IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 24, 2019, accepted March 18, 2019, date of publication March 27, 2019, date of current version April 16, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2907729

Sentiment-Aware Deep Recommender System With Neural Attention Networks

AMINU DA'U[1,2] AND NAOMIE SALIM[1]

[1]School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, Johor Bahru 81310, Malaysia
[2]Hassan Usman Katsina Polytechnic, Katsina 820252, Nigeria

Corresponding author: Aminu Da'u (dauaminu@gmail.com)

This work was supported in part by the Ministry of Higher Education (MOHE), and in part by the Research Management Centre (RMC), Universiti Teknologi Malaysia (UTM), under the Research University Grant Py/2017/01348.

ABSTRACT With the advent of web technology, user-generated textual reviews are becoming increasingly accumulated on many e-commerce websites. These reviews contain not only the user comments on different aspects of the products but also the user sentiments associated with the aspects. Although these user sentiments serve as vital side information for improving the performance of recommender systems, most existing approaches ignore to fully exploit them in modeling the fine-grained user-item interaction for improving recommender system performance. Thus, this paper proposes a sentiment-aware deep recommender system with neural attention network (SDRA), which can capture both the aspects of products and the underlying user sentiments associated with the aspects for improving the recommendation system performance. Particularly, a semi-supervised topic model is designed to extract the aspects of the product and the associated sentiment lexicons from the user textual reviews, which are then incorporated into a long short term memory (LSTM) encoder via an interactive neural attention mechanism for better learning of the user and item sentiment-aware representation. Furthermore, a co-attention mechanism is introduced to better model the fine-grained user-item interaction for improving predictive performance. The extensive experiments on different datasets showed that our proposed **SDRA** model can achieve better performance over the baseline approaches.

INDEX TERMS Recommender system, LSTM, deep learning, user sentiment, neural attention mechanism, neural co-attention.

I. INTRODUCTION

Recommendation system aims to tackle the problem of information overload thereby assisting the customers to get their best choices from various alternative options. Basically, recommendation system can be achieved using various methods such as collaborative filtering (CF) and content-based [1]. CF methods have been shown as the most widely used techniques for recommender systems [2]. The basic idea of these approaches is that users who have similar consumption habits in the past tend to share similar items in the future. Most of the collaborative filtering methods are typically based on the matrix factorization (MF) method [3] which particularly uses latent factors to compute the unknown ratings of the user on an item. Although these methods have shown remarkable successes in many applications, yet, they generally suffer from several issues including the data sparseness.

The associate editor coordinating the review of this manuscript and approving it for publication was Jerry Chun-Wei Lin.

With the advancement of e-commerce websites such as Amazon and Yelp, nowadays, several approaches have been proposed [4]–[6] to utilize the free-text review for improving the performance of recommender systems. Many of these approaches use topic modelling [4], [7], [8] to automatically extract aspects and integrate them with the latent factor model for rating prediction. One of the major drawbacks of the existing topic model-based methods is their inability to capture the contextual information of words [9] which has been proven crucial for the effective performance of recommender systems.

With the recent success of representation learning, several approaches have been introduced to exploit deep learning for building recommender system [1], [10]–[14]. Most of these approaches exploit convolutional neural network (CNN) model [1], [12], [13] to jointly model both the user and item reviews for improving the predictive performance.

Although these approaches have been proven more effective than the state-of-the-art approaches for rating prediction,

however, all the above methods specifically ignore to fully consider the underlying user sentiment when modeling the user and item representation. In practical situations, users not only comment on different aspects of a specific product but also express their different sentiment polarities towards these aspects. For example, when a user wrote a review for a restaurant, he may write some sentences to express his displeasure with the location of the restaurant and its services and some sentences to express his satisfaction with the prices and its dishes. Unfortunately, these user sentiments are commonly neglected when modeling the user-item representation. However, the user sentiments serve as the key indicator of his preference and often express the extent of displeasure or satisfaction of the user toward an item. This paper proposes to fully consider this crucial information in the user/item representation learning for better predictive performance.

Another drawback of the abovementioned methods is that they specifically model latent feature vectors in a static and independent manner. In this way, user and item are projected into fixed low dimensional representations vectors in a shared space. It is intuitive that, not all words in the review are equally important and relevant to ratings of the user on a specific product. For instance, some word in the review may be explaining a plot of a film and such information may not correlate with the overall user preference. This is basically because each word in the review typically focuses on a distinct aspect of the user experience such as a price of the laptop, its performance or even the durability of its battery. Identifying the relevant semantic information by considering both the user and item review could be a new avenue for improving the performance of recommender system.

Considering the above motivations, this paper proposes a Sentiment-aware Deep Recommender System by incorporating topic model into a deep learning method to effectively capture domain-specific aspects of the product and the corresponding user sentiments using neural attention mechanism.

The model comprises four main components: (1) LSTM encoder which aims to capture the contextual and long dependencies information of words (2) semi-supervised topic model for extracting the domain-specific aspects and sentiment lexicons. (3) a co-attention mechanism to better learn the aspect importance for both the user and item review document and (4) prediction layer to estimate the user ratings on an item.

The major contributions of the proposed approach can be highlighted as follows:

- We propose a sentiment-aware deep recommender system which incorporates a semi-supervised topic model into a deep learning technique via an interactive attention mechanism for better performance of recommendation system.
- We design a semi-supervised topic model to extract domain-specific aspects and the associated user sentiment lexicons for effective sentiment aware user/item representation learning.
- We design a co-attention mechanism for better learning of the fine-grained user-item interaction.
- We perform a series of experiments on publicly accessible datasets to evaluate the performances of the proposed model against the baseline methods.

The remainder of the paper is arranged as follows:

Section II and Section III present the related work and an overview of the proposed approach respectively. Section IV describes the experimental study of the research and finally, section V concludes the paper.

II. RELATED WORK

This section reviews different approaches that are particularly relevant to our proposed approach. This includes topic-based recommendation systems, deep learning-based recommendation systems, and attention-based recommendation systems. In the following subsections, we review each of these categories.

A. TOPIC MODEL-BASED RECOMMENDER SYSTEM

In the past few years, several topic modelling approaches have been proposed to incorporate fine-grained information for more accurate rating prediction. Most of these approaches used latent topics from user review, others learn latent factors based on the factorization machine. For example, [4] and [5] used topic modelling to integrate the latent factor with latent topics based on a defined transformation process. References [15] and [16] aligned latent topics together with latent factors to generate a latent representation of users and items for rating modelling. Reference [6] applied topic modelling to learn features from user textual reviews using Gaussian method for rating prediction. Reference [17] introduced a joint model to jointly exploit aspect rating and user sentiment to alleviate the cold start problem.

All of the above methods are typically based on the traditional (Latent Dirichlet Allocation) LDA model [18] for the rating prediction, and that LDA models generally use bag-of-word method. As such they cannot effectively capture the contextual information of the word. In this paper, unlike the above methods, we design a novel semi-supervised topic model to learn the domain-specific aspects and sentiment-lexicons for better user/item representation learning.

B. DEEP LEARNING-BASED RECOMMENDER SYSTEMS

With the recent success of deep learning methods in various application such as computer vision and natural language processing [19]. Many approaches have been proposed to exploit deep learning techniques for recommender systems. These include Denoising auto-encoders [10], [20], convolutional neural network (CNN) [1], [9], [12], [13] and recurrent neural networks (RNN) [21]–[24]. Specifically, Wu *et al.* [25] and Sedhain *et al.* [26] exploited Denoising-Autoencoder for rating prediction. This approach intrinsically suffers from the data sparseness problem. To address this issue, Wang *et al.* [10] introduced collaborative deep learning (CDL)

model by integrating the probabilistic topic model with the collaborative filtering technique. CDL is an extended variant of the Collaborative Topic Regression (CTR) approach introduced in [7]. Due to their topic modeling affinity, these models ignore semantic contextualization of words

Owing to its remarkable success in image processing and pattern recognition, CNN model has been widely used for building recommender systems. Particularly, Zheng *et al.* [1] proposed Deep-CoNN to exploit two CNN networks to jointly model both the user and item reviews for improving rating prediction. Catherine and Cohen [13] proposed a model called Transnet as an extension of the Dee-pCoNN. The authors used an additional top layer for improving the rating prediction. A similar approach has been introduced recently by [27]. The authors proposed a scalable deep recommender system, using two separate CNN models.

