sum_{k=1}^{K}l_{ijk}\cdot z_k\right)\otimes y_j+\curlywedge\cdot\sum_{j=1}^{J}\sum_{k=1}^{K}l_{ijk}\cdot(\rho\cdot w_{ijk}-u_{ijk})\cdot y_j\otimes z_k$$

$$\frac{\partial\phi}{\partial y}=\sum_{i=1}^{I}m_{ij}\cdot e_{ij}\cdot\left(\sum_{k=1}^{K}l_{ijk}\cdot r_{ijk}\cdot y_k\right)\otimes x_j+\sum_{j=1}^{J}m_{ij}\cdot(v_{ij}+\rho\cdot h_{ij})\cdot\left(\sum_{k=1}^{K}l_{ijk}\cdot z_k\right)\otimes x_i+\curlywedge\cdot\sum_{i=1}^{I}\sum_{k=1}^{K}l_{ijk}\cdot(\rho\cdot w_{ijk}-u_{ijk})\cdot x_i\otimes z_k$$

$$\frac{\partial\phi}{\partial z}=\sum_{i=1}^{I}\sum_{j=1}^{J}e_{ij}\cdot I_{ijk}\cdot r_{ijk}\cdot\ x_i\otimes z_j+\sum_{i=1}^{I}\sum_{j=1}^{J}I_{ijk}\cdot v_{ij}\cdot\ x_i\otimes z_j+\rho\cdot\sum_{i=1}^{I}\sum_{j=1}^{J}I_{ijk}\cdot h_{ij}\ \cdot\ x_i\otimes z_j+\curlywedge\cdot\sum_{i=1}^{I}\sum_{j=1}^{J}l_{ijk}\cdot(\rho\cdot w_{ijk}-u_{ijk})\cdot x_i\otimes y_j$$

Fig. 3. Partial derivatives of the objective function ϕ.

constrained objective function f can be converted into the unconstrained objective function as given below:

$$\phi=f+\sum_{i=1}^{I}\sum_{J=1}^{J}m_{ij}\cdot v_{ij}\cdot h_{ij}+\frac{\rho}{2}\sum_{i=1}^{I}\sum_{J=1}^{J}m_{ij}\cdot h_{ij}^2+\frac{1}{2\rho}\sum_{i=1}^{I}\sum_{J=1}^{J}\sum_{k=1}^{K}I_{ijk}\cdot\left\{\left[\max\left(0,u_{ijk}+\rho g_{ijk}\right)\right]^2-u_{ijk}^2\right\}\tag{14}$$

where v_{ij} and u_{ijk} are the multipliers of the equality-constraints on h_{ij} and inequality constraints g_{ijk}, and ρ *is the* penalty parameter.

Let $e_{ij}\equiv\hat{r}_{ij}-r_{ij}$ represents the model prediction error. The partial derivatives of the unconstrained-objective-functions ϕ w.r.t. model parameters $\boldsymbol{x}_i$, $\boldsymbol{y}_j$, $\boldsymbol{z}_k$ are demonstrated in Fig. 3. Where $\curlywedge$ denotes whether $u_{ijk}+\rho\cdot g_{ijk}$ is greater than 0 ($\curlywedge=1$) or not ($\curlywedge=0$), and the operator $\otimes$ represents the element-wise multiplication. With these, the gradient-descent algorithms can be used for estimating the optimal-matrices **X, Y,** and **Z**.

Having computed **X, Y,** and **Z**, and hence $\mathcal{W}$ computed using Eq. (8). Now we can obtain the weighted aspect rating matrix $\bar{\boldsymbol{R}}^k$ as follows:

$$\bar{\boldsymbol{R}}^k=\mathcal{W}\otimes\boldsymbol{R^k},\tag{15}$$

where the matrix $\mathcal{W}$ represents the weights users gives on the aspects a_k of items, meaning that each entry is estimated by $\hat{r}_{ijk}=w_{ijk}\times\ \boldsymbol{r}_{ijk}$.

4.3. Overall rating-prediction

Having estimated the weighted aspect rating matrix, the overall user ratings can be predicted when those ratings are available. To achieve that, similar to Wang et al. (2012), we integrate the overall rating matrix R with the weighted aspect rating matrices $\overline{\boldsymbol{R}}^1,\overline{\boldsymbol{R}}^2....\bar{\boldsymbol{R}}^K$ forming a new 33rd-orders-tensor $\bar{\mathcal{R}}$ which can be factorized by utilizing the CP WOPT approach (Acar et al., 2011). Assuming that the factor-matrices $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are the results from the CPs decompositions of $\bar{\mathcal{R}}$. The predicted-value of the ratings that user v_i will give for a product $p_j$can be obtained as:

$$\tilde{r}_{ij}=\hat{r}_{ij1}=\sum_{d=1}^{D}a_{id}b_{jd}c_{1d}\tag{16}$$

where the D is the positive-integer which represents the D-dimension of the tensor.

5. Experiment study

In this section, we present different experiments to assess the performance of our proposed approach based on the two components of the model: Aspect extraction and recommendation module. Our experiments aim to answer the following questions: (1) what is the performance of the proposed AODR model in terms of aspect extraction task. (2) What is the performance of the AODR model in terms of the item recommendation (rating prediction and Top-N item ranking)?

Table 2
Statistics of the datasets for aspect extraction.

