IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received September 25, 2020, accepted October 12, 2020, date of publication October 16, 2020, date of current version October 30, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3031621

Rating Prediction Based on Merge-CNN and Concise Attention Review Mining

YUN-CHENG CHOU[1], HSING-YU CHEN[2], DUEN-REN LIU[1], AND DER-SHIUAN CHANG[1]
[1]Institute of Information Management, National Chiao Tung University, Hsinchu 30010, Taiwan
[2]Department of Materials Science and Engineering, National Chiao Tung University, Hsinchu 30010, Taiwan

Corresponding author: Duen-Ren Liu (dliu@mail.nctu.edu.tw)

This work was supported in part by the Ministry of Science and Technology of Taiwan under Grant MOST 108-2410-H-009-046-MY2.

ABSTRACT Online review websites provide an open platform for users to write reviews or give ratings on items (business services) as well as share their consumption experience. However, the volume of reviews is large, while the rating scores provide users with a quick picture of the items without reading all reviews. Recommendation systems can help users find items of interest by predicting user's ratings on unrated items. Review contents contain more personalized preference features than simply user ratings. Therefore, it is important to consider both ratings and review contents when making rating predictions. This research proposes a novel approach that combines deep learning and review mining with attention mechanism for rating predictions. Review mining with attention mechanism is adopted to extract concise attention reviews with important words and sentences. A merge convolutional neural network (merge-CNN) model is proposed to consider both the target user's preference features and performance features of target business for rating prediction. This method extracts quality business performance features from the quality reviews written by elite (credible) users. Moreover, the proposed method uses the concise attention reviews of target user's neighbors to simulate target users' reviews on unrated target business. Experiments were conducted on Yelp data sets to evaluate our proposed methods. The results show that the proposed method, i.e. considering concise attention reviews and quality reviews written by elite users, outperforms traditional methods in improving prediction accuracy. The experiment result also shows that our review simulation methods can well simulate target user's reviews on unrated target business.

INDEX TERMS Recommender system, rating prediction, matrix factorization, review mining, deep learning.

I. INTRODUCTION

Currently, it is very common for users to write reviews or give ratings on items (products, services or businesses) and share their consumption experience through online review websites such as Yelp or Epinion.com. In the online review website Yelp, the businesses referred to the stores or restaurants which provide various kinds of services to customers. The customers can give ratings or reviews on the businesses to express and share their experiences with others. While these reviews are the best reference for people's consumption, due to the rapid accumulation of information, the volume of reviews is quite huge, it leads to the so-called information overload problem. In order to solve this issue, online review websites use the rating scoring mechanism to simplify the problem. Users are not only asked to leave reviews but also to give rating scores. The advantage of using rating scores is that they are good indicators; as a result, others can get a quick picture of the businesses without reading all the reviews.

Rating scores can represent user's preferences, and recommendation systems can help users find potential business items of interest by predicting users' ratings on unrated businesses. Review contents contain more personalized preference features than just user ratings. Therefore, it is important to consider both ratings and review contents in making rating predictions. Recommendation technology [1]–[3] is becoming increasingly important in various applications. The most commonly used algorithms in the recommender system are collaborative filtering (CF) [2], [4] and matrix factorization (MF) [5]. The former uses target users' consumption records and a rating matrix to find neighbors with similar user interests, and recommend the neighbors'

The associate editor coordinating the review of this manuscript and approving it for publication was Hazrat Ali.

preferred items, or businesses in the case of Yelp, to the target user; the latter takes advantage of the matrix decomposition result. By decomposing the rating matrix into latent features of users and businesses, we can estimate the predicted ratings by the dot product of user and business latent vectors.

CF and MF are unable to discern the user preference differences from users with the same ratings, which means they can only learn users' general preferences without considering their specific preferences expressed in reviews. In order to alleviate the above issue, analyzing user review texts offers a solution. Accordingly, recommendation models that consider both rating scores and review texts have gradually been proposed [6]–[8], in which review texts and neural networks have been adopted, in order to improve the accuracy of rating predictions. Deep learning methods, such as recurrent neural networks (RNNs) and convolutional neural networks (CNNs), have been used successfully in natural language processing (NLP) and document classification [9], [10]. Neural network models such as CNNs were also proposed to jointly learn user and business features from the review text [6], [8].

Although several recommendation models have been proposed, some points can be improved. First, texts generally contain words and sentences of different importance [11], [12]. Second, business performance, such as the tastes of foods or the quality of services provided by a restaurant, also plays an important role in deciding whether or not the users will give high rating scores. It remains to determine whether we can generate quality business performance features considering the quality reviews written by credible users. Third, to make rating predictions for the target user on the target business, we need the target user's review on the target business; however, in real conditions, the target user may not yet have commented on the target business, so the target user's review on an unrated target business may not exist. The question arises as to whether there is a method to simulate target users' reviews to make rating predictions.

This research proposes a novel approach that combines deep learning and review mining with attention mechanism for rating predictions. We used deep learning attention models to filter out unimportant words and sentences in the reviews and reform the concise reviews, which consists of the important sentences discovered from the original reviews. For simplicity, we named it concise attention reviews in the rest of this paper. Moreover, we proposed a merge convolutional neural network (merge-CNN) model that considers both the target user's preference features on target business and the performance features of the target business for rating predictions. To generate business performance features, we proposed a method to extract quality business performance features from the quality reviews of elite (credible) users. Lastly, since the review comments of target users are not available for unrated target business, in order to simulate target users' reviews, we proposed two review simulation methods, the Cluster-based simulation method and the Prediction-based simulation method to find neighbors who are most similar to the target user. We then extracted the concise attention reviews of those neighboring users to determine the target user's preference features on an unrated target business, and use them as the input of the merge-CNN model for predicting the ratings on the target business.

We conducted experiments on Yelp data sets to evaluate our proposed methods. The experiment result showed that the proposed model considering both user preference features and business performance features along with concise attention reviews and quality reviews written by elite users can effectively improve prediction accuracy. By analyzing the concise attention reviews with important words and sentences, we not only generated better quality recommendations, but also reduced the model processing time. In addition, the quality reviews written by elite (credible) users can better indicate the performance of a business than all the reviews of the business, and also provide better prediction accuracy. The experiment results also show that our review simulation methods can well simulate target user's reviews. The results indicate that the proposed methods outperform the traditional methods and can improve the accuracy of rating prediction, thereby increasing the commercial value of online review websites.

The contributions of our proposed work are summarized as follows. Existing studies have not considered concise attention reviews and quality reviews written by elite users for rating predictions. The issue of how to simulate target user's review on unrated target business has also not been addressed. This work proposes a novel approach that combines deep learning and review mining with attention mechanism for rating predictions by extracting concise attention reviews with important words and sentences. A merge convolutional neural network (merge-CNN) model is proposed to learn both the target user's preference features and performance features of target business for rating prediction based on concise attention reviews and quality reviews written by elite users. Moreover, the proposed approach uses the concise attention reviews of target user's neighbors to simulate target user's review on unrated target business. The proposed novel approach can take both concise attention reviews of users and quality business performance features to make effective rating predictions.

The rest of this paper is organized as follows. Section II illustrates related work. The proposed approach is presented in section III. Section IV describes the evaluation results. The final section concludes the research and future work.