RNN has been also widely exploited for recommender system in various applications such as movie recommendation [14], the next basket personalized recommendation and news recommendation [28]. particularly, Bansal *et al.* [22] introduced a multitasking learning approach based on the Gated Recurrent Unit (GRU) model to encode document for implicit user feedback. Gao *et al.* [29] introduced a dynamic RNN framework for modeling the dynamic interest of users in a unified framework. Hidasi *et al.* [24] exploited RNNs to develop a session-based recommender system.

Despite their state-of-the-art success, these models are limited, in that, they particularly derive user/item latent feature vectors in a static and independent manner and ignore the complex fine-grained user-item interaction. In this way, item and the user only interact at the top layer where the learned user/item representation is used for the overall rating prediction. As such it is difficult in these models to provide insights behind the user ratings on the items.

C. ATTENTIVE RECOMMENDER SYSTEMS

The main idea of the attention mechanism is intuitively similar to the visual attention in humans. Particularly, it equips a neural network to be capable of selecting the most important parts of the target input such as a specific word in a given review or a particular region in an image. This idea has been usefully applied to a number of applications such as computer vision [30], machine translation [31] and natural language processing [32]. More recently neural attention has been exploited for building recommender system [12], [21], [33]. For example, Bahdanau *et al.* [31] proposed an attention-based approach to accurately align the encoder-decoder framework for machine translation. With the achievement of self-attentive methods in machine translation, [34] and [35] utilized self-attention for modeling user behaviors. A multi-level attention mechanism was proposed by [36] for video/image recommendation. Seo *et al.* [12] proposed an attentive CNN network and use factorization machine at the top layer for the rating prediction. The authors designed two attention mechanisms, namely local and global attention to better model user and item representation.

TABLE 1. Notations

Notation	Definition
D	Corpus with a set of reviews
$d_{u,i}$	User, u review for item i
$r_{u,i}$	User u ratings for item i
Ω_k^s	Sentiment words distribution
Ω_m^a	Aspect words distribution
Φ_k^s	The probabilities of sentiment topic K
Φ_m^a	The probabilities of sentiment topic M
p^s, p^a	Head-tail probabilities of sentiment and aspect words
θ^s, θ^s	Sentiment topic, aspect topic
$N_{k,w}^s, N_{k,w}^a$	Number of occurrences of sentiment and aspect topic with word w in the document.
$\Phi_{k,w}^s$	Probability of word under sentiment topic
$\Phi_{m,w}^a$	Probability of word under aspect topic
$\mathbb{Q}_u$, $\mathbb{Q}_i$	User representation, Item representation
$\emptyset_u$ ϑ_i	Importance of aspect for user u, importance of aspect for item i

Our co-attention approach is closely related to the neural attention method used in [37], which is viewed as a form of pairwise neural attention. The authors apply the neural co-attention for visual questions and answers. Different from the above approaches, our proposed model exploits two different attention mechanisms namely, integrated neural attention to capture the most informative aspects and the associated user sentiment from the user text review, and neural co-attention network to model the fine-grained user-item interaction.

III. METHOD

This section describes the proposed SDRA (**S**entiment-aware **D**eep **R**ecommender system with neural **A**ttention) model which utilizes a semi-supervised topic model and LSTM encoder with neural attention mechanisms. We first specify the notations used in this paper in Table 1 and describe the problem settings for the research. The overview of the model architecture and the objective function to be optimized are then elaborated.

A. PROBLEM SETTINGS AND NOTATIONS

Let D be a corpus for a given set of items i written by a set of users u andeach review $d_{u,i} \epsilon D$ is accompanied by an overall rating $r_{u,i}$ which shows the overall satisfaction of the user on the item. Each item-user interaction can be denoted as a tuple $(u, i, r_{u,i}, d_{u,i})$.

The primary goal is to compute the unknown ratings $\hat{r}_{u,i}$ of a user u on an item i that has not beenobserved by the user. Table 1 shows the notations used in this paper.

B. OVERVIEW OF SDRA

The overall architecture of the proposed SDRA model is shown in Figure1. It comprises four (4) main components: (1) LSTM encoder to better learn the semantic and contextual

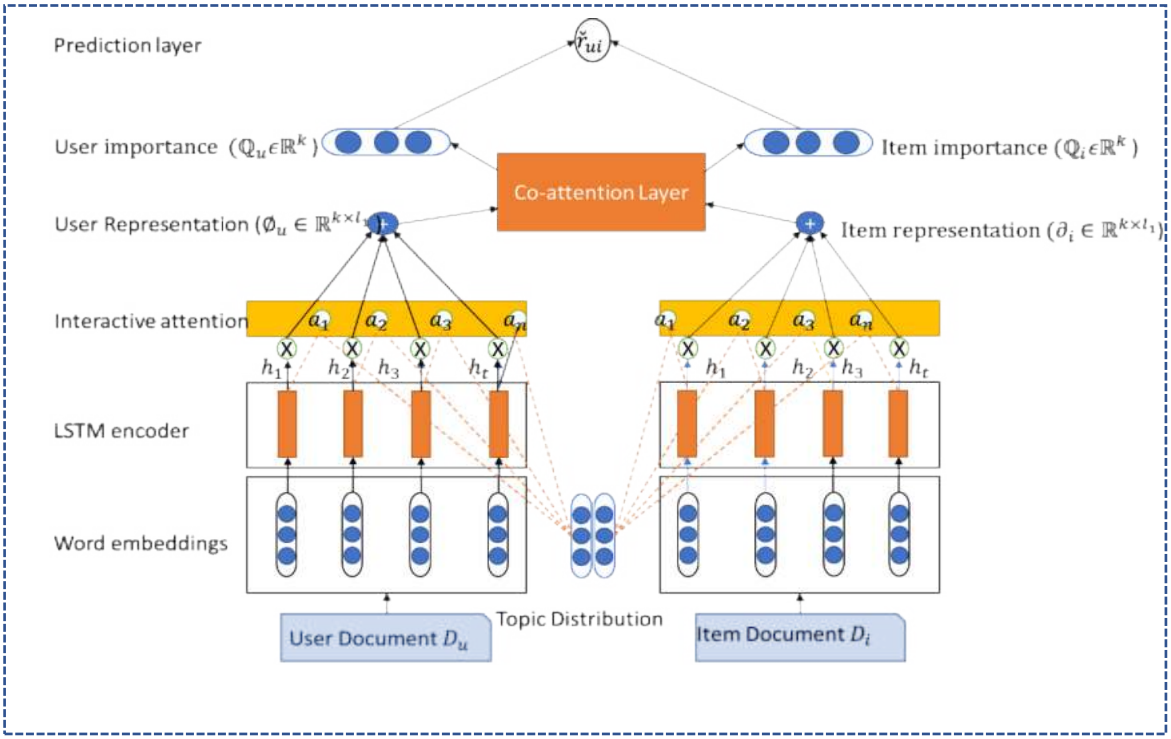

FIGURE 1. The Architecture of the proposed model.

information of words. (2) Semi-supervised topic modeling for extracting the domain-specific aspects and sentiment lexicons. (3) Co-attention network layer for estimating the user and item aspect importance. (4) Rating prediction layer for estimating the predicted ratings. Detail description of the model is presented in the following subsections.

C. EMBEDDING LAYER

The embedding layer takes a set of sequence words from document D and map them into n – dimensional matrix $x_i = R^n$. This representation technique is used to encode semantic and syntactic information carried by words. The embedding layer can be initialized using pretrained word vectors such as Glove or Word2vec. In this paper Word2vec [38] pretrained on a large corpus of Google news is used to initialize the embedding layer.

D. SEQUENCE ENCODING LAYER

The main purpose of the sequence encoding layer is to provide contextual annotation of the input sequence of words. Here an LSTM model is employed because it performs well on several applications and has been successfully used in sequence modeling [39]. Like many RNN variants, LSTM model consists of a chain repeating components in neural networks. Instead of assigning the repeating components as a simple structure, LSTM uses a cell state whose information can be updated. Formally, given an input x at time step t, the previouscell state c_t and current hidden state h_t can be updated as follows:

$$i_t = \partial(W_i x_t + U_i h_{t-1} + b_i) \quad (1)$$
$$f_t = \partial(W_f x_t + U_f h_{t-1} + b_f) \quad (2)$$
$$O_t = \partial(W_o x_t + U_o h_{t-1} + b_o) \quad (3)$$
$$c_t = i_t \odot \tilde{c}_t + f_t \odot c_{t-1} \quad (4)$$
$$\tilde{c}_t = \tanh((W_c x_t + U_c h_{t-1} + b_c) \quad (5)$$
$$h_t = O_t \odot \tanh(c_t) \quad (6)$$

where W and U represent weight matrices to be learned, i_t, f_t, $O_t, \tilde{c}_t$ is the input gate, forget gate, output gate and memory cel l respectively, ∂, and $\bigodot$ are sigmoid function and element wise multiplication respectively. The output of the LSTM is obtained as a sequence of hidden states $(\mathrm{h}_1, \mathrm{h}_2 \ldots . \mathrm{h}_\mathrm{t}) \in \mathbb{R}^\mathrm{d}$. Each annotation contains information about the whole review with focus on the a $i - th$ word surrounding. Where d is the size of hidden states for the LSTM encoder.