Datasets	#Review	#Sentence	#Aspect
Cello Phone	99	740	49
Nikon	34	346	96
MP3	95	1716	57
DVD	41	546	67
Canon	45	597	79

5.1. Aspect extraction task

To validate the aspect extraction performance, we carry out experiments on the Amazon datasets designed by Hu and Liu (2004) which have been previously used in the literature (Popescu & Etzioni, 2005; Poria et al., 2016). The datasets comprise textual reviews from five different products, namely, Canon, Nikon, DVD, MP3, and Cellophane. The datasets were annotated with aspects present in each sentence and their opinion polarity. Table 2 shows the statistic of the datasets.

5.1.1. Evaluation

For evaluating the parformance of our proposed approach in term of aspect extraction, we compare our model with the following baselines: (1) CNN+LP (Poria et al., 2016), the state-of-the-art method for Aspect extraction using CNN model. (2) OPINE (Popescu & Etzioni, 2005), the unsupervised learning method that uses product's features for opinion mining. (3) Pop-dep (Guang & Bing, 2009), a method that uses double propagation approach with word dependency for aspect extraction. For a fair comparison, we directly use the results of all the baselines as presented in their respective studies. These baselines are compared with the following different settings of our MCNN model:

(1) ***MCNN*** − ***Random*:** This is similar to the CNN-random setting proposed in Kim (2014). It typically considers only one input channel of the CNN for training and ignores the POS Tag embeddings. Here, the word embedding channel is randomly initialized.
(2) ***MCNN*** + ***W2V***: Similar to the above, this setting also uses only one input channel, but in this case, unlike the above setting, the word embedding layer is initialized with a pre-trained word2vec (Mikolov et al., 2013). This version is particularly aimed to examine the impact of the pre-trained word embeddings.
(3) ***MCNN*** + ***W2V*** + ***POS*:** Here, all the two input layers (word embeddings and POS Tag embeddings) of the network are used for training. Particularly, we use the pre-trained word2vec in the first input and POS Tag embeddings in the second input.

Table 3
Aspect extraction performance in terms of F1 score.

Datasets	Opine	Prop-Dep	CNN+P	MOD-1	MOD-2	MOD-3
Cello phone	83.57%	88.90%	**90.44%**	82.88%	84.44%	88.94%
Nikon	82.89%	82.47%	84.87%	76.63%	80.10%	**86.84%**
MP3	80.81%	87.95%	**89.27%**	78.42%	83.80%	88.93%
DVD	84.26%	83.89%	87.19%	79.75%	87.30%	**88.54%**
Canon	79.98%	85.26%	88.64%	78.20%	81.29%	**89.46%**

Table 4
Statistics of the Amazon and Yelp datasets for rating prediction evaluation.

Datasets	Users	Items	Reviews
Musical Instruments (MI)	67,005	14,115	84,408
Automotive (Auto)	133,254	47,539	188,387
Instant Video (IV)	112,539	23,367	153,733
Yelp	169,257	63,300	1,659,678

For brevity, we use **MOD-1, MOD-2,** and **MOD-3** to represent MCNN-random, MCNN+W2V, and MCNN + W2V + POS respectively.

For our MCNN settings, following the work in Kim (2014), particularly, we use filter sizes of (3, 4, 5) for both embedding and POS convolution. We apply 100 feature maps for each filter and use a dropout of 0.5. The model is trained through stochastic gradient descent (SGD). We use ReLU (rectified linear unit) as the activation function and 128 as the hidden rate. All the parameters were determined based on the 5-folds cross-validation strategy.

For the evaluation measure, in this paper, we use the F1 score metric which is computed in terms of precision and recall and widely used in the literature (Guang & Bing, 2009; Popescu & Etzioni, 2005; Poria et al., 2016).

5.1.2. Analysis of the aspect extraction results

Table 3 indicates the results of the aspect extraction task in comparison with the baselines. The performances of the three different versions of the MCNN model are recorded for each dataset. The results are presented in terms of F1 score accuracy and the best results are indicated in bold for each dataset.

It can be shown that the best performing setting (MOD-3) of the MCNN outperforms the other baselines on most of the datasets and based on the statistical test, all the gains are significant at the confidence level of 95%. From Table 3, it can be observed that CNN+P performs better than other baselines. This clearly indicates the benefits of the deep learning-based methods over the handcrafted and unsupervised methods for the aspect extraction task. As can be seen from Table 3, that different versions of the model show different performance in various cases. MOD-3 outperforms all the other versions in all the cases followed by the MOD-2 while the MOD-1 records relatively lowest performance in all the cases. This indicates the impact of using two input channels, leveraging pre-trained word embeddings and POS tag embedding for aspect extraction. These results clearly indicate the quality of our extracted aspects for building the recommendation system.

Two important factors can be considered as the reasons for the better performance of our model: (1) the POS embeddings, which helps to better facilitate sequence labeling for the aspect detection. (2) the pre-trained embeddings which are trained on a large corpus of reviews and helps in better capturing of the semantic information of words.

5.2. Recommendation generation

To investigate the performance of our proposed model in terms of the item recommendation, we evaluate the model on both rating prediction and item ranking tasks using Amazon and Yelp datasets. The datasets are described in the following:

Amazon – This is the largest dataset used for RSs evaluation. The dataset is collected by McAuley (McAuley & Leskovec, 2013) and comprised of reviews and metadata from different varieties of Amazon products. The dataset which originally comprises 24 individual product categories has been used by many researchers in the literature (Catherine & Cohen, 2017; Ling et al., 2014; Seo, Huang, Yang, & Liu, 2017; Zheng et al., 2017). For the original datasets are too large, we specifically use three categories of the datasets for our experiments: Musical instruments (MI), Automotive (Auto) and Instant video (IV). In particular, we use the five-core version (where each user or item has at least 5 interactions).