II. RELATED WORK

In this section, we introduce some recommendation systems first, and then introduce researches on deep learning for text analysis and sentiment analysis.

A. RECOMMENDER SYSTEMS AND MATRIX FACTORIZATION

Recommender systems have been applied in various areas, such as news [13]–[15], products [16], music [17], and fashion [18]. Collaborative filtering (CF) [2] focuses on preferences that may be similar between users; it is intuitive to recommend similar items for users with similar preferences. The content-based filtering (CBF) system analyzes a user's preferences for items to derive a user feature profile, and then recommends items with similar features [19]. Hybrid recommender systems [20] combine CBF and CF for making recommendations. A recommendation model is proposed by combining CF and deep neural networks [21]. Moreover, product recommendation is proposed by considering reviewer credibility and sentiment analysis based User profile modelling [22].

Matrix factorization (MF) is a useful mathematical operation that decomposes one matrix into several low dimension matrices, which allows us to discover the latent features. The MF technique [5] was first used for a recommender system by decomposing the user-item rating matrix into two latent feature matrices corresponding to latent factors of users and items, and later predicting rating scores by the dot product of the two decomposed matrices. Equation (1) expresses the loss function L of MF:

$$L = \sum_{u,i \in S} \left(r_{ui} - x_u^T \cdot y_i \right)^2 + \lambda_x \sum_u \|x_u\|^2 + \lambda_y \sum_i \|y_i\|^2 \tag{1}$$

where x_u and y_i denoted the latent factors of user u and item i respectively, and r_{ui} is the actual rating of user u on item i; λ_x and λ_y are regularization parameters. To minimize the loss function L, Stochastic Gradient Descent (SGD) takes the derivative with respect to each variable in the model and uses the derivate to adjust the latent factors of users and items respectively.

B. DEEP LEARNING FOR TEXT ANALYSIS AND REVIEW MINING

Deep learning methods, such as recurrent neural networks (RNNs) and convolutional neural networks (CNNs), have been used successfully in many fields, such as computer vision [25], natural language processing (NLP) and document classification [9], [10], and so on. Deep learning is successfully applied in text analysis to extract opinions or sentiments from online reviews. Extracting users' opinions needs several NLP steps to identify subjectivity in the given text [11].

Extracting sentiment is to assign the polarity (e.g. positive, neutral, and negative) to the sentences or text by using deep learning methods based on the sentiment analysis. CNNs and RNNs learn data from multiple deep layers of modules. Instead of image pixels, the inputs can be sentences or documents represented as a matrix. CNNs have the ability to extract important n-gram features from sentences to generate the latent semantic representations for NLP tasks, such as sentence, sentiment, and subjectivity classifications [12].

RNNs include the notion of time in neural network models and can carry previous information to the current neural state [26]. Unlike CNN characterized by its ability to extract regional features, RNNs are characterized by their ability to model units in a sequence. User review content usually represents the user's most realistic emotional response. Hochreiter and Schmidhuber [27] designed the long-short-term-memory units (LSTM) to optimize neural units in general RNN. Moreover, a hierarchical structure was proposed for sentiment classification [28], in which a document is composed of sentences, and sentences are composed of words. Moreover, gated recurrent units (GRUs) were introduced [29]. GRUs have fewer parameters than LSTM and do not have an output gate. A bi-direction GRU can be used to extract both forward and backward features from sentence representations for generating the document representations and sentiment classification [28].

The attention mechanism is an interesting design that simulates the habit of attention in the human brain; it helps to focus on important information and improve the effectiveness of information processing. Different sentences and words make different contributions in representing the meaning of a document. A hierarchical attention network based on attention mechanism and bidirectional GRU (Bi-GRU) was proposed for text categorization [30]. Their proposed hierarchical architecture shows the different impact of words and sentences on the text structure. Word-level and sentence-level of attention mechanisms are adopted to enable differential attention to the more important content when generating the document representation.

C. REVIEW-AWARE RECOMMENDATION

The recommender systems have difficulty in analyzing user preferences due to the cold start and rating sparsity problems. The review texts contain useful information, not only for comprehending the users' rating motivation for items, but also to solve the data sparsity problem [23], [24]. Both the topic and sentiment information discovered in users' reviews are valuable information for predicting users' overall preferences [31]. In recent years, the latent topic model [23], [24] or the aspect model [32], [33] have been adopted to successfully model the characteristics and semantics in the review text. It is helpful to improve preference predictions by discovering the latent reasons that users may like or dislike items through the reviews.

Moreover, deep learning techniques have also been applied to design review-aware recommendation. For example, a neural network model was proposed to jointly learn user preferences and item characteristics from the review text [6]. An attention-based convolutional neural network (CNN) was proposed to jointly learn the latent features of each user/item by using the aggregated review text of a user/item [8]. An attentive aspect model was also proposed by using multiple layers of attention networks [34]. It alleviates the sparse aspect problem and the issue due to varied user aspects for different items.

The literature review shows the innovation progress and various properties of the recommender systems. In recent years, deep learning techniques are widely involved and have got great contributions in the field of review-aware recommendations. However, extant studies have not considered concise attention reviews and quality reviews written by elite users for rating predictions. For the scenario that review comments of target users are usually not available for unrated target business, existing studies have not addressed the conundrum of how to simulate target user's reviews on unrated target business. In this paper, we proposed a solution that takes above issues into account for the task of rating prediction.

III. PROPOSED APPROACH

Our research goal is to predict the ratings of target users on target businesses that the target users have no consumption experiences or even awareness of. Assume there is a set of reviews D commented on a set of business items I by a set of users U. The primary problem is to predict the target user's preferences or ratings toward the unrated and uncommented business items.

A. MODEL OVERVIEW

In this section, we introduce our proposed rating prediction approach which combines deep learning and review mining with attention mechanism. FIGURE 1 shows the overview of our proposed approach, which mainly comprises three stages.

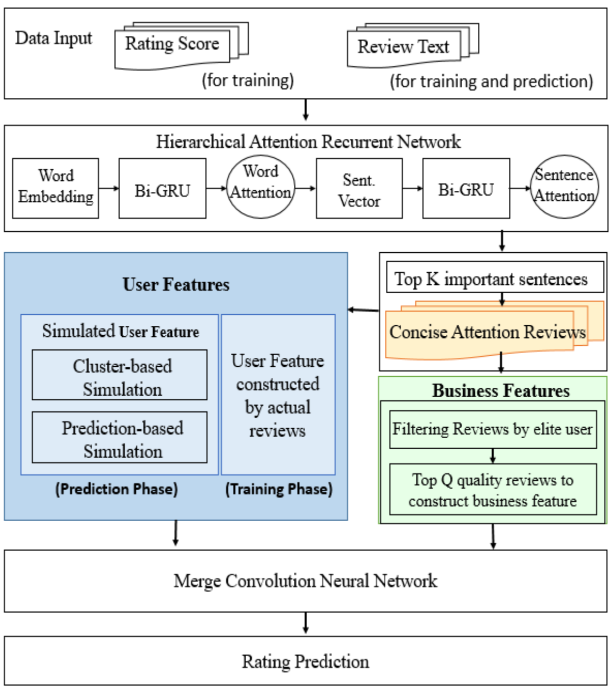

FIGURE 1. Overview of the proposed rating prediction method.