E. SEMI-SUPERVISED TOPIC MODEL

To better capture the domain-specific aspects and the associated user sentiments lexicons, a semi-supervised topic modeling is designed. Inspired by [40] we extend the vanilla LDA model [18] exploiting seed words for guiding the topic model design. The topics automatically extract and categorize sentiment words and aspect terms in the associated aspect and sentiment-lexicon categories.

we assumed that each review comprises two categories of topics: k Sentiment topics with two polarities (negative

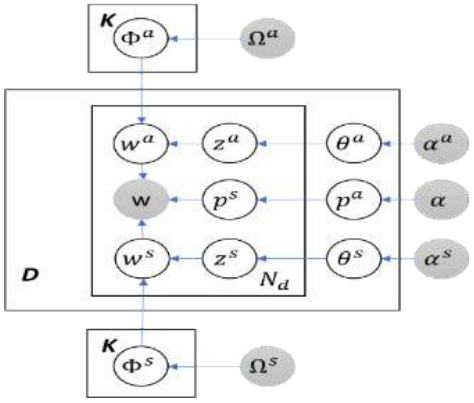

FIGURE 2. The graphical representation of the semi-supervised topic model. Shaded nodes denote observation and white nodes denote random variables. Large plates represent the part of the repeated part of the graphs.

and positive emotions) and m aspect topics. Each topic is related to a multinomial distribution over words. Assuming that the corpus vocabulary ccomprises $\mathcal{U}$ different words indexed by $1 \ldots .\mathcal{U}$. For each review, we get two topic distributions Φ^{s}, Φ^{a} representing the probabilities of sentiment topic K and aspect topic m. Let $\Phi^{\mathrm{s}}_{\mathrm{k,w}}$ and $\Phi^{\mathrm{a}}_{\mathrm{m,w}}$ denote the probabilities of word ω under sentiment topic k and aspect m respectively. For each sentiment topic K, its words distribution $\Phi^{\mathrm{s}}_{\mathrm{k}}$ is selected from a Dir $(\Omega^{\mathrm{s}}_{\mathrm{k}})$, where $\Omega^{\mathrm{s}}_{\mathrm{k}}$ is a $\mathcal{U}$- dimensional vector. The $\mathcal{U}$-dimensional vector can be computed as follows:

$$\Omega^{\mathrm{s}}_{\mathrm{k,w}} = \gamma_0 (1 - \beta_{\mathrm{w}}) + \gamma_1 \beta_{\mathrm{w}}, \quad \text{for } w \epsilon \{1, \ldots \mathcal{U}\} \tag{7}$$

where the scalers γ_0 and γ_1 are hyper parameters of the model. $\beta_{\mathrm{w}} = 1$ if the word w is a seed word in sentiment topic K, otherwise $\beta_{\mathrm{w}} = 0$. Ideally, a seed word from topic k is enforced by the biased prior $\Omega^{\mathrm{s}}_{\mathrm{k}}$. Conversely, aspect topic distribution $\Phi^{\mathrm{a}}_{\mathrm{m}} \backsim$ Dirichlet $(\Omega^{\mathrm{a}}_{\mathrm{m}})$ is constructed in a similar way. The generative method of the topic model can be highlighted in the following. The graphical representation of the model is shown in figure 2:

1) *For* each sentiment *topic*$K \epsilon \{0,1\}$:
   a) draw $\Phi^{\mathrm{s}}_{\mathrm{k}} \backsim$ Dirichlet $(\Omega^{\mathrm{s}}_{\mathrm{k}})$.
2) For each aspect topic $m \in \{1, ..\mathrm{m}\}$:
   a) draw$\Phi^{\mathrm{a}}_{\mathrm{m}}$ Dirichlet $(\Omega^{\mathrm{a}}_{\mathrm{m}})$.
      i) For each document:
         A) draw $(\mathrm{p}^{\mathrm{s}}, \mathrm{p}^{\mathrm{a}}) \sim$ Dir (α)
         B) draw $\theta^{\mathrm{s}}, \sim$ Dirichlet (α^{s}), $\theta^{\mathrm{a}}, \sim$ Dir (α^{a})
         C) for each word in the document:
            i. Draw a class indicator $\mathrm{S} \sim Bern(\mathrm{p_s})$
            ii. If S = "sentiment topic"
            A. Draw $\mathrm{z}^{\mathrm{s}} \backsim$ Multinomial (θ^{s})
            B. Draw w$\sim$ Multinomial $(\Phi^{\mathrm{s}}_{\mathrm{z}^{\mathrm{s}}})$
            C. Emit word w
            iii. Otherwise
   A. Draw $\mathrm{z}^{\mathrm{a}} \backsim$ Multinomial (θ^{a})
   B. Draw w$\sim$ Multinomial $(\Phi^{\mathrm{a}}_{\mathrm{z}^{\mathrm{a}}})$
   C. Emit word w.

To estimate the unknown parameters, Gibbs sampling algorithm [41] is used in this paper. Due to the space limit, interested reader can refer to the Gib sampling methods in [41]. To obtain the words distribution of sentiment and aspect topics. Formally Φ^{s} and Φ^{a} can be obtained as follows:

$$\Phi^{\mathrm{s}}_{\mathrm{k,w}} = \frac{\Omega^{\mathrm{s}}_{\mathrm{k,w}} + \mathrm{N}^{\mathrm{s}}_{\mathrm{k,w}}}{\sum_{\mathrm{w}=1}^{\mathrm{U}} (\Omega^{\mathrm{s}}_{\mathrm{k},W'} + \mathrm{N}^{\mathrm{s}}_{\mathrm{k},W'})},$$
$$\text{and } \Phi^{\mathrm{a}}_{\mathrm{m,w}} = \frac{\Omega^{\mathrm{a}}_{\mathrm{k,w}} + \mathrm{N}^{\mathrm{a}}_{\mathrm{k,w}}}{\sum_{\mathrm{w}=1}^{\mathrm{U}} (\Omega^{\mathrm{a}}_{\mathrm{k},W'} + \mathrm{N}^{\mathrm{a}}_{\mathrm{k},W'})} \tag{8}$$

where $\mathrm{N}^{\mathrm{s}}_{\mathrm{k,w}}$ and $\mathrm{N}^{\mathrm{a}}_{\mathrm{k,w}}$ represent the occurrences of sentiment and aspect topic with word w in the document. The reader may refer to [41] for detail of the sampling procedures.

Next, the $\mathcal{U}$-dimensional word distribution of sentiment topic i and aspect topic j are transformed to low dimensional sentiment embeddings e^{s} and aspect embedding e^{a} via a fully connected layer so as to obtain the same dimension with the LSTM hidden state. Thus, we have:

$$\mathrm{e}^{\mathrm{s}}_{\mathrm{i}} = \tanh(\mathrm{W}^{\mathrm{s}} \Phi^{\mathrm{s}}), \quad \mathrm{e}^{\mathrm{a}}_{\mathrm{j}} = \tanh(\mathrm{W}^{\mathrm{a}} \Phi^{\mathrm{a}}) \tag{9}$$

where the matrices $W^s \epsilon \mathbb{R}^{U \times d}$ and $W^a \epsilon \mathbb{R}^{U \times d}$ are trainable parameters, d is the dimension of the hidden states of the LSTM and tanh is a non–linear function.