Yelp - This is a dataset from an online review platform comprising reviews of various businesses in different metropolitan regions across four countries. As the dataset was originally very large, we pre-processed it to get the set with at least five interactions.

To provide clean datasets, we first removed all reviews with plain texts, the review of unknown users and the reviews with ratings only from the datasets. Similar to McAuley and Leskovec (2013) and Zheng et al. (2017), we randomly categorized our datasets into 80% and 10% for training and testing respectively where the rest 10% of the set is reserved for validation to fine tune the hyperparameters. The Statistics of the datasets are given in Table 4 accordingly.

5.2.1. Evaluationmetrics

To better assess the performance of our proposed approach in terms of item recommendation, we evaluate the system using both rating prediction and item ranking performance. For the rating performance, we use Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) metrics which have been widely used in the literature (D'Addio, Domingues, & Manzato, 2017; Kim, Park, Oh, Lee, & Yu, 2016). These metrics are given as follows:

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

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

where T is the testing set. Smaller values of *RMSE*/*MAE* indicate high accuracy.

For the Top-N ranking performance, we select the top-100 items with the highest predicted ratings for each user and use the prec@N (*pre*@10) and mean-average-precision (MAP) metrics. The prec@N estimates how many relevant-items are returned w.r.t n sample of the total ranking: this is given as:

$$pre@10 = \frac{relevant\ items\ in\ n}{n} \tag{19}$$

We use this metric with $n = 10$. The MAPs metric, in turns, evaluates the whole rankings but emphasizes more for occurrencessof the relevant items in early positions of the rankings. It is a measure that generates a value corresponding to the average of j quarries, where each query generates ranking scores average of differents$n$ precision levelss.

Formally, lets$\{i_1 \ldots\ldots i_m\}$ of associated-items for query $q_i \in Q$ andR_{jk} be the results returned from the first item up to the

Table 5
Comparison results of the proposed model with the baselines, in terms of the rating prediction. The symbol * denotes the improvement over the second best model is significant at 95%.

Method	MI		Auto		IV		Yelp	
	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
MF	1.1704	0.9542	1.4367	1.1875	0.9583	0.7521	1.2571	0.9586
HFT	1.0061	0.7621	1.2634	0.9863	0.8172	0.5687	0.9875	0.7895
RBLT	1.0105	0.7705	1.2513	0.9805	0.8061	0.5587	0.9719	0.7709
AODR-1	1.0140	0.7780	1.2651	0.9902	0.8079	0.5694	0.9846	0.7807
AODR-2	0.9967	0.7601	1.2609	0.9850	0.8001	0.5589	0.9716	0.7795
AODR-3	**0.9857***	**0.7450***	**1.2501**	**0.9785***	**0.7990***	**0.5497***	**0.9702**	**0.7686***

i_k item, then MAP can be computed as in (D'Addio et al., 2017):

$$\mathrm{MAP} = \frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{1}{m_{ij}}\sum_{k=1}^{m_{ij}} prec@R_{jk} \tag{20}$$

5.2.2. Evaluation

For evaluating the accuracy of our model in terms of both rating prediction and item ranking, we make comparison with the following methods: (1) (MF) (Koren, Bell, & Volinsky, 2009) which is widely used as a standard baseline for CF RSs. We used this baseline to assess the impact of the TF method for CF. (2) HFT (Hidden Factor and Hidden Topics) (McAuley & Leskovec, 2013), a state-of-the-art topic modeling-based RS which exploits user text review in addition to the user explicit ratings for RS. (3) RBLT (rating boosted latent topics) (Tan, Zhang, Liu, & Ma, 2016) model which specifically exploits user opinions rather than weighted preferences from the user textual review for RS.

We compare these baseline methods with our proposed model based on the three different settings of the MCNN (MOD-1, MOD-2, and MOD-3) as described in Section 4. In this way, we use AODR-1, AODR-2 and AODR-3 corresponding to the three settings of the MCNN respectively. To provide a fair comparison with the baseline methods, we use an open RS library, *MyMedialite*[1] for estimating the value of MF model. For the HTF and RBLT, we use 5 as the K value in our experiment.

5.3. Rating prediction performance

Table 5 shows the performance of the three various settings of our AODR approach w.r.t rating prediction in comparison with the baselines. The best performance is bolded in each case.

As can be seen from the Table 5, all the variants of our model achieve the best results on most of the datasets with the exception of AODR -1 which shows the lowest performance compared to the baselines. The statistical t-test shows that all the improvements are significant at 95% in most of the cases except in the Auto and Yelp datasets where our model shows slight improvement compared to the second best method in terms of RMSE. AODR-3 outperforms all the other variants including the baseline methods followed by the AODR -2. Some key observation can be drawn from this. First, AODR-1 variant performs wost simply because it uses only a single channel CNN model for the aspect extraction and that the word embedding channel is randomly initialized. AODR -3 performs better simply because of the multiple channels CNN model which particularly uses both word embeddings and POS tag embeddings for aspect extraction. This is consistent with the aspect extraction results reported in Section 4 and particularly indicates the usefulness of using both word embedding and POS tag for sequential labeling which consequently leads to the better performance of RS.