In the first stage, we use pre-trained GloVe word vectors [35] to map each word in the review onto word embedding vectors. The review is made up of sentences that consist of words, which is a hierarchical structure; hence, we represent sentences by concatenating word vectors and express the review by concatenating sentences. Moreover, we use an attention network to calculate the importance weight of each sentence in the review, and filter out the unimportant sentences to extract the concise attention reviews.

The main task of the second stage is to construct features for both user and business. For constructing user preference features, the procedure can be separated as the training phase and prediction phase. For the training phase, the user preference features are constructed by using the target users' actual concise attention reviews on the target businesses that users have rated and commented. For the prediction phase, the target users' ratings and comments on target businesses are unknown. Accordingly, we construct target user's preference features by simulating the target user's reviews on unrated (commented) target business based on neighbors' reviews on the target business. We define two simulation methods based on the MF clustering and MF rating prediction - Cluster-based simulation method and Prediction-based simulation method. Both simulation methods will find similar neighbors of the target user; we adopt the reviews given by similar neighbors to extract important words and sentences as the target user's preference features. For constructing business performance features, we consider the high-quality reviews that are written by elite users on a business and generate business performance features from those reviews. The key notations utilized in the proposed model are shown in Table 1.

TABLE 1. Notations and definitions.

Notation	Definition
$Sentv_s$ $ASentv_s$	The sentence vector of sentence s. The attention sentence vector of sentence s.
$ARR_r(u,b)$	The concise attention review representation of user u's review r on business b.
$RAttTopS_r$	A set of sentences with top-K highest sentence attention weights in review r.
$f_{u,b}$	User preference feature of user u on business b.
f_b	Business b's performance feature.
$Rev(elite,b)$	A set of top Q elite users and reviews on business b.
$Sf_{u,b}$	The simulated target user u's preference features on business b.
$NbrRev_{u,b}^{mfC}$	A set of target user u's top-M similar neighbors regarding business b that is derived from clustering based simulation approach.
$NbrRev_{u,b}^{mfR}$	A set of target user u's top-M similar neighbors regarding business b that is derived from prediction based simulation approach.
$CNNUP_{u,b}$	The convolutional user preference feature of user u's review on business b
$CNNBP_b$	Business b's convolutional business performance feature.

In the third stage, we propose a neural network model: merge-CNN, which is a variant of the general CNN to train our rating prediction model. Our proposed merge-CNN is designed in two parts, user part and business part, which separately model user preference features and business performance features. By separating user and business feature models, we can extract latent features of users and business items from their review text; after that we concatenate user features and business features to make the rating predictions.

B. DERIVING ATTENTION REVIEW FEATURES

In this section, we introduce the approach used to derive the attention features for our model training; they include user preference features and business performance features, which are both extracted from the review. The user preference features are generated from the reviews written by the user, since that user's words are the best representation of his/her preference. The business performance feature refers to the reviews received by the business, which are users' general reflection of the business performance.

1) EXTRACT CONCISE ATTENTION REVIEW FEATURES

There are many ways to map words onto vectors of real numbers like word2vec and bag-of-word. Our approach uses GloVe [35] pre-trained word vectors to transform words. A review is composed of several sentences composed of multiple words, which is a hierarchical structure; therefore, we express each review in a three-dimensional tensor, and the detailed steps are listed below. Let x_{st} denote the word embedding of word t in sentence s, as defined in (2). We use GloVe pre-trained word vectors to map words onto d-dimensional vectors.

$$x_{st} = Embed(w_{st}) \in R^{d}, \quad t \in [1, N] \tag{2}$$

where w_{st} corresponds to the word t in sentence s; d is the dimension of word embedding vector; N is the maximum number of words in a sentence.

Let $Sentv_s$ represent the sentence vector of sentence s. $Sentv_s$ is derived by using (3) to stack the word embedding vectors of the N words in each sentence s.

$$Sentv_s = [x_{s1}, x_{s2}, \ldots, x_{sN}] \in R^{N\times d}, \quad s \in [1, M] \tag{3}$$

where M is the maximum number of sentences in a review. Sentences with fewer than N words are filled up with the zero vector, and sentences with more than N words only take the first N words.

After that, we concatenate M sentence vectors as the review vectors. Reviews with fewer than M sentences are filled up with the zero vector. For reviews with more than M sentences, only the first M sentences are used. Let $Review_r$ denote the review vector of review r. The review vector of each review r in the review set D is expressed in a three-dimensional tensor, as defined in (4).

$$Review_r = [Sentv_{r1} \oplus \ldots \oplus Sentv_{rs} \ldots \oplus Sentv_{rM}], \tag{4}$$

where $Sentv_{rs}$ is the sentence vector of sentence s in review r; the symbol $\oplus$ is utilized to represent the concatenate and stack operations of arrays.

The above review vector representation does not consider the importance of sentences in a review. User review often contains some redundant or meaningless words and sentences. In order to improve the quality of user review, we use the sentence weights of attention layer in the hierarchical attention network (HAN) [30] to filter out unimportant parts of reviews and use important parts to obtain the concise attention reviews. The word-level and sentence-level attention mechanisms of the HAN approach [30] are adopted to generate the attention weights (importance) of words and sentences in a review.

In the word encoder layer, HAN uses bidirectional GRU (Bi-GRU) [36] to derive the latent representations of words by calculating the influence of information from both directions, as defined in (5). Let h_{st} denote the latent representation of word t in sentence s, w_{st}. The word embedding vectors are used as the input of Bi-GRU to derive the output h_{st}. $\overrightarrow{h_{st}}$ and $\overleftarrow{h_{st}}$ correspond to the forward and backward hidden representations of the word w_{st}, respectively.

$$\begin{aligned} \overrightarrow{h_{st}} &= \overrightarrow{GRU}\,(x_{st}), \quad t \in [1, N]; \\ \overleftarrow{h_{st}} &= \overleftarrow{GRU}\,(x_{st}), \quad t \in [N, 1]; h_{st} = [\overrightarrow{h_{st}}, \overleftarrow{h_{st}}] \end{aligned} \tag{5}$$

where x_{st} is the word embedding of w_{st}.

After the encoder layer, HAN uses a word attention layer to calculate the importance of each word in the sentence. Let z_{st} represent the word hidden representation of h_{st}. z_{st} is generated from one layer of MLP. β_{st} is the importance (attention weight) of word w_{st}, as defined in (6).

$$z_{st} = \tanh\,(W_w h_{st} + b_w); \quad \beta_{st} = \frac{\exp(z_{st}^T z_{wc})}{\sum_t \exp(z_{st}^T z_{wc})} \tag{6}$$

where W_w and b_w are the weight matrix and bias matrix of words; z_{wc} is the word level context vector that is randomly generated initially. β_{st} is measured by the similarity between z_{st} and z_{wc} through a softmax function.

Let $ASentv_s$ denote the attention sentence vector of the sentence s. $ASentv_s$ is derived by summing up the weighted latent representations of the words in sentence s, as defined in (7).

$$ASentv_s = \sum_t \beta_{st} h_{st} \tag{7}$$

where β_{st}/h_{st} is the attention weight/latent representation of word w_{st}.