F. INTERACTIVE TOPICAL ATTENTION

The interactive attention mechanism is basically aimed to capture the most relevant information from the input text review for learning the aspect/sentiment aware document representation. Formally, given the hidden states $[\mathrm{h}_1, \mathrm{h}_2..\mathrm{h}_t]$ from the LSTM encoder as well as the sentiment embeddings $[\mathrm{e}^{\mathrm{s}}_1, \mathrm{e}^{\mathrm{s}}_2 \ldots\ldots \mathrm{e}^{\mathrm{s}}_n]$and aspect embeddings $[\mathrm{e}^{\mathrm{a}}_1, \mathrm{e}^{a}_2 \ldots\ldots \mathrm{e}^{a}_n]$ learned from the topic modelling, the interactive attention network generates the sentiment lexicon and aspect specific attention weight respectively as follows:

$$\alpha^s_t = \frac{exp(\gamma(h_t, e^s))}{\sum_{t=1}^{T} exp(\gamma(h_t, e^s))}, \tag{10}$$
$$\alpha^a_t = \frac{exp(\gamma(h_t, e^s))}{\sum_{t=1}^{T} exp(\gamma(h_t, e^s))} \tag{11}$$

where $\alpha^{\mathrm{s}}_{\mathrm{t}}$ and $\alpha^{\mathrm{a}}_{\mathrm{t}}$ are the attention scores indicating how likely the associated hidden state $\mathrm{h_t}$ serves as an aspect and sentiment indicator respectively. Where γ is the score function that indicates the importance of the hidden state $\mathrm{h_t}$ in the context. The γ score can be defined as:

$$\gamma\left([h_t; e^s]\right) = U_s^T \tanh(W^s[h_t; e^s] + b_s) \tag{12}$$
$$\gamma\left([h_t; e^a]\right) = U_a^T \tanh(W^a[h_t; e^a] + b_a) \tag{13}$$

where U_s^T, U_a^T, W^s and W^a are the projection parameters to be learned, and b_s and b_a are the biases. After computing the interactive attention vectors of the sentiment and aspect words, the sentiment aware and aspect specific representation can be obtained as follows:

$$V^s_u = \sum_{t=1}^{k} \alpha^s_t . h_t, \tag{14}$$

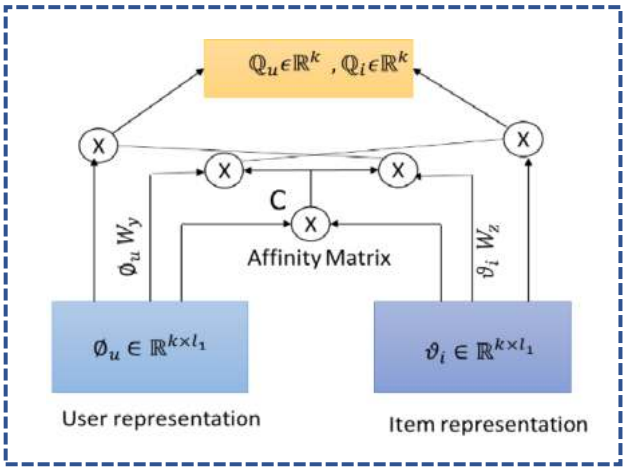

FIGURE 3. Neural Co-attention Mechanism with the affinity matrix and the user-item interaction layer.

$$V_u^a = \sum_{t=1}^{N} \alpha_t^a . h_t \tag{15}$$

To finally obtain the Aspect/sentiment aware of the user review document representation, the lexicon, and aspect-based representation can be concatenated as follows:

$$\emptyset_u = \tanh\left(\rho W_s V_u^s\right) + (1-\rho) W_a V_u^a) \tag{16}$$

where W_s *and* W_a are projection parameters and $\rho \in [0, 1]$ is used to control the effect of the sentiment and aspect aware representations. Conversely the Item representation ∂_i can be obtained in similar way.

representations $\{\emptyset_u, \partial_i\}$, can be obtained and then used for the rating prediction task. In this way, the aspect/sentiment aware of the user and item

G. CO-ATTENTION NETWORK LAYER

Intuitively, user preference toward aspects may vary depending on the item being considered. Therefore, rather than having static user/item importance, our aim is to model the dynamic fine-grained user-item interaction at the word level by computing the aspect importance for each user-item pair. Thus, inspired by the work of [37], we introduce a co-attention mechanism to capture the relative importance between different aspects for both user and item to better learn interactive user-item representation.

Specifically, the user representation is used as the context to learn the item aspect importance, and likewise, the item representation is used as the context to learn the user aspect importance. The output of this operation would be a K-dimens ionalvector showing the importance of each aspect with respect to the associated user sentiment for the item and a corresponding vector for the user. The overview structure of the co-attention mechanism is illustrated in figure 3.

To achieve the dynamic fine-grained user-item interaction, there is a need to compute the similarity between the target user and item. Specifically, given the user representation $\emptyset_u \in \mathbb{R}^{k \times l_1}$ and the item representation $\vartheta_i \in \mathbb{R}^{k \times l_1}$, the affinity matrix $C \in \mathbb{R}^{k \times k}$ is computed as:

$$C = tanh(\vartheta_i^T W_c \emptyset_u) \tag{17}$$

where $W_c \in \mathbb{R}^{l_1 \times l_1}$, is the weight matrix to be learned.

Following [37], we apply matrix $C \in \mathbb{R}^{k \times k}$ as features to compute the user and item importance as follows:

$$H_u = tanh(\emptyset_u W_y + C^T(\vartheta_i W_z)), \mathbb{Q}_u = softmax\ (H_u V_y) \tag{18}$$

$$H_i = tanh(\vartheta_i W_z + C(\vartheta_u W_y)), \mathbb{Q}_i = softmax\ (H_i V_z) \tag{19}$$

where W_y, $W_z \epsilon \mathbb{R}^{l_1 \times l_2}$ *and* V_y, $V_z \in \mathbb{R}^{l_2}$ are the weight parameters to be learned. $\mathbb{Q}_u \epsilon \mathbb{R}^k$ and $\mathbb{Q}_i \epsilon \mathbb{R}^k$ are the user and item importance respectively.

In this way, the user and item importance, $(\mathbb{Q}_u, \mathbb{Q}_i)$ can be obtained and used in the rating prediction layer.

H. PREDICTION LAYER

The rating prediction layer is where the actual rating prediction task for the recommendation process occurs. The prediction layer typically accepts as an input the user/item representations ($\emptyset_u, \vartheta_i$), and the aspect impotence $(\mathbb{Q}_u, \mathbb{Q}_i)$ and pass them into factorization machine. Factorization machine accepts real-valued feature vectors and treats the pairwise interaction. Thus, the overall ratings can be inferred as:

$$\tilde{r}_{u,i} = \sum_{a \epsilon A} (\mathbb{Q}_u \cdot \mathbb{Q}_i \cdot (\emptyset_u (\vartheta_i)^T) + b_u + b_i + \mu \tag{20}$$

where b_u, b_i and μ are the user, item and, global bias respectively. The back propagation method can be used to learn the model parameters and the mean squared error function is used as the loss function:

$$\mathcal{L} = \sum_{(r_{u,i}) \epsilon D} (\tilde{r}_{u,i} - r_{u,i})^2 \tag{21}$$

where $r_{u,i}$ is the observed rating in the review D and $\tilde{r}_{u,i}$ is the unobserved rating.

IV. EXPERIMENTAL STUDY

This section presents different experiments to assess the effectiveness of the proposed SDRA model. The experimental study is aimed to answer the following questions:

RQ1. Can the SDRA model perform better than the baseline methods?

RQ2. Is the SDRA model sensitive to hyperparameters such as dropout, latent factors, and embedding dimensions?

RQ3. How does the proposed model perform in terms of the cold star settings?

RQ4. What are the impacts of different components in the model architecture?

A. DATASETS

To evaluate the SDRA performances, two categories of datasets are used: Amazon review datasets [4] and Yelp 2017 challenge datasets. The characteristics of the two datasets are described as follows:

Amazon Product Review Theses datasets are organized into 24 individual product categories and have been widely

TABLE 2. Statistics of the datasets.

Datasets	User	Item	Review
Musical I.	67005	14115	84408
Pet	160494	17490	216612
Automotive	133254	47539	188387
Health	311634	39275	421627
Gourmet	112539	23367	153733
Yelp 2017	169257	63300	1659678

used for rating prediction tasks by many researchers in the previous works [1], [6], [12], [13]. For the original datasets are too large, for our experiment, the 5 categories of the datasets are particularly used. Specifically, we use the 5-core version where each user or item has at least 5 interactions.

Yelp 2017 Datasets Challenge- Yelp is an online review platform which contains a review of local businesses in 12 metropolitan areas across 4 countries. This dataset has been previously used in many research works [1], [42], [43]. Since the original dataset is very large and sparse, it was pre-processed to obtain a version with at least five ratings for each user.