Regarding the baseline models, it can be seen from Table 5 that the MF method which solely relies on the user ratings for predictive performance records relatively lowest performances on all the datasets compared to other baseline models which utilize textual review in addition to the overall user ratings for improving the predictive performance. This reaffirms many previous findings which show the benefit of using text reviews for improving the predictive performance (D'Addio et al., 2017; Zheng et al., 2017). Compared to the HFT model, our model achieves significant gain with a substantial margin in terms of both RMSE and MAE. RBLT model, a boosted latent method and one of the powerful review-based rating prediction models, still our model manages to outperform it across all the datasets. This apparently indicates the impact of incorporating the weighted user opinions into the CF method for RS.

A key advantage of our model compared to the baselines is that our model takes into consideration the user's weighted opinions on different aspects of the item in addition to the robust aspect extraction method used which generates quality aspect terms used for building the RS. This clearly shows that a better aspect extraction method can generally lead to better performance of the recommendation system.

5.4. Top-N ranking performance

This section reports the performance of the proposed model in terms of ranking performance. The goal of the top-N ranking is to recommend a set of top-N ranked products that a user prefers most. In contrast to the rating prediction, it is more practical in a real-world situation because the system usually expects to suggest products that are likely most appealing to the user.

Table 6 shows the MAP and Prec@10 results of different versions of the proposed model in comparison with the baseline methods on all the datasets. It can be observed that in most cases, MOD-3 perform better on the IV and Auto datasets followed by MOD-2 while MOD-1 performs worst compared to other models. It can also be seen that MOD-3 consistently outperforms all the baselines including the MF models on all the datasets except on the MI and Yelp datasets where MF and HFT record the best result on MAP and Prec@10 respectively. One possible reason may be attributed to the size and relative sparseness of the datasets. While yelp review is relatively the largest and more sparse, the MI is relatively the smallest and less sparse datasets. This reaffirms many finding in the literature (Zheng et al., 2017).

Considering the results in rating prediction and top-N recommendation, it can be clearly observed that, the relative improvement of our model on Top-N recommendation compared to the baseline methods is not obvious as those of rating prediction. This is consistent with the previous findings (Wu et al., 2018) which showed that many sophisticated methods parform worse in top-N recommendation compared to the simple matrix factorization method. It reveals that the better performance in RMSE and MAE usually do not translate to the improvement of accuracy as for the TopN ranking performance.

[1] www.mymedialite.net.

Table 6
Results on the top N ranking. The best result is bolded in each case. All the improvements are significant at the 95% significance level.

Method	MI		Auto		IV		Yelp	
	Prec@10	MAP	Prec@10	MAP	Prec@10	MAP	Prec@10	MAP
MF	0.0810	0.0574	0.0820	0.0529	0.0895	0.0585	**0.0969**	0.0467
HFT	0.0957	**0.0638**	0.0914	0.0572	0.0908	0.0635	0.0952	0.0506
RBLT	0.0874	0.05891	0.0958	0.0602	0.0921	0.0612	0.0957	0.0513
MOD-1	0.0871	0.05815	0.0928	0.0598	0.0907	0.0601	0.0925	0.0498
MOD-2	0.0881	0.05904	0.0959	0.0631	0.0908	0.0652	0.0960	0.0524
MOD-3	**0.0989**	0.0635	**0.1075**	**0.0719**	**0.1282**	**0.0748**	0.0968	**0.0629**