The framework of the sentence encoder/attention layer is the same as the framework of the word encoder/attention layer. Let h_s denote the latent representation of sentence s. The attention sentence vectors are used as the input of Bi-GRU to derive the output h_s, as expressed in (8). $\overrightarrow{h_s}$ and $\overleftarrow{h_s}$ correspond to the forward and backward latent representations of the sentence s, respectively.

$$\begin{aligned} \overleftarrow{h_s} &= \overleftarrow{GRU}\,(Asentv_s), \quad s \in [1, M]; \\ \overleftarrow{h_s} &= \overleftarrow{GRU}\,(Asentv_s), \quad s \in [M, 1] \\ h_s &= [\overrightarrow{h_s}, \overleftarrow{h_s}] \end{aligned} \tag{8}$$

where $ASentv_s$ is the attention sentence vector of sentence s.

Next, a sentence attention layer is used to calculate the attention weight of each sentence in a review. Let z_s denote the sentence hidden representation of h_s. z_s is generated from one layer of MLP. α_s is the attention weight of sentence s, as defined in (9).

$$z_s = \tanh(W_S h_s + b_S); \quad \alpha_s = \frac{\exp(z_s^T z_{sc})}{\sum_s \exp(z_s^T z_{sc})} \tag{9}$$

where W_s and b_s are the weight matrix and bias matrix of sentences; z_{sc} is the sentence level context vector that is randomly generated initially. α_s is measured by the similarity between z_s and z_{sc} through a softmax function.

The sentence attention weight α_s calculated by (9) can well represent the importance of the sentence to the meaning of the review. Therefore, we proposed a novel review attention method based on the sentence attention weights to derive concise attention review representations for reviews.

Let $ARR_r(u, b)$ denote the concise attention review representation of user u's review r on business b. We first adopt the HAN model to calculate the sentence attention weights of the sentences in review r by (9). We then select K sentences with the top K highest sentence attention weights and concatenate the sentence vectors of those sentences to derive $ARR_r\,(u, b)$, as expressed in (10).

$$ARR_r\,(u, b) = \bigoplus_{s \in RAttTopS_r} Sentv_s \tag{10}$$

where $RAttTopS_r$ is the set of sentences with top-K highest sentence attention weights in review r; $Sentv_s$ is the sentence vector of sentence s. Notice that the symbol $\oplus$ is utilized to represent the concatenate and stack operations of arrays.

2) USER PREFERENCE FEATURES AND BUSINESS PERFORMANCE FEATURES

A user's words are the best representation of the user's preferences; therefore, we use a user's concise attention review representation $ARR_r(u, b)$ generated from (10) to represent the user preference features $f_{u,b}$ of user u on business b, as shown in (11).

$$f_{u,b} = ARR_r(u, b) \tag{11}$$

Business performance features f_b refer to the features of business reviews received by the business. The user data contain personal details and special tag to indicate whether the user is a certified elite who had written enough reviews and received enough responses from others. Let $RS_{u,b}$ be the user u's rating score on business b and $\overline{RS_b}$ be the average rating score of business b. In order to generate the high-quality business performance feature of business b, we restrict the reviews written by elite users and then select top Q reviews with rating score $RS_{u,b}$ closest to the average rating score $\overline{RS_b}$ on business b. The rationale of our design is to construct the business performance features based on the reviews of the normal population of elite users. The reviews of elite users who have ratings extremely deviated from the normal population are not considered. It makes the reviews of business b less affected by the outliers.

Let f_b denote the business b's performance features. f_b is derived by concatenating the concise attention review representations of elite users' top-Q reviews on business b, as defined in (12).

$$f_b = \bigoplus_{(u,r) \in Rev(elite,b)} ARR_r(u, b) \tag{12}$$

where $ARR_r(u, b)$ is the concise attention review representation of user u's review r on business b. $Rev\,(elite, b)$ is the set of the pairs of elite user u and u's review r on business b, where $\left|RS_{u,b} - \overline{RS_b}\right|$ is the top-Q smallest among $\left|RS_{v,b} - \overline{RS_b}\right|$ for any elite user v who had review and rating on business b.

C. MERGE-CNN MODEL BASED ON ATTENTION REVIEW FEATURES

We propose a neural network model: merge-CNN, which is a variant of the general CNN, to build our prediction model. The architecture of the proposed merge-CNN is shown in FIGURE 2. In the merge-CNN, the user preference features and business performance features are modeled by the user network and the business network separately. The same structure is used for the user and business network. Thus, we only explain the user network.

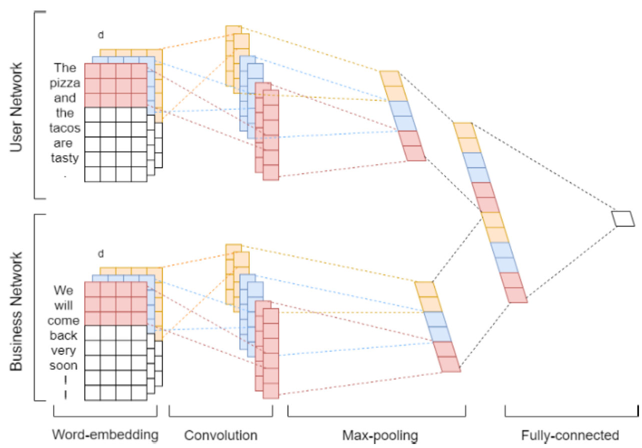

FIGURE 2. The architecture of proposed merge-CNN.

The first layer in the user network is the convolution layer and the input of the convolution layer is the user preference feature $f_{u,b}$, which is the user u's concise attention review representation on business b and is derived by (11). The user preference feature $f_{u,b}$ consists of K sentences and each sentence have N words with d embedding dimensions; therefore, the size of each user preference feature will be $(K \times N) \times d$.

We use n_{filter} filters in the convolution layer and each filter will map user preference feature according to different filter sizes. Let $c_{i,j}$ denote the output features of the ith convolution of user preference feature for the jth filter, and c_j denote the output feature maps of the jth filter of the convolution layer, as expressed in (13) and (14).

$$c_{i,j} = g\left(w_j * f_{u,b}[i : i+H-1] + b_j\right), \quad i \in [1, K \times N - H + 1] \tag{13}$$

$$c_j = \left[c_{1,j}, \ldots, c_{i,j}, \ldots, c_{K\times N-H+1,j}\right], \quad j \in [1, n_{filter}] \tag{14}$$

where H corresponds to the different filter (context window) sizes of the filter; H decides the number of surrounding words. $f_{u,b}[i : i+H-1]$ is the word embedding vectors within the filtering window $[i : i+H-1]$ of the user preference feature $f_{u,b}$. g is a non-linear activation function; in our model, we use rectified linear unit (RELU) as the activation function. $*$ is a convolution operator. Each filter with shared weight $w_j \in R^{H\times d}$ is applied to a user preference feature $f_{u,b}$ with bias b_j, for $j \in [1, n_{filter}]$. Let $n_{filterSize}$ denote the number of filter sizes; thus, the number of output feature maps will be $n_{filter} \times n_{filterSize}$. In our model, $n_{filter} = 128$, $n_{filterSize} = 3$, and H is set to 3, 4, 5, respectively. For the convolution layer, the parameters of strides = [1,1,1,1] and padding= "valid".