All the datasets are pre-processed as follows: all the infrequent terms and duplicates are removed to ensure a 5-core version. All the stop words and the non-vocabulary words from the review documents are filtered. Following [1], [13], we randomly split each dataset into training, validation and testing sets using a ratio of 80%, 10% and 10% for the training, validation, and testing respectively. The statistics of the datasets is given in table 2.

B. BASELINES

For a fair comparison, three classes of different baseline models are used in this paper: a purely rating based method (MF), Topic model-based method (HFT) and deep learning based methods (DeepCoNN, D-att, Transnet). Each of the baseline models is explained as follows:

- MF [3]: This is the well-known standard baseline for collaborative filtering method. It typically uses the inner product to represent the user and item for rating prediction.
- HFT [4]: This is one of the most successful topic modeling based approach which simultaneously models ratings and reviews with latent topics.
- D-attn [12]: This model typically uses local and global attention to select local and global information of the words. It specifically uses the inner productfor inferringrating prediction.
- Deep-CoNN [1]: This is the state-of-the-art deep learning based rating prediction model which uses two parallel CNN model to learn user and item representation. It uses matrix factorization at the shared layer for rating prediction.
- Transnet [13]: This is an extended version of the Deep-CoNN model which additionally uses transform layers for ratingprediction.

C. EXPERIMENTAL SETTINGS

The proposed model and the baselines are implemented in python using TensorFlow library. Adam optimizer with the learning rate of 0.001 is used for the optimization. All the models are trained until convergence. For a fair comparison, we use the open recommender system library, MyMedialite to estimate the value of MF model. For the rest of the baselines, we use their source codes as provided by the authors accordingly. Specifically, for the HTF, we use item topic at $K = 5$. For the rest of the baselines models (i.e., Deep-CoNN, Trasnets, and D-Atten), we use the parameters as reported in their respective papers. All the parameters were fine-tuned to obtain the optimal performance and the best performance is then reported based on the testing set. We selected LSTM with 100 units, batch size 32 and dropout 0.5 as regularization. We use 300-dimensional word embeddings trained on Google news corpus. For the topic model settings, we set 4 as the number of the aspect topic. Other parameters used in the topic modeling include: $\alpha^s = \alpha^a = 0.25, \gamma_0 = 0.15$, and $\gamma_1 = 0.85$. We use the list of seed words similar to the ones used in [44]. Specifically, the seed words list comprised of seven positive words (i.e., *excellent, good, nice, positive, fortunate,correct*, and *superior*) and seven negative words (i.e.,*nasty, bad, poor, negative, unfortunate, wrong*, and *inferior*). The fourteen words are chosen for their lack of sensitivity to text.

D. EVALUATION METRICS

To evaluate the performance of our model, following the works of [1], [12], [13], we adopt mean squared error, MSE metric which estimates the average squared error between predicted ratings and the actual ratings. Formally, MSE can be given as the following equation:

$$\text{MSE} = \frac{1}{N}\sum_{n=1}^{N}(r_n - \check{r}_n)^2 \tag{22}$$

where N, r_n *and* $\check{r}_n$ is the actual ratings, the predicted ratings and the total number of the testing set respectively. Lower value of MSE indicates better performance of the model.

E. RESULTS AND DISCUSSION

The performance of the SDRA model and the baselines are recorded in terms of MSE in Table 3 on 5 sub-datasets from Amazon, and yelp 2017 challenge datasets. It can be observed from the results that our proposed model significantly outperformed the state-of-the-art baseline methods and the improvement is statistical significant at $p < 0.05$. This justifies the effective performance of our proposed model.

From table 4, it can be seen that the performance of MF which uses only ratings is relatively lower compared to the rest of the baseline models. The remaining models (HFT, Deep-CoNN, D-Att, and Transnet) which perform better than

TABLE 3. Performance comparison on 6 benchmark datasets. The best results are highlighted in boldface.

Datasets	MF	HFT	DeepCoNN	Transnet	D-Attn	SDRA
Musical I.	6.512	1.450	1.472	1.131	1.012	**0.915**
Pet	1.724	1.702	1.480	1.345	1.334	**1.310**
Automotive	5.010	1.884	1.125	0.956	0.901	**0.850**
Health	1.786	1710	1.254	1.243	1.270	**1.235**
Gourmet	1.521	1.512	1.201	1.126	1.141	**1.121**
Yelp 2017	1.712	1.601	1.342	1.359	1.425	**1.315**

TABLE 4. Attention visualization. (a) Yelp Item Document Item Document. (b) Musical I User Document User Document.

(a)

Item Document	Item Document
Some parts of the dinner were bad. while the other parts were just average, just nothing special ... I decided to take my mother out to dinner who was visiting all the way from America based on the great things. The night started off in a awful way when our waiter was very rude to us. We asked for a few minutes when our waiter asked us if we wanted still, sparkling or tap water and his reply in a indeed condescending tone was" right	... unsatisfactory food and a substandard service. I will not be coming back anytime .... If I was the manager, I would sack the waiters and promote the busboy to waiter as he was great tonight and he was the only reason I gave a 10% tip.. It is unfair for him to suffer because of the waiters. .... I probably should have bought something a bit more flexible and less rugged since I constantly coil uncoil it for washing car but that's my fault not a product fault.

b.

User Document	User Document
I frequently use this to feed an amplifier from various instruments. I use this cable to patch a preamp to an amplifier. It's good..... ..... having many styles of ends gives you better options when using a system together. it is fun and so simple to operate! I stopped messign with the reverbs in my amplifier and now only use this for a much better sound.	I frequently use this to feed an amplifier from various instruments. I use this cable to patch a preamp to an amplifier. It's good..... ..... having many styles of ends gives you better options when using a system together. it is fun and so simple to operate! I stopped messign with the reverbs in my amplifier and now only use this for a much better sound.

the MF model, all utilize user review text as the side information for rating prediction. This is not surprising, as the textual review is complementary to the overall ratings and it can be utilized for improving the representation of latent factors. This clearly translates the important of using the textual review as the auxiliary source of information for building recommender models. Specifically, the HFT which typically uses topic modeling with the bag-of-words method completely ignores the contextual information of words. As such the model records lower performance compared to the deep learning-based models

Further, it can be observed that the deep learning-based models usually perform better than the classical approaches, including HFT which also utilizes reviews for user/item modeling. This is consistent with previous findings [1], [12] which indicated that deep learning-based models such as CNN perform better than the topic modeling approaches such as LDA in text processing.

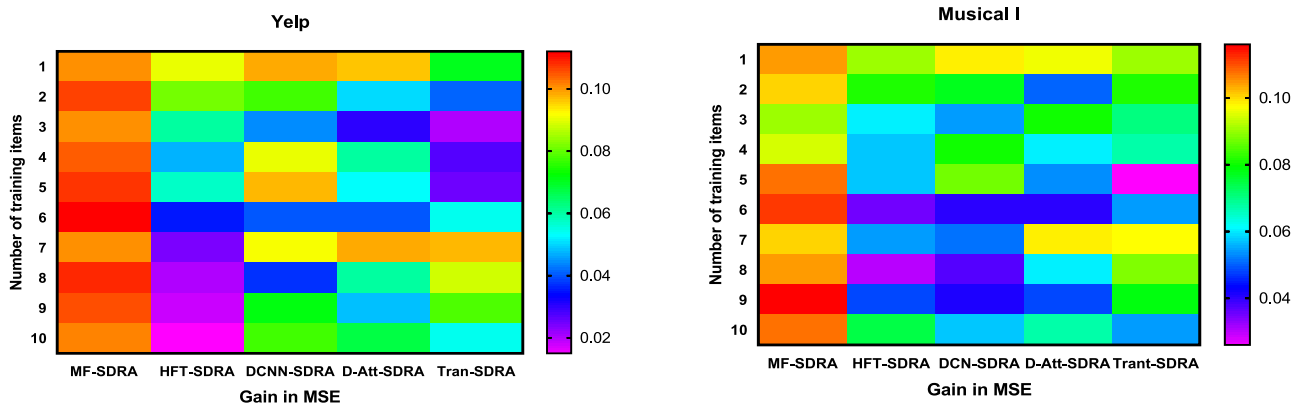

FIGURE 4. Gain in MSE for user with limited training data on two individual datasets.