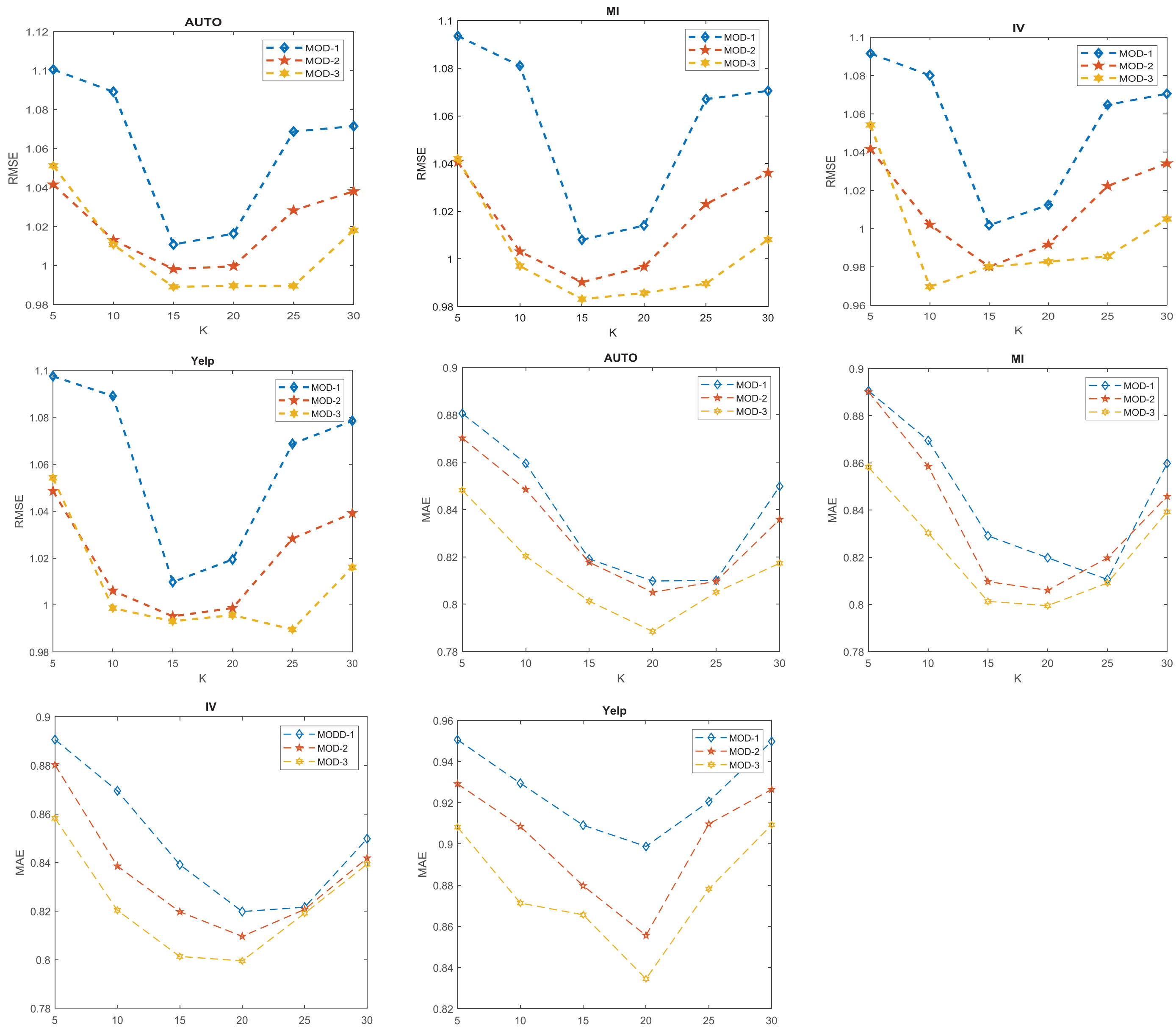

Fig. 4. RMSE and MAE performance of all the variants of AODR on different K values across all the datasets.

5.5. Sensitivity of parameter K

As explained earlier, we used the LDA method for clustering the aspect terms into K subgroups. To investigate the impact of different values of K on the performance of the model in terms of the rating prediction task, we conducted various experiments by applying different values of K. Fig. 4 shows the MAE and RMSE results on various values of K for all the datasets.

As can be observed from the figures, the best results are achieved when the K value is between 15 and 20 on most datasets. It is important to note that the increase in K value cannot always provide further improvement. This is because whenever the value

