IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 12, 2018, accepted December 9, 2018, date of publication December 17, 2018, date of current version January 11, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2887100

Integrating an Attention Mechanism and Convolution Collaborative Filtering for Document Context-Aware Rating Prediction

BANGZUO ZHANG[1], (Member, IEEE), HAOBO ZHANG[1], XIAOXIN SUN[1], GUOZHONG FENG[1], AND CHUNGUANG HE[2]

[1]School of Information Science and Technology, Northeast Normal University, Changchun 130117, China
[2]State Environmental Protection Key Laboratory of Wetland Ecology and Vegetation Restoration, Northeast Normal University, Changchun 130024, China

Corresponding author: Guozhong Feng (gzfeng@outlook.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 71473035 and Grant 11501095, in part by the Scientific and Technological Development Program of Jilin Province of China under Grant 20170520051JH, Grant