Regarding the relative performance of the deep learning models, it can be observed that transnet always outperforms Deep-CoNN in most of the datasets, this reaffirms the claims of [13]. However, the relative ranking of D-Att and Transnet changes over different datasets. This may be as a result of using the textual reviews as an additional data source in Transnet while D-Att does not make use of this information. Regarding our proposed model, it can be observed from Table 3 that our model outperformed all the baselines including the Deep-CoNN, Transnet and D-att which are also the deep learning-based approaches with significant improvements across all the datasets.

Even though the text review is very important for improving the predictive performance, however, the performance largely depends on how the textual information is utilized. In our model, we utilized an interactive and co-attention mechanisms. These allow the approach to better model the fine-grained user-item interaction which lead to an improved performance according to results.

F. COLD START PROBLEM

Recommender system datasets are inherently sparse in real-world situations [42]. This sparseness leads to the issue of the cold start problem which is one of the major challenges of recommender systems. Given limited ratings, generally, items and users are modeled only with the biased terms. Therefore, collaborative filtering is often facing difficulties to effectively recommend due to the cold start problem. By integration of user review in the user and item latent learning, the issue of the cold start could be addressed by our model to a great extent, since rich information about the user preference on the aspects are contained in the review. Thus, to show the effectiveness of our model in alleviating the cold start problem, we report the performance of a subset of users based on their rating popularity. To this end, we select users with 1 to 10 ratings in the training sets and take the average values of the MSE for those users in our experiment. Figure 4(a) and (b) show the gain in MSE values against the number of ratings by the users in the training sets. Here the gain in MSE is given as the MSE of the baseline minus that of our model. A positive value means that our model achieves better performance of prediction in the cold start condition. Here due to the space limit, we only experiment on two sub datasets: Yelp and Musical Instrument

G. PARAMETER SENSITIVITY

For further analysis, we examine how different parameter settings may impact the performance of the proposed SDRA model. Particularly, we examine the model sensitivity to the number of factors l_1 *and* l_2, impacts of the dropout rates, word embeddings dimension and length of the document accordingly.

1) NUMBER OF LATENT FACTORS

We examine the performance of the SDRA model by varying the different number of factors used for l_1 *and* l_2 The 3D figure 5 shows the validation results of the model by adjusting both l_1 from 10 to 50 *and* l_2 from 5 to 25, across various datasets. It should be noted that l_1 represents the number of latent factors for the user and item representation and $l_2$determines the size of the hidden layers for the user importance based on the affinity matrix.

From figures 5(a) to (f), several observations can be made. Firstly, it can be observed that a large number of latent factors is not necessarily required for encoding the user/item representation to improve the modal performance and that, the best results are reached when l_2 is around 20 to 30 in most of the datasets. However, it can be observed that the model records poor performance when the latent factor is insufficiently used for the user and item representation. it can also be observed that, there is no significant impact on the aspect importance on the overall model improvement w.r.t number of latent factors used. Therefore, our choice of 20 and 10 for the l_1 and l_2 respectively is ideal for our model.

2) IMPACT OF DROPOUT

Dropout has been shown to be an effective method in addressing the issue of overfitting in the neural network model with

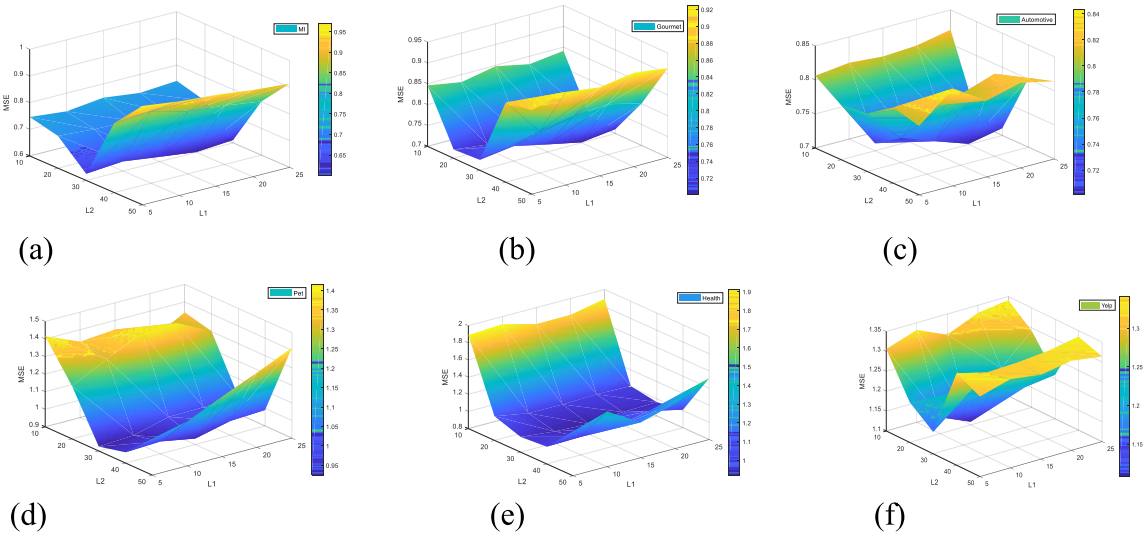

FIGURE 5. Performance on a varying value of L1 and L2 factors.

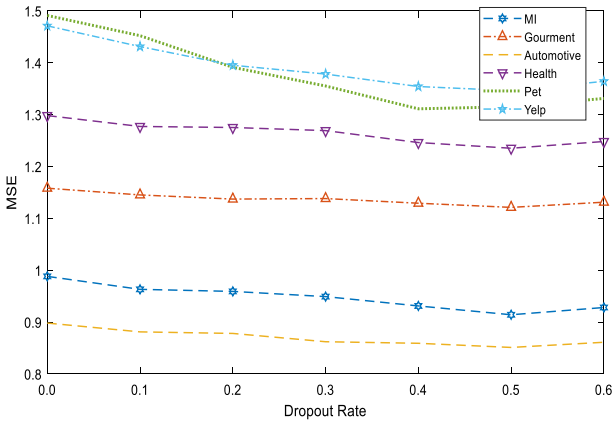

FIGURE 6. Dropout rate.

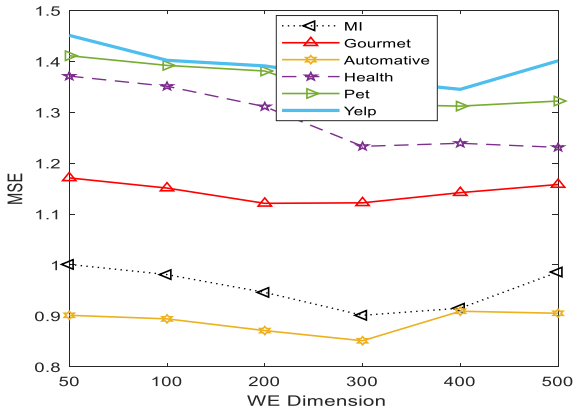

FIGURE 7. Word embedding dimension.

less training sample size [45]. Therefore, to examine the impacts of dropout, we vary the dropout with different values. Figure 6 shows that the dropout is very important in reducing the prediction error compared to the case in which drop out is not used.

It can be observed from figure 6 that, there is a significant improvement of the model across all the datasets. However, the gain of MSE varies across different datasets. The model records the best results between 0.4 and 0.5 in most of the datasets while it records the worst performance when the dropout is not used. It can also be observed that the dropout shows more impacts in relatively smaller datasets compared to the large-scale datasets. This is consistent with the previous observations [9]and reaffirms that dropout is more important for small-scale datasets. Thus, as can be seen in figure 5, an alternative value for the model is between 0.4 and 0.5.

3) EMBEDDING DIMENSION

To evaluate the sensitivity of the model towards word embeddings dimensions, different word embedding dimensions were used {50, 100, 200, 300, 400}. Figure 7 shows the results on the various datasets. One can observe that our model improves consistently across a wide range of values of word embedding dimensions. It can be seen that even with much smaller value of 50, the model shows good prediction performance in most datasets. The results indicate the highest performance at around 300 dimensions in most of the datasets and relatively remains stable above 400. This particularly implies the sensitivity of the model towards the dimension of word embeddings. This indicates that further use of larger values does not show significant improvement. Thus, we choose 300 as the word embedding dimension in our experiment.