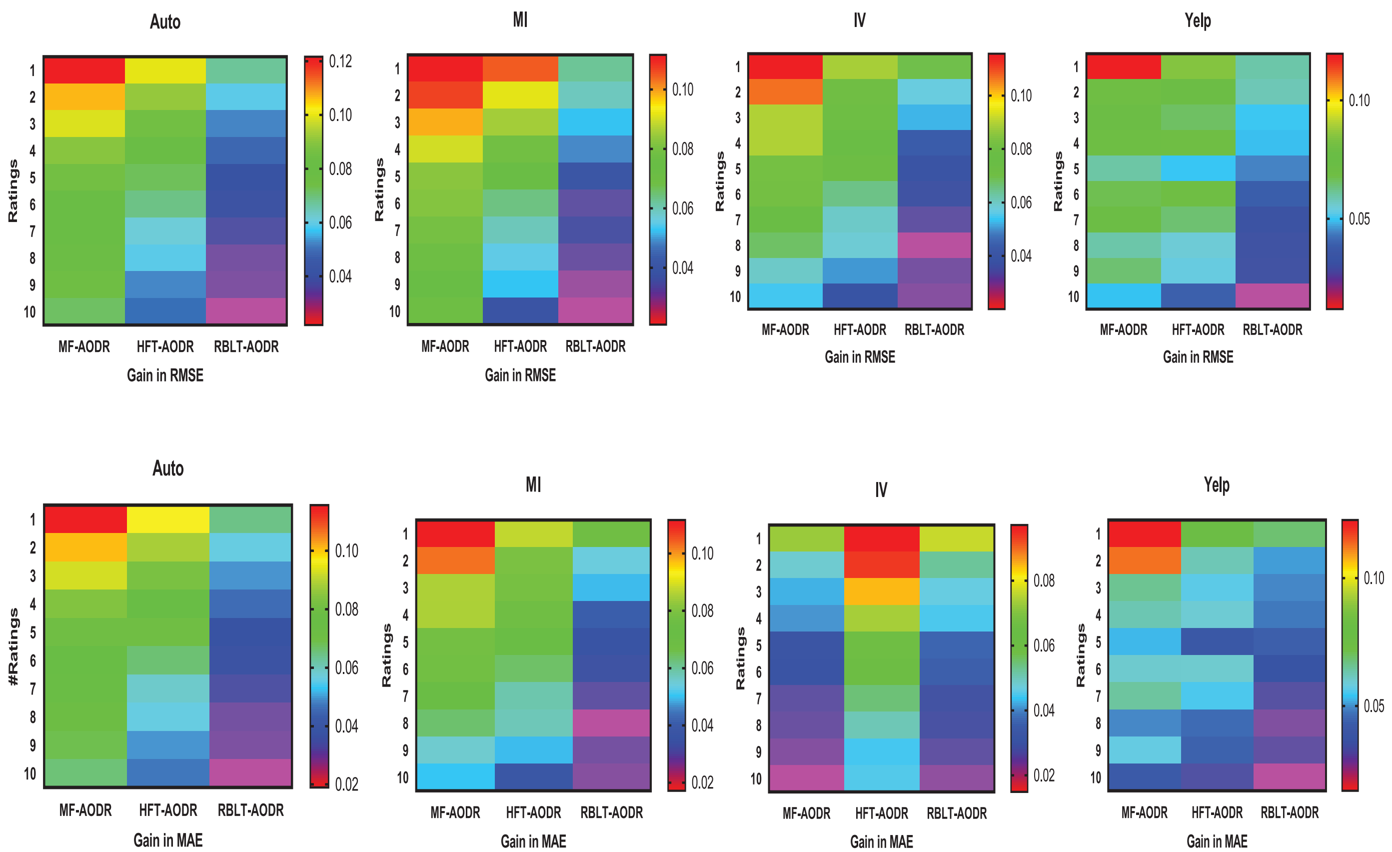

Fig. 5. RMSE and MAE results of the cold start performance on all the datasets.

Table 7
Top 5 Aspect clusters.

Tuner	software	Mouthpiece	mute	Microphone
Guitar	program	Cream	stylus	Mics
Capo	drivers	Fog	strings	Stand
Guitars	interface	Reeds	cartage	Wireless
Picks	windows	Harmonica	stick	Microphones

of K is larger enough the one-one relationship between the aspect and the latent factors eventually break. Therefore, in our experiments, we chose 15 as our parameter K. In Table 7 we show the clusters generated by LDA method when the number of clusters is set to 5.

5.6. Cold start problem

In a real-world application, RS datasets are typically sparse. This sparseness essentially leads to the cold start issue which is typically characterized by the inability of the system to provide an effective recommendation. Given limited rating data, usually, items/users are modeled only with biased terms. By exploiting user text reviews, cold start issue could be alleviated by our proposed method substantially because of the user/item side information contained in the review.

To examine the effectiveness of our AODR model in addressing the cold start condition, similar to Ling et al. (2014) and Tan et al. (2016), we compute the performance of a subset of users according to their ratings and take users with 1–10 ratings in the training sets and use their average values of the RMSE and MAE respectively. Fig. 5 shows the gains in RMSE and MAE values against the number of ratings by the users in the training sets. Here the gains in RMSE and MAE are given as the RMSE and MAE of the baseline methods minus that of our model respectively. A positive value indicates that our approach outperforms the baseline methods (i.e MF, HFT, and RBLT) in the cold start condition. Here we use the AODR -3 variant for the experiment due to its best performance compared to other variants.

6. Conclusion

In this paper, we proposed a recommendation model based on weighted aspect-based opinion mining using MCNN model. We first demonstrated how the MCNN model can be used for extracting aspect terms using different channels of the model and then described how the extracted aspects from the user text review can be used to generate aspect ratings using lexicon-based method. Further, we described how the aspect ratings are used to generate weighted opinions and then infer rating prediction by using TF technique. We evaluated the performance of the model in terms of both aspect extraction and item recommendation using real-world datasets. Experiment results demonstrated that our model achieved better results compared to the baseline methods on both aspect extraction and recommendation performances.

A key advantage of our proposed model over the baselines is that our model takes into consideration the user's weighted specific sentiments on different aspects of products and that a robust aspect extraction method is used to specifically generate reasonable aspect terms for CF. This clearly shows that a model with the best quality of the extracted aspects leads to better performance of the recommender system. As a future direction of the research, we think that using a stacked CNN network for the aspect extraction task and exploiting more additional side inform for rating predictive performance is worth exploring.

Declaration of Competing Interest

The authors have no any conflict of interest to declare.

Credit authorship contribution statement

Aminu Da'u: Data curation, Formal analysis, Investigation, Methodology, Resources, Writing - original draft, Writing - review & editing. **Naomie Salim:** Funding acquisition, Project administration, Resources, Supervision, Writing - review & editing. **Idris Rabiu:** Resources, Validation, Writing - review & editing. **Akram Osman:** Resources, Software, Visualization, Writing - review & editing.

Acknowledgments

This research work was supported by the University of Technology Malaysia under the Research University Grant (GUP) program, with the grant reference Number: Phy/2017/01348. Under the supervision of Prof. Dr. Naomie Salim.

References

Acar, E., Dunlavy, D. M., Kolda, T. G., & Mørup, M. (2011). Scalable tensor factorizations for incomplete data. *Chemometrics and Intelligent Laboratory Systems, 106*(1), 41–56. https://doi.org/10.1016/j.chemolab.2010.08.004.

Baccianella, S., Esuli, A., & Sebastiani, F. (2008). *SentiWordNet 3.0 : an enhanced lexical resource for sentiment analysis and opinion mining: 0* (pp. 2200–2204).

Bai, B., Fan, Y., Tan, W., & Zhang, J. (2017). DLTSR: A deep learning framework for recommendation of long-tail web services. *IEEE Transactions on Services Computing, 1374*(c), 1–13. https://doi.org/10.1109/TSC.2017.2681666.

Bao, Y., Hui, F., & Zhang, J. (2014). TopicMF: simultaneously exploiting ratings and reviews for recommendation. In *Aaai* (pp. 2–8). Retrieved from. http://www.ntu.edu.sg/home/zhangj/paper/aaai14-bao.pdf .

Bauman, K., Liu, B., & Tuzhilin, A. (2017). Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining – KDD '17* (pp. 717–725). https://doi.org/10.1145/3097983.3098170.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research, 3*(4–5), 993–1022.