The second layer in the network is the pooling layer. Let $CNNUP_{u,b}$ denote the convolutional user preference feature of user u's review on business b. $CNNUP_{u,b}$ is derived by using max-pooling to perform a down sampling operation on the output feature map c_j of the convolution layer, as expressed in (15).

$$CNNUP_{u,b} = [max(c_1), max(c_2), \ldots, max(c_{n_{filter}})], \tag{15}$$

where $max(c_j)$ extracts the maximum feature from each feature map of c_j derived from different filter size H of the convolution layer. After max-pooling, the size of $CNNUP_{u,b}$ is $n_{filter} \times n_{filterSize}$.

Similarly, the convolutional business performance feature $CNNBP_b$ is generated from the business network. The input of the convolution layer is the business performance feature f_b, which is derived from (12) by concatenating the concise attention review representations of elite users' top-Q reviews on business b. We concatenate the convolutional user preference feature $CNNUP_{u,b}$ and the convolutional business performance feature $CNNBP_b$ to generate the rating prediction feature $Z_{u,b}$, as shown in (16).

$$Z_{u,b} = [CNNUP_{u,b} \oplus CNNBP_b], \tag{16}$$

The final layer of our merge-CNN model is the fully-connected (FC) layer with *softmax_cross_entropy with_logits* function; the rating prediction feature $Z_{u,b}$ will be sent to the FC layer for making the final rating prediction. The weights in FC layer are learnt based on the user-item ratings to represent the relationships among user features, business features and the ratings of users on business items. Notice that the symbol $\oplus$ is utilized to represent the concatenate and stack operations of arrays.

During the model training, we use MAE (Mean Absolute Error) to define the training loss function $L\left(RS_{u,b}, \widehat{RS}_{u,b}\right)$, where $RS_{u,b}$ / $\widehat{RS}_{u,b}$ is the rating score/predicted rating score of user u on business b. We adopt Adam as the optimizer to compute the gradients for a loss, and apply gradients to optimize the weight and bias of the model until the loss converges.

D. REVIEW SIMULATION FOR TARGET BUSINESS

In predicting the target user's unknown preference for the target business, the target user does not have review (consumption experience) with the target business. Therefore, to solve the review lacking problem of the target user in the prediction phase, we adopted the strategy that simulated the target user's reviews by using the reviews of target user's neighbors. The MF (matrix factorization) technique constructs latent factors (vectors) for users and items through the rating interactions between them, and has performed well on rating prediction. Thus, we design two simulation methods based on the MF technique, namely the Cluster-based simulation and Prediction-based simulation. The Cluster-based simulation first uses MF to generate user latent vectors, then groups users using user latent vectors, and finally generates the target user's simulated review by using the reviews of users in target user's group. The Prediction-based simulation first uses MF to make a rating prediction on the target user, then uses the predicted rating score to find neighbors with rating scores similar to those of the target user, and finally generates the target user's simulated review by using the reviews from those neighbors.

1) CLUSTER-BASED REVIEW SIMULATION

We consider that users in the same user group will have a high degree of preference similarity. Accordingly, simulating the target user's review by using user reviews in the same group may yield good results; thus, we design a Cluster-based simulation method. The Cluster-based simulation method uses MF to decompose the rating matrix into the user's latent vector U and the business latent feature B; then, by using the K-means technique, we cluster users into groups based on the user latent vectors U. Moreover, we use $UC(u)$ to represent the set of users that are in the same group as u.

Since we want to simulate the target user's review, we need to find the reviews of several similar users to generate it. We select reviews that are written for the target business b whose authors are also in the same group as the target user u. Then we concatenate the top-M similar neighbors' concise attention reviews on business b to represent the target user u's simulated preference features regarding the target business b. The cosine similarity is adopted to compute the similarity between users u and v in terms of their user latent vectors derived from matrix factorization. Let $Sf_{u,b}$ represent the simulated target user u's preference features on business b, as shown in (17).

$$Sf_{u,b} = \bigoplus_{v \in NbrRev^{mfC}_{u,b}} f_{v,b} \tag{17}$$

where $f_{v,b}$ is the user v's preference feature on business b, i.e., the concise attention review representation $ARR_r(v, b)$

derived from (10). $NbrRev_{u,b}^{mfC}$ is the set of target user u's top-M similar neighbors regarding business b, where the similarity between target user u and the neighbor is the top-M highest among users in $UC(u)$ who had review on business b.

2) PREDICTION-BASED REVIEW SIMULATION

The Prediction-based simulation finds the reviews of neighbors with true rating scores closest to the MF predicted rating score of the target user. The method uses MF to predict the rating score $RS_{u,b}$ for target user u on target business b by multiplying target user u's latent vector U_u and target business b's latent feature B_b.

Then, we select reviews that are written for the target business b, and calculate the gap $RSdiff_{u,v,b}$ between the true rating score $TRS_{v,b}$ of the review and the predicted rating score $RS_{u,b}$ of the target user u on the target business b.

Finally, we use the rating score difference to select the concise attention reviews of top-M similar neighbors with smallest rating difference on business b; the smaller the gap, the higher the similarity. We concatenate those concise attention reviews to represent the target user $u's$ simulated preference feature $Sf_{u,b}$ on the target business b, as shown in (18).

$$Sf_{u,b} = \bigoplus_{v \in NbrRev_{u,b}^{mfR}} f_{v,b} \tag{18}$$

where $f_{v,b}$ is the user v's preference feature on business b. $NbrRev_{u,b}^{mfC}$ is the set of target user u's top-M similar neighbors regarding business b, where the neighbor's rating difference $RSdiff_{u,v,b}$ with target user u is the top-M smallest among users who had rated business b. Notice that $RSdiff_{u,v,b}$ is the difference between the true rating score $TRS_{v,b}$ of the review and the predicted rating score $RS_{u,b}$ of the target user u on the target business b.

IV. EXPERIMENT AND EVALUATION

We first adjust the parameters to find the best combination of parameters that fit the proposed rating prediction method. In addition, we evaluate the effect of proposed concise attention review and business performance features for rating predictions. Finally, we compare our proposed rating prediction methods with the baseline methods.

A. DATA DESCRIPTION AND EXPERIMENT SETUP

The dataset used is from Yelp Academic Dataset provided by Yelp. The dataset contains collections of data on the business, user, and review. The user data contain personal details and special tags to indicate whether the user is a Yelp certified elite who had written enough reviews and received sufficient responses from others. The business data contain multiple different categories of business information. As the catering business is the largest category in the business data, we only consider business related to catering services. Moreover, the region where the business is located will also affect the qualities of the business, so we only consider business located in Arizona.

Users with too few reviews are classified as inactive users, and it is difficult to extract user preferences for inactive users. Similarly, it is also difficult to extract business performance features from businesses that received too few comments. Therefore, to improve the data quality, we filter out users who have fewer than twenty-five reviews and businesses that received less than twenty-five reviews. After pruning the dataset, we split the review data into training data and test data. For users with fewer than thirty reviews, all of their reviews will be used as a training set, and for users with more than thirty reviews, 75% of their reviews are for the training set and 25% for the testing set. We use Mean Absolute Error (MAE) metric to evaluate our experiments. The lower the MAE, the better the performance of our model. The proposed methods are developed using software tools, including TensorFlow, scikit learn, genism and pymongo. Our experiments are executed under the hardware environment of Intel Core i7-7700 CPU, 32GB RAM and single NVIDIA GeForce GTX 1080 Ti GPU.