4) IMPACT OF LENGTH OF THE DOCUMENT

To assess the impact of the length of document information on the model, we adjust the length of the document in different ways and evaluate the model on the Musical Instrument and Yelp datasets. It can be observed from figure 8 that the model performs better when the maximum length of the document is 200 and then becomes steady. This indicates that additional

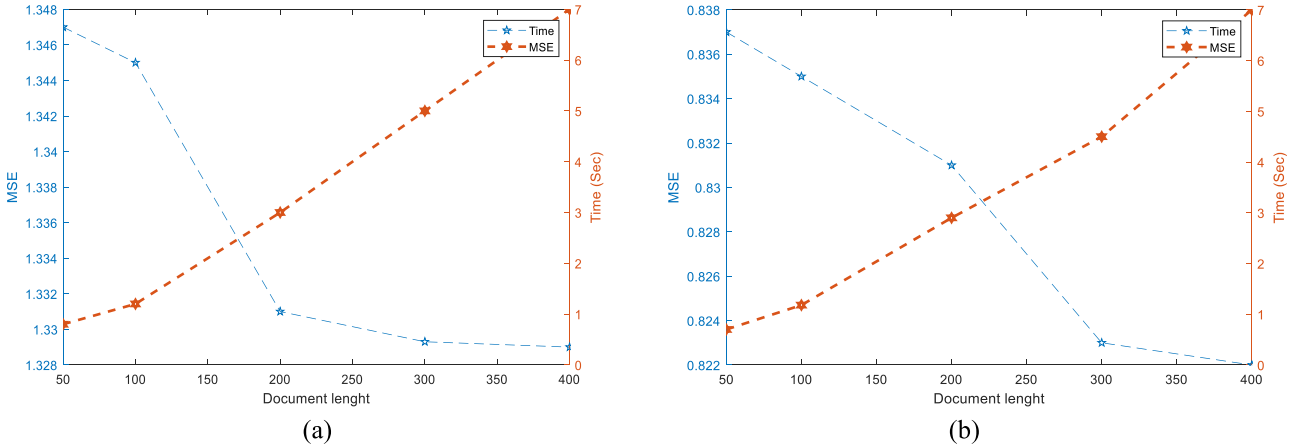

FIGURE 8. Document length on Musical I and Yelp datasets. (a) Musical I. (b) Yelp.

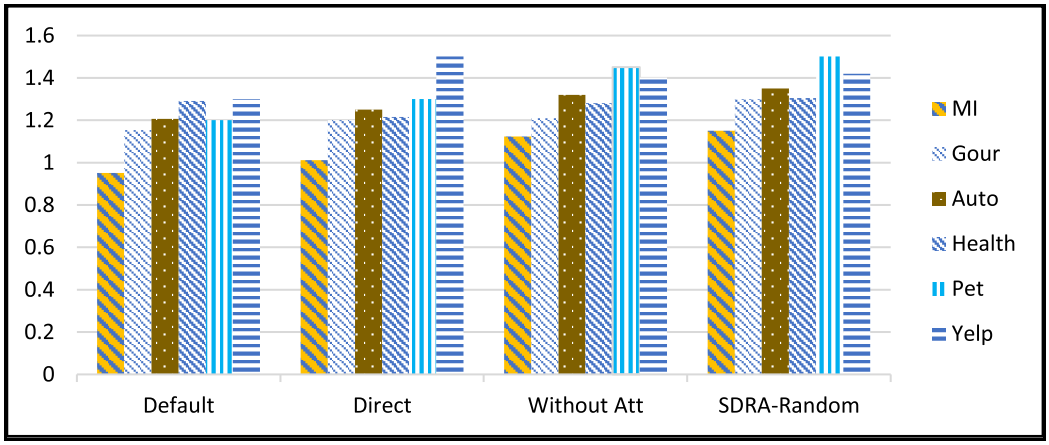

FIGURE 9. MSE results of the model variants on the musical instrument datasets.

document information can be consistently used for the document latent vectors accurately until it reaches the maximum length of 200, and further information cannot be obtained even if the document length is greater than 200. Meanwhile, since the document information to be processed is increased, the model requires more time for effective training. Thus, when processing document information, it is important to take into consideration the trade-off between the training time and the length of the document.

H. ABLATION STUDY

To further examine the performance of our model based on the contribution of the different components, we perform an ablation analysis. To this end, three variants of the model were designed. These different variants are further compared with the default method based on the settings as specified in section 3. The ablation results on all the datasets are shown in figure 9.

- Default Method: In this setting, all the standard components of the model architecture are used as described in section 3
- Direct user/item representation-: In this variant, instead of using the aspect/sentiment aware, we directly derive document representations as in [1] and ignore the aspect and sentiment lexicon embeddings. This is particularly used to examine the impact of the aspect/sentiment aware document representation.
- Without Co-attention: In this setting, we remove the co attention and directly use the user and item latent vectors for rating prediction without considering the user-item interaction at the word level. Here the user-item interaction occurs at the prediction layer.
- SDRA-Random: Instead of using pre-trained word vector for the context vectors, word embedding is learned from scratch.

Due to space limit, we only report the results on the musical instrument datasets. It can be seen that ignoring the sentiment embeddings in the model degrades the performance of the model. It can also be observed that, removing the co attention component from the architecture lower the performance of the model. This clearly indicates the impacts of user-item interaction for better learning user and item representation.

TABLE 5. Some of the aspects learned by our model. Words are shown with the K = 5 for musical instrument datasets.

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

Additionally, it shows that learning word embeddings from scratch is not as good as initializing the word embedding using pre-trained word vectors such as Glove or Google Word2vec.

I. QUALITATIVE ANALYSIS

1) ATTENTION VISUALIZATION

To further understand the working process behind our recommender system, we manually investigate whether our neural attention networks can identify relevant words from both user and item document. In table 4a we highlight words that are regarded as the most important and informative to be considered by the attention module, we choose two review examples from Yelp.

The highlighted words are typically from high-attention scores in the user or item attention network. Different observations can be made from this table. For example, adjective words that describe item's properties are likely highlighted in the review document of the item. Yelp review clearly indicates properties of a specific restaurant by highlighting words such as *bad*, *awful*, *great*, *substandard* etc. more personalized words such as *diner*, *decide*, *waiter* and *fault* are also highlighted in a review of the user.

This shows that the most relevant and important words can indeed be identified by the attention network in the reviews. Similar textual review but different highlights by the item network and user network can be observed in table 4b. The item network and user network are differently trained using a different set of documents. The important part of the review for the final score is decided by the joint training of objective function. Thus, the two networks select different attention words as expected.

We also present in table 5 the top 5 aspects extracted by our topic model.

V. CONCLUSION

This paper proposed a sentiment-aware deep recommender system which extracts the domain-specific aspects and the associated user sentiment lexicon to better learn the user and item sentiment aware representation for improving the performance of the recommender system. The model particularly combines a topic modeling and LSTM sequence encoder using a neural attention mechanism. Specifically, a semi-supervised topic model is designed to learn the domain-specific aspects and the sentiment lexicons for better learning the aspect/sentiment-aware representation of the user and item. The LSTM encoder is used to better capture the semantic and contextual information of words. We introduced a neural co-contention mechanism to better model the fine-grained user-item interaction.

The main impression of our model is that the system is capable to better learn a domain-specific aspect of the products and the associated sentiment lexicons thereby improving the performance of the recommendation system. Experimental results have shown the effectiveness of our proposed model with significant improvement over the state-of-the-art methods. One interesting feature direction is to extend the model such that it deals with both implicit and explicit feedbacks for rating and ranking performance,