Catherine, R., & Cohen, W. (2017). TransNets: Learning to transform for recommendation. In *RecSys'17* (pp. 288–296). https://doi.org/10.1145/3109859.3109878.

D'Addio, R. M., Domingues, M. A., & Manzato, M. G. (2017). Exploiting feature extraction techniques on users' reviews for movies recommendation. *Journal of the Brazilian Computer Society, 23*(1), 7. https://doi.org/10.1186/s13173-017-0057-8.

Da'u, A., & Salim, N. (2019). Sentiment-aware deep recommender system with neural attention networks. *IEEE Access, 7*, 45472–45484. https://doi.org/10.1109/access.2019.2907729.

Devi, D. V. N. V. N., Kumar, C. K., & Prasad, S. (2016). A feature based approach for sentiment analysis by using support vector machine. In *2016 IEEE 6th international conference on advanced computing (IACC)* (pp. 3–8). https://doi.org/10.1109/IACC.2016.11.

Diao, Q., Qiu, M., Wu, C.-Y., Smola, A. J., Jiang, J., & Wang, C. (2014). Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). In *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining – KDD '14* (pp. 193–202). https://doi.org/10.1145/2623330.2623758.

Guang, Q., & Bing, L. (2009). Opinion word expansion and target extraction through double propagation. *Computational Linguistics, 37*(1), 1–19. Retrieved from https://scholar.google.com.br/scholar?start=220&q=tudonotítulo:+Extraction+Emotion+OR+Feeling+OR+Sentiment+OR+Opinion+OR+Personality+OR+Subjectivity&hl=pt-BR&as_sdt=0,5&as_ylo=2005&as_yhi=2015#11 .

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the 2004 ACM SIGKDD international conference on knowledge discovery and data mining - KDD '04* (p. 168). https://doi.org/10.1145/1014052.1014073.

Irsoy, O., & Cardie, C. (2014). Opinion mining with deep recurrent neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 720–728). https://doi.org/10.3115/v1/D14-1080.

Jakob, N., & Ag, S. (2009). Beyond the stars exploiting free-text user reviews. *TSA09*, 57–64.

Jebbara, S., & Cimiano, P. (2016). Aspect-based relational sentiment analysis using a stacked neural network architecture. In *On artificial intelligence, 29 August-2* (pp. 1–9). https://doi.org/10.3233/978-1-61499-672-9-1123.

Jia, Y., Zhang, C., Lu, Q., & Wang, P. (2014). Users' brands preference based on SVD++ in recommender systems. In *Proceedings - 2014 IEEE workshop on advanced research and technology in industry applications, WARTIA 2014* (pp. 1175–1178). https://doi.org/10.1109/WARTIA.2014.6976489.

Jing, H., & Smola, A. J. (2017). Neural survival Recommender. In *WSDM 2017, ACM* (pp. 515–524). https://doi.org/10.1145/3018661.3018719.

Kim, D., Park, C., Oh, J., Lee, S., & Yu, H. (2016). Convolutional matrix factorization for document context-aware recommendation. In *Proceedings of the 10th ACM conference on recommender systems - RecSys '16* (pp. 233–240). https://doi.org/10.1145/2959100.2959165.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 emperical methods in natural language processing (EMNLP)* (pp. 23–31). https://doi.org/10.1145/1599272.1599278.

Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer, 42*(8), 30–37. https://doi.org/10.1109/MC.2009.263.

Lafferty, J., Mccallum, A., Pereira, F. C. N., & Pereira, F. (2001). Conditional random fields. In *Proceedings of the 18th international conference on machine learning 2001 (ICML 2001)* (pp. 282–289). Retrieved from. http://repository.upenn.edu/cis_papers%5Cn .

Li, F., Liu, N., Jin, H., Zhao, K., Yang, Q., & Zhu, X. (n.d.). Incorporating Reviewer and Product Information for Review Rating Prediction. 1820–1825.

Li, X., & She, J. (2017). Collaborative variational autoencoder for recommender systems. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining - KDD '17* https://doi.org/10.1145/3097983.3098077.

Ling, G., Lyu, M. R., & King, I. (2014). Ratings meet reviews, a combined approach to recommend. In *Proceedings of the 8th ACM conference on recommender systems - RecSys '14* (pp. 105–112). https://doi.org/10.1145/2645710.2645728.

Liu, P., Joty, S., & Meng, H. (2015). Fine-grained opinion mining with recurrent neural networks and word embeddings. In *Proceedings of the 2015 conference on empirical methods in natural language processing (EMNLP-2015)* (pp. 1433–1443).

Lu, Y., Smyth, B., Dong, R., & Smyth, B. (2018). Coevolutionary recommendation model: mutual learning between ratings and reviews. In *Proceedings of the 2018 world wide web conference on world wide web - WWW '18* (pp. 773–782). https://doi.org/10.1145/3178876.3186158.

Ma, X., Lei, X., Zhao, G., & Qian, X. (2017). Rating prediction by exploring user ' s preference and sentiment. *Multimedia Tools and Applications*, 1–20. https://doi.org/10.1007/s11042-017-4550-z.

McAuley, J., & Leskovec, J. (2013). Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on recommender systems* (pp. 165–172). https://doi.org/10.1145/2507157.2507163.