In the experiment, we compared the following methods, including various proposed methods and baseline methods.

- Attention-based-merge-CNN with user preference features and business performance features (AMCNN-UPBP): Using the attention mechanism to generate concise reviews, and extract user preference features and business performance features from the concise attention reviews to train the merge-CNN model. AMCNN-UPBP uses the rating prediction feature $Z_{u,b}$ (16) to train the model.
- Merge-CNN with user preference features and business performance features (MCNN-UPBP): The only difference from the AMCNN-UPBP method is that MCNN-UPBP uses original reviews rather than the concise attention reviews to derive the user preference features and business performance features.
- Attention-based-CNN with user preference features (ACNN-UP): Using the attention mechanism to generate concise reviews, and extract only user preference features from the concise attention reviews to train the CNN model. ACNN-UP does not consider the business performance features.
- CNN with user preference features (CNN-UP): The only difference from the ACNN-UP method is that CNN-UP uses original reviews rather than the concise attention reviews to derive the user preference features. CNN-UP does not consider the business performance features.
- Matrix Factorization (MF): The MF decomposes the rating matrix into latent factor matrices of users and items for making the rating prediction. In the MF parameter setting, we set the dimensions of the latent factor to 512, dropout to 0.6 and epochs to 150 with batch size of 5000.
- Hierarchical Attention Network (HAN): The HAN [30] is a hierarchical neural network model with attention mechanisms for document classification. The method is adopted by using the review text as the input and ratings

as the output, which proved to be effective on rating prediction. In the HAN parameter setting, we set hidden layers to 128, learning rate to 0.001, dropout to 0.6 and epochs to 50 with batch size of 256.

B. THE PARAMETER ADJUSTMENT FOR THE PROPOSED METHODS

1) EVALUATION OF ADJUSTMENT OF BUSINESS PERFORMANCE FEATURES

The business performance features were extracted from top-Q elite users' reviews, and each elite user review was represented by the top-K important sentences. We evaluated the effects of both the number of reviews and sentences on business performance feature extraction. In our experiment, we compared top-Q important reviews received by the business to extract the business performance features (BPF). We fixed another parameter, the number of sentences (K) for each review.

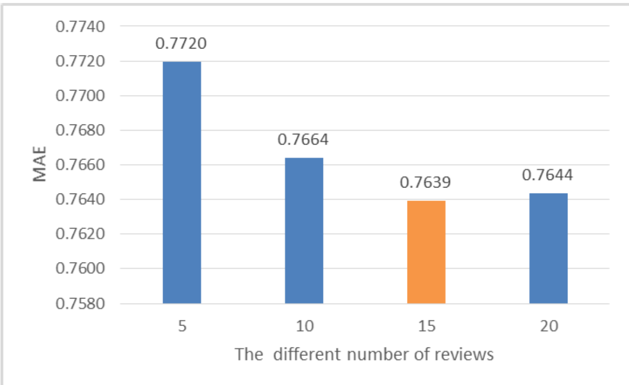

FIGURE 3. Comparison of the different number of reviews in BPF.

The experiment results are presented in **FIGURE 3**. We can observe that the MAE is the highest in five reviews and decreases as the number of reviews increases; it has a minimum MAE in fifteen reviews. We infer that when the model extracts business performance features from too few reviews, it can easily lead to biased results; however, when the model extracts business performance features from a sufficient number of reviews, it can extract most of the key performance features on business, thereby reducing the MAE.

Similarly, business performance features are also affected by the content of the review, which consists of numbers of sentences K. The experiment results shown in **FIGURE 4** indicate that the performance is best when the number of sentences in each review is five.

2) EVALUATION OF SIMULATED USER PREFERENCE FEATURES

The target user's preference features (UPF) on unrated target business are simulated from top-M similar users' reviews; each similar user's review is represented by the top-K important sentences. We conducted several comparisons to evaluate

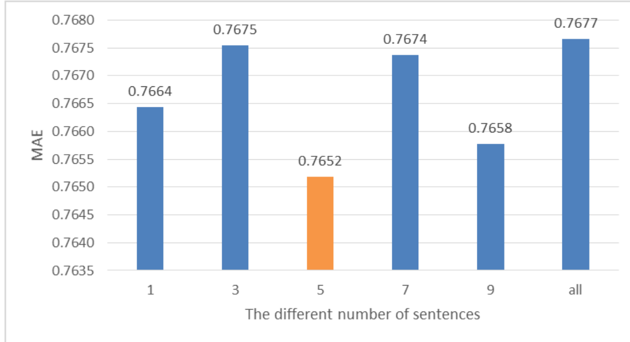

FIGURE 4. Comparison of different number of sentences in BPF.

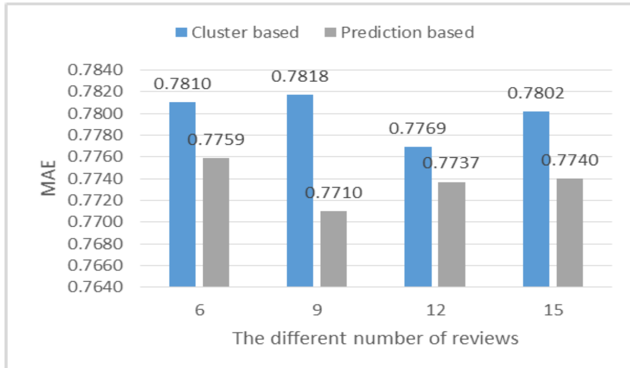

FIGURE 5. Comparison of the different number of reviews in UPF.

the effect of the different number of reviews and sentences. Figure 5 shows the MAE of the proposed cluster-based and prediction-based review simulation under different number of reviews. The results show that the prediction-based review simulation performs better than the cluster-based review simulation.

For prediction-based review simulation, the MAE declined with the increase in the number of reviews M and maintained slight fluctuations after reaching the minimum of nine reviews. We infer that, as long as the number of reviews M used to extract user preference features exceeds a certain number, nine, it can adequately represent the target user's preferences feature on unrated target business.

In addition, we evaluated the effect of different numbers of sentences K on the extraction of user preference features. The experiment results shown in **FIGURE 6** indicate that the prediction-based review simulation performs better than the cluster-based review simulation. Moreover, MAE gradually decreased from the highest in one sentence to the lowest in three sentences, and then rebounded again. It implies that the prediction-based review simulation model can adequately extract user preference features from the review represented by three sentences. The experimental result is quite reasonable because the original review always contains some redundant sentences.

TABLE 2. The comparison of models with and without the attention mechanism.

	Prediction-based		Cluster-based	
Without Attention	MCNN-UPBP	CNN-UP	MCNN-UPBP	CNN-UP
MAE	0.7424	0.7445	0.761	0.7664
With Attention	AMCNN-UPBP	ACNN-UP	AMCNN-UPBP	ACNN-UP
MAE	0.7268	0.7379	0.7389	0.7494
Improvement rate	2.1%	0.89%	2.9%	2.22%

TABLE 3. The comparison of models with and without the business performance features.