REFERENCES

[1] L. Zheng, V. Noroozi, and P. S. Yu, "Joint deep modeling of users and items using reviews for recommendation," in *Proc. WSDM*, Feb. 2017, pp. 1–10.
[2] S. Zhang *et al.*, "Deep learning based recommender system: A survey and new perspectives," *ACM J. Comput. Cult. Herit. Artic.*, vol. 52, no. 1, pp. 1–36, Feb. 2017.
[3] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Comput.*, vol. 42, no. 8, pp. 30–37, Aug. 2009.
[4] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: Understanding rating dimensions with review text," *Proc. 7th ACM Conf. Recomm. Syst.*, Oct. 2013, pp. 165–172.
[5] Y. Bao, F. Hui, and J. Zhang, "TopicMF: Simultaneously exploiting ratings and reviews for recommendation," in *Proc. AAAI*, Jun. 2014, pp. 2–8.
[6] G. Ling, M. R. Lyu, and I. King, "Ratings meet reviews, a combined approach to recommend," in *Proc. 8th ACM Conf. Recomm. Syst. RecSys*, Oct. 2014, pp. 105–112.
[7] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. - KDD*, vol. 11, Aug. 2011, pp. 448–456.
[8] Z. Jin *et al.*, "Jointly modeling review content and aspect ratings for review rating prediction," in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. SIGIR*, Jul. 2016, pp. 893–896.
[9] D. Kim, C. Park, J. Oh, S. Lee, and H. Yu, "Convolutional matrix factorization for document context-aware recommendation," in *Proc. 10th ACM Conf. Recomm. Syst. RecSys* , Sep. 2016, pp. 233–240.
[10] H. Wang, N. Wang, and D.-Y. Yeung, "Collaborative deep learning for recommender systems," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, Aug. 2015, pp. 1235–1244.
[11] D. Kim, C. Park, J. Oh, and H. Yu, "Deep hybrid recommender systems via exploiting document context and statistics of items," *Inf. Sci.*, vol. 417, pp. 72–87, Nov. 2017.
[12] S. Seo, J. Huang, H. Yang, and Y. Liu, "Interpretable convolutional neural networks with dual local and global attention for review rating prediction," in *Proc. Elev. ACM Conf. Recomm. Syst. RecSys*, vol. 17, Aug. 2017, pp. 297–305.
[13] R. Catherine and W. Cohen, "TransNets: Learning to transform for recommendation," in *Proc. 8th ACM Conf. Recommender Syst.*, Aug. 2017, pp. 288–296.
[14] C.-Y. Wu, A. Ahmed, A. Beutel, A. J. Smola, and H. Jing, "Recurrent recommender networks," *Proc. Tenth ACM Int. Conf. Web Search Data Min. WSDM*, vol. 17, Feb. 2017, pp. 495–503.

[15] Y. Tan, M. Zhang, Y. Liu, and S. Ma, "Rating-boosted latent topics: Understanding users and items with ratings and reviews," in *Proc. IJCAI Int. Jt. Conf. Artif. Intell.*, Jan. 2016, pp. 2640–2646.
[16] W. Zhang and J. Wang, "Integrating topic and latent factors for scalable personalized review-based rating prediction," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 11, pp. 3013–3027, Nov. 2016.
[17] Q. Diao, M. Qiu, C.-Y. Wu, A. J. Smola, J. Jiang, and C. Wang, "Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS)," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining* , Aug. 2014, pp. 193–202.
[18] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.
[19] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, Apr. 2017.
[20] S. Li, J. Kawale, and Y. Fu, "Deep collaborative filtering via marginalized denoising auto-encoder," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manag. CIKM*, vol. 15, Oct. 2015, pp. 811–820.
[21] Y. Lu, B. Smyth, R. Dong, and B. Smyth, "Coevolutionary recommendation model: Mutual learning between ratings and reviews," *Proc. World Wide Web Conf. World Wide Web*, Apr. 2018, pp. 773–782.
[22] T. Bansal, D. Belanger, and A. McCallum, "Ask the GRU: Multi-task learning for deep text recommendations," in *Proc. 10th ACM Conf. Recommender Syst.*, Aug. 2016, pp. 107-114.
[23] H. Wu, Z. Zhang, K. Yue, B. Zhang, J. He, and L. Sun, "Dual-regularized matrix factorization with deep neural networks for recommender systems," *Knowl.-Based Syst.*, vol. 145, pp. 46–58, Apr. 2018.
[24] B. Hidasi, M. Quadrana, A. Karatzoglou, and D. Tikk, "Parallel recurrent neural network architectures for feature-rich session-based recommendations," in *Proc. 10th ACM Conf. Recomm. Syst. RecSys*, vol. 16, Sep. 2016, pp. 241–248.
[25] Y. Wu *et al.*, "Collaborative denoising auto-encoders for top-N recommender systems," in *Proc. 9th ACM Int. Conf. Web Search Data Min. WSDM*, vol. 16, May 2016, pp. 153–162.
[26] S. Sedhain, A. K. Menon, S. Sanner, and L. Xie, "AutoRec: Autoencoders meet collaborative filtering," in *Proc. 24th Int. Conf. World Wide Web*, Jun. 2015, pp. 111–112.
[27] D. Hyun, C. Park, M.-C. Yang, I. Song, J.-T. Lee, and H. Yu, "Review sentiment-guided scalable deep recommender system," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 965–968.
[28] Y. Song, A. M. Elkahky, and X. He, "Multi-rate deep learning for temporal recommendation," in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2016, pp. 909–912.
[29] J. Gao, T. Zhang, and C. Xu, "A unified personalized video recommendation via dynamic recurrent neural networks," in *Proc. ACM Multimedia Conf.*, Oct. 2017, pp. 127–135.
[30] P. Nadrowski *et al.*, "Plasma level of N-terminal pro brain natriuretic peptide (NT-proBNP) in elderly population in Poland—The PolSenior Study," *Exp. Gerontol.*, vol. 48, no. 9, pp. 852–857, Sep. 2013.
[31] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. ICLR*, 2014, pp. 5013–5014.
[32] Z. Y. Gao and C. Chen, "deepSA2018 at SemEval-2018 task 1: Multi-task learning of different label for affect in tweets," in *Proc. 12th Int. Workshop Semantic Eval.*, 2018, pp. 226–230.
[33] J. Y. Chin, K. Zhao, S. Joty, and G. Cong, "ANR: Aspect-based neural recommender," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.*, Jun. 2018, pp. 147–156.
[34] J. Chen, F. Zhuang, X. Hong, X. Ao, X. Xie, and Q. He, "Attention-driven factor model for explainable personalized recommendation," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, May 2018, pp. 909–912.
[35] C. Zhou *et al.*, "ATRank: An attention-based user behavior modeling framework for recommendation," in *Proc. AAAI*, Apr. 2018, pp. 45–65.
[36] J. Chen and X. He, "Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention," in *Proc. SIGIR*, Aug. 2017, pp. 335–344.
[37] J. Lu, J. Yang, D. Batra, and D. Parikh, "Hierarchical question-image co-attention for visual question answering," in *Proc. NIIPS*, 2016, pp. 289–297.
[38] T. Mikolov, W. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," in *Proc. NAACL-HLT*, Nov. 2013, pp. 746–751.
[39] M. Shi, S. Member, Y. Tang, and J. Liu, "Functional and contextual attention-based LSTM for service recommendation in Mashup creation," *IEEE Trans. Parallel Distrib. Syst.*, to be published.
[40] M. Yang, "Learning Domain-specific Sentiment Lexicon with Supervised Sentiment-aware LDA," in *Proc. AAAI*, Jul. 2015, p. 2014.
[41] W. M. Darling, "A Theoretical and practical implementation tutorial on topic modeling and Gibbs sampling," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, Jun. 2015, pp. 175–196.
[42] Z. Cheng, Y. Ding, L. Zhu, and M. Kankanhalli, *Aspect-Aware Latent Factor Model: Rating Prediction with Ratings and Reviews*, document IW3C2 2018, 2018.
[43] Y. Tay, L. A. Tuan, and S. C. Hui, "Multi-pointer co-attention networks for recommendation," in *Proc. 24th ACMSIGKDD Inter National Conf. Knowl. Discovery Data Mining*, May 2018, pp. 458–468.
[44] P. D. Turney and M. L. Littman, "Measuring praise and criticism," in *ACM Trans. Inf. Syst.*, vol. 21, no. 4, pp. 315–346, 2003.
[45] A. G. Khachaturyan and G. A. Shatalov, "Elastic energy of heterophase systems of lamellar inclusions.," *Phys. Met. Metallogr.*, vol. 31, no. 6, pp. 1–5, 1971.

AMINU DA'U received the B.Sc. degree in computer science from Usmanu Danfodiyo University Sokoto, Nigeria, and the M.Sc. degree in information technology from NOUN, Lagos, Nigeria. He is currently pursuing the Ph.D. degree in computer science with the Faculty of Engineering, School of Computing, Universiti Teknologi Malaysia. He is a Lecturer with the OTM Department, Hassan Usman Katsina Polytechnic, Katsina, Nigeria. His current research interests include machine learning, data mining, deep leaning techniques, natural language processing (NLP), and recommender systems.

NAOMIE SALIM received the B.Sc. degree in computer science from the Universiti Teknologi Malaysia, the M.Sc. degree in computer science from the University of Western Michigan, and the Ph.D. degree in information studies from the University of Sheffield. She is currently a Professor with the Faculty of Engineering, School of Computing, Universiti Teknologi Malaysia, where she is the Deputy Dean (research and innovation) of the Faculty of Engineering. She has authored over 100 journals and conference papers since the inception of her research career. Her main research interests include text mining, machine learning, information retrieval, cheminformatics, and natural language processing.