Mikolov, T., Yih, W., & Zweig, G. (2013). Linguistic regularities in continuous space word representations. In *Proceedings of NAACL-HLT* (pp. 746–751). Retrieved from. http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Linguistic+Regularities+in+Continuous+Space+Word+Representations#0%5Cnhttps://www.aclweb.org/anthology/N/N13/N13-1090.pdf .

Popescu, A., & Etzioni, O. (2005). Extracting product features and opinion from reviews. In *Human language technology and empirical methods in natural language processing* (pp. 339–346). https://doi.org/10.3115/1220575.1220618.

Poria, S., Cambria, E., & Gelbukh, A. (2016). Aspect extraction for opinion mining with a deep convolutional neural network. *Knowledge-Based Systems, 108*, 42–49. https://doi.org/10.1016/j.knosys.2016.06.009.

Qiu, L., Gao, S., Cheng, W., & Guo, J. (2016). Aspect-based latent factor model by integrating ratings and reviews for recommender system. *Knowledge-Based Systems, 110*, 233–243. https://doi.org/10.1016/j.knosys.2016.07.033.

Rockafellar, R. T. (1973). The multiplier method of Hestenes and Powell applied to convex programming. *Journal of Optimization Theory and Applications, 12*(6), 555–562. https://doi.org/10.1007/BF00934777.

Seo, S., Huang, J., Yang, H., & Liu, Y. (2017). Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In *Proceedings of the eleventh ACM conference on recommender systems - RecSys '17* (pp. 297–305). https://doi.org/10.1145/3109859.3109890.

Shu, L., Xu, H., & Liu, B. (2017). Lifelong learning CRF for supervised aspect extraction. In *148 Proceedings of the 55th annual meeting ofthe association for computational linguistics (Short Papers)* (pp. 148–154).

Snyder, B., & Barzilay, R. (2007). Multiple aspect ranking using the good grief algorithm. In *Proceedings of NAACL HLT* (pp. 300–307). https://doi.org/10.1.1.129.4132.

Tan, Y., Zhang, M., Liu, Y., & Ma, S. (2016). Rating-boosted latent topics: Understanding users and items with ratings and reviews. In IJCAI international joint conference on artificial intelligence, 2016–Janua (pp. 2640–2646).

Titov, I., & McDonald, R. (2008). A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of ACL08 HLT: 51* (pp. 308–316). https://doi.org/10.1039/b003067h.

Wang, F., & Chen, L. (2015). Review mining for estimating users' ratings and weights for product aspects. *Web Intelligence, 13*(3), 137–152. https://doi.org/10.3233/WEB-150317.

Wang, F., Chen. , L., Wang, F., & Chen, L. (2012). Recommending inexperienced products via learning from consumer reviews. In I 2012 IEEE/WIC/ACM international conferences OnWeb Intel- Ligence, *WI'12* (pp. 596–603). https://doi.org/10.1109/WI-IAT.2012.209.

Wang, H., Lu, Y., & Zhai, C. (2010). Latent aspect rating analysis on review text data : A rating regression approach. *KDD'10*, 1–10.

Wang, H., Lu, Y., & Zhai, C. (2011). Latent aspect rating analysis without aspect keyword supervision. In *Proceedings of the 17th ACM SIGKDD International conference on knowledge discovery and data mining - KDD '11: 138* (p. 618). https://doi.org/10.1145/2020408.2020505.

Wang, Y., Liu, Y., & Yu, X. (2012). Collaborative filtering with aspect-based opinion mining: A tensor factorization approach. In *Proceedings - IEEE international conference on data mining, ICDM* (pp. 1152–1157). https://doi.org/10.1109/ICDM.2012.76.

Wu, H., Zhang, Z., Yue, K., Zhang, B., He, J., & Sun, L. (2018). Dual-regularized matrix factorization with deep neural networks for recommender systems. *Knowledge-Based Systems, 145*, 1–14. https://doi.org/10.1016/j.knosys.2018.01.003.

Wu, Y., & Ester, M. (2015). FLAME: A probabilistic model combining aspect based opinion mining and collaborative filtering. *WSDM*, 199–208. https://doi.org/10.1145/2684822.2685291.

Xu, H., Liu, B., Shu, L., & Yu, P. S. (2018). Double embeddings and CNN-based sequence labeling for aspect extraction. In *Proceedings of the 56th annual meeting ofthe association for computational linguistics* (pp. 592–601). Retrieved from. http://arxiv.org/abs/1805.04601 .

Yang, C., Yu, X., Liu, Y., Nie, Y., & Wang, Y. (2016). Collaborative filtering with weighted opinion aspects. *Neurocomputing, 210*, 185–196. https://doi.org/10.1016/j.neucom.2015.12.136.

Yoon, J., & Kim, H. (2017). Multi-Channel Lexicon Integrated CNN-BiLSTM Models for Sentiment Analysis. In *The 2017 Conference on computational linguistics and speech processing* (pp. 244–253). Retrieved from.http://www.aclweb.org/anthology/O17-1023.

Zhang, Y., Lai, G., Zhang, M., Zhang, Y., Liu, Y., & Ma, S. (2014). Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th international ACM SIGIR conference on research & development in information retrieval - SIGIR '14* (pp. 83–92). https://doi.org/10.1145/2600428.2609579.

Zheng, L., Noroozi, V., & Yu, P. S. (2017). Joint deep modeling of users and items using reviews for recommendation. In *WSDM 2017 ACM* (pp. 1–10). https://doi.org/10.1145/3018661.3018665.