	Prediction-based		Cluster-based	
Without BP	ACNN-UP	CNN-UP	ACNN-UP	CNN-UP
MAE	0.7379	0.7445	0.7494	0.7664
With BP	AMCNN-UPBP	MCNN-UPBP	AMCNN-UPBP	MCNN-UPBP
MAE	0.7268	0.7424	0.7389	0.761
Improvement rate	1.5%	0.28%	1.4%	0.7%

C. EVALUATION OF OUR PROPOSED METHODS AND BASELINE METHODS

1) THE EFFECT OF ATTENTION MECHANISM AND BUSINESS PERFORMANCE FEATURES

Experiments were conducted to evaluate the effect of concise attention reviews derived by using the attention mechanism to extract important words and sentences in reviews. We compared the methods with and without attention mechanism. The comparison results presented in Table 2 show that the methods with attention mechanism (AMCNN-UPBP and ACNN-UP) performed better than the methods without attention mechanism (MCNN-UPBP and CNN-UP). The prediction-based review simulation performed better than the cluster-based review simulation.

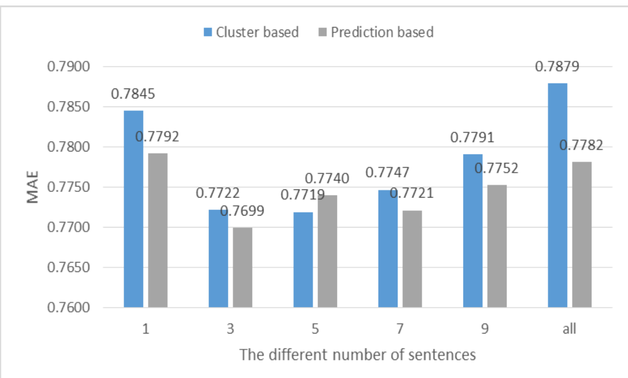

FIGURE 6. Comparison of the different number of sentences in UPF.

Our proposed methods use user preference features and business performance features to make rating predictions. For example, the proposed AMCNN-UPBP contains a merge-CNN model with user preference features and business performance features based on concise attention reviews.

We compared methods with and without the business performance features. The comparison results in Table 3 show that the methods with business performance features (AMCNN-UPBP and MCNN-UPBP) perform better than the methods (AMCNN-UP and MCNN-UP) without business performance features; the maximum improvement can reach 1.5%.

2) THE COMPARISON OF PROPOSED METHODS AND BASELINE METHODS

In addition, we compared the performance of each proposed methods using Cluster-based review simulation and the Prediction-based review simulation, and finally compared the proposed methods with the baseline methods for rating prediction.

The comparison results in **FIGURE 7** show that Prediction-based review simulation is superior to Cluster-based review simulation among all of our proposed rating prediction methods. In addition, the proposed rating prediction method (AMCNN-UPUB) with attention mechanism and merge-CNN model considering both user preference and business performance features is the best of all the proposed methods. Specifically, AMCNN-UPBP performs better than the MCNN-UPBP method. The result implies that the proposed concise attention reviews derived by using attention mechanism to extract important words and sentences in reviews can effectively improve the quality of recommendations. Moreover, AMCNN-UPBP performs better than the ACNN-UP method. It implies that the proposed merge-CNN model with user preference features and business performance features can further improve the rating predictions.

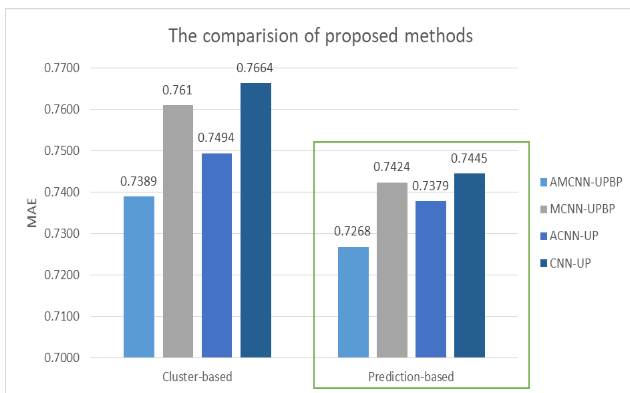

FIGURE 7. Effectiveness of the proposed methods under different review simulation methods.

In the following comparison with baseline methods, we compared our proposed rating prediction methods using Prediction-based review simulations to the others; all the methods using Prediction-based review simulations

TABLE 4. Comparison between proposed methods and baseline methods.

Proposed method	AMCNN-UPBP	MCNN-UPBP	ACNN-UP	CNN-UP
MAE	0.7268	0.7424	0.7379	0.7445
MF method	0.7694			
Improvement rate over MF	5.53%	3.51%	4.09%	3.24%
HAN method	0.747			
Improvement rate over HAN	2.7%	0.62%	1.22%	0.33%

outperform Cluster-based review simulations, as shown in **FIGURE 7**. The comparison results presented in Table 4 show that the methods with attention mechanism, AMCNN-UPBP and ACNN-UP outperform methods without attention mechanism, MCNN-UPBP and CNN-UP. In addition, the methods with business preference features, AMCNN-UPBP and MCNN-UPBP outperform the methods without business preference features, ACNN-UP and CNN-UP. Finally, the method with both the attention mechanism and business performance features, which is the AMCNN-UPBP, performed the best, with maximum improvement over MF and HAN for 5.53% and 2.7%, respectively. Therefore, we can infer that by using attention mechanism and business performance features, our proposed methods can improve the rating predictions.

V. DISCUSSIONS AND CONCLUSION

In this paper, we proposed a rating prediction model, merge-CNN, based on a deep learning framework and review mining with attention mechanism. By using GloVe to convert user reviews into vector input, our proposed merge-CNN model can effectively extract important user preference features from user reviews to make better recommendation predictions.

Since user preference features are extracted directly from user reviews, the quality of user preference features is influenced by the quality of the user reviews. Accordingly, our proposed method, considering concise attention reviews and quality reviews written by elite users, can improve the prediction accuracy. Through the attention mechanism, we can filter out unimportant sentences in user reviews to get a concise attention review with higher quality. The quality reviews written by elite users can better indicate the performance of business than all the reviews of the business. Accordingly, extracting business performance features from the quality reviews of elite (credible) users can also provide better prediction accuracy. In addition, we can effectively improve the accuracy of rating prediction by combining business performance features with user preferences. The experimental results verify our inference that the recommendation model adopting attention mechanism and business performance features can improve accuracy over the baseline models.

Moreover, a target user does not actually have consumption experience with the target business, not to mention writing a review on the target business. In order to extract the preference features from the target user, we designed two user review simulation methods to simulate the target user's possible review. Through the proposed methods, we can extract the possible preference features of the target user from the target user's most similar neighbors, and then make a rating prediction for the target user. As a result, our experiments have verified that this approach is feasible.

Existing studies have not considered concise attention reviews and quality reviews written by elite users for rating predictions. Our experiment results show that the proposed novel merge-CNN model can take both concise attention reviews of users and quality business performance features to make effective rating predictions. Our research results help to improve the accuracy of rating prediction and increase the commercial value of online review websites.

There are some issues that can be improved in the future. We extracted the business performance features from multiple credible reviews received by a business. Determining how to define a credible review will affect the quality of the business performance features. With the strict Yelp elite qualification requirements, the reviews written by elite Yelp users are credible; therefore, we selected credible reviews written by Yelp elite users to extract business performance features. However, other online review web sites may not have the elite authentication mechanism like Yelp. To apply our recommendation models to other data set, it is necessary to design an approach that can automatically evaluate the credibility of the review. We plan to investigate such approach and evaluate our approach using more datasets and more baselines in future work. In addition, our model has not been able to accurately predict the ratings of users who have too few consumer records (the cold-start problem). These are the issues that can be further studied and improved in the future.

REFERENCES

[1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.

[2] F. Alyari and N. J. Navimipour, "Recommender systems: A systematic review of the state of the art literature and suggestions for future research," *Kybernetes*, vol. 47, no. 5, pp. 985–1017, May 2018.

[3] D.-R. Liu, Y.-S. Liao, Y.-H. Chung, and K.-Y. Chen, "Advertisement recommendation based on personal interests and ad push fairness," *Kybernetes*, vol. 48, no. 8, pp. 1586–1605, Sep. 2019.

[4] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," in *Proc. 10th Int. Conf. World Wide Web (WWW)*, 2001, pp. 285–295.

[5] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[6] L. Zheng, V. Noroozi, and P. S. Yu, "Joint deep modeling of users and items using reviews for recommendation," in *Proc. 10th ACM Int. Conf. Web Search Data Mining (WSDM)*, Cambridge, U.K., 2017, pp. 425–434.

[7] D. Tang, B. Qin, T. Liu, and Y. Yang, "User modeling with neural network for review rating prediction," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 1340–1346.

[8] 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. 11th ACM Conf. Recommender Syst.*, Como, Italy, Aug. 2017, pp. 297–305.

[9] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Doha, Qatar, 2014, pp. 1–6.
[10] Y. Zhang and B. Wallace, "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification," in *Proc. 8th Int. Joint Conf. Natural Lang. Process.*, Taipei, Taiwan, 2015, pp. 253–263.
[11] S. Sun, C. Luo, and J. Chen, "A review of natural language processing techniques for opinion mining systems," *Inf. Fusion*, vol. 36, pp. 10–25, Jul. 2017.
[12] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing [Review Article]," *IEEE Comput. Intell. Mag.*, vol. 13, no. 3, pp. 55–75, Aug. 2018.
[13] D.-R. Liu, Y.-S. Liao, and J.-Y. Lu, "Online news recommendations based on topic modeling and online interest adjustment," *Ind. Manage. Data Syst.*, vol. 119, no. 8, pp. 1802–1818, Sep. 2019.
[14] Z. Zhu, D. Li, J. Liang, G. Liu, and H. Yu, "A dynamic personalized news recommendation system based on BAP user profiling method," *IEEE Access*, vol. 6, pp. 41068–41078, 2018.
[15] C. Feng, M. Khan, A. U. Rahman, and A. Ahmad, "News recommendation systems-accomplishments, challenges & future directions," *IEEE Access*, vol. 8, pp. 16702–16725, 2020.
[16] P. D. Lakshmi, K. Ramani, and B. E. Reddy, "Neighborhood algorithm for product recommendation," in *Computational Intelligence and Big Data Analytics*. Singapore: Springer, 2019, pp. 37–53.
[17] A. Patel and R. Wadhvani, "A comparative study of music recommendation systems," in *Proc. IEEE Int. Students' Conf. Electr., Electron. Comput. Sci. (SCEECS)*, Feb. 2018, pp. 1–4.
[18] Q. Wu, P. Zhao, and Z. Cui, "Visual and textual jointly enhanced interpretable fashion recommendation," *IEEE Access*, vol. 8, pp. 68736–68746, 2020.
[19] P. Lops, M. De Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Boston, MA, USA: Springer, 2011, pp. 73–105.
[20] R. Burke, "Hybrid recommender systems: Survey and experiments," *User Model. User-Adapted Interact.*, vol. 12, no. 4, pp. 331–370, Nov. 2002.
[21] L. Zhang, T. Luo, F. Zhang, and Y. Wu, "A recommendation model based on deep neural network," *IEEE Access*, vol. 6, pp. 9454–9463, 2018.
[22] S. Hu, A. Kumar, F. Al-Turjman, S. Gupta, S. Seth, and Shubham, "Reviewer credibility and sentiment analysis based user profile modelling for online product recommendation," *IEEE Access*, vol. 8, pp. 26172–26189, 2020.
[23] R. Dong, M. Schaal, M. P. O'Mahony, and B. Smyth, "Topic extraction from online reviews for classification and recommendation," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, Beijing, China, 2013, pp. 1310–1316.
[24] G. Ling, M. R. Lyu, and I. King, "Ratings meet reviews, a combined approach to recommend," in *Proc. 8th ACM Conf. Recommender Syst. (RecSys)*, Silicon Valley, CA, USA, 2014, pp. 105–112.
[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," presented at the 25th Int. Conf. Neural Inf. Process. Syst., Lake Tahoe, Nevada, 2012.
[26] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci. USA*, vol. 79, no. 8, pp. 2554–2558, 1982.
[27] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[28] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," presented at the Conf. Empirical Methods Natural Lang. Process., Lisbon, Portugal, 2015.
[29] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*. [Online]. Available: http://arxiv.org/abs/1406.1078
[30] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2016, pp. 1480–1489.
[31] G. Ganu, N. Elhadad, and A. Marian, "Beyond the stars: Improving rating predictions using review text content," in *Proc. 12th Int. Workshop Web Databases (WebDB)*, 2009.
[32] X. He, T. Chen, M.-Y. Kan, and X. Chen, "Trirank: Review-aware explainable recommendation by modeling aspects," in *Proc. Int. Conf. Inf. Knowl. Manage.*, 2015, pp. 1661–1670.
[33] Z. Cheng, X. Chang, L. Zhu, R. C. Kanjirathinkal, and M. Kankanhalli, "MMALFM: Explainable recommendation by leveraging reviews and images," *ACM Trans. Inf. Syst.*, vol. 37, no. 2, pp. 1–28, Mar. 2019.
[34] X. Guan, Z. Cheng, X. He, Y. Zhang, Z. Zhu, Q. Peng, and T.-S. Chua, "Attentive aspect modeling for review-aware recommendation," *ACM Trans. Inf. Syst.*, vol. 37, no. 3, pp. 1–27, Jul. 2019.
[35] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.
[36] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

YUN-CHENG CHOU received the M.S. degree in information management from National Chiao Tung University, Taiwan, in 2012, where he is currently pursuing the Ph.D. degree with the Institute of Information Management. His research interests include data mining, machine learning, and recommender systems.

HSING-YU CHEN received the B.S. and M.S. degrees in mechanical engineering from National Central University, Taiwan, in 2013 and 2015, respectively. He is currently pursuing the Ph.D. degree with National Chiao Tung University, Taiwan. His research interests include deep learning, recommender systems, and knowledge engineering.

DUEN-REN LIU received the B.S. and M.S. degrees in computer science from National Taiwan University, Taiwan, in 1985 and 1987, respectively, and the Ph.D. degree in computer science from the University of Minnesota, USA, in 1995. He is a Professor with the Institute of Information Management, National Chiao Tung University, Taiwan. His research interests include data mining, knowledge engineering, electronic commerce, and recommender systems.

DER-SHIUAN CHANG received the B.S. degree in information management from National Central University, Taiwan, in 2016, and the M.S. degree in information management from National Chiao Tung University, Taiwan, in 2018. His research interests include information systems, data mining, electronic commerce, and Web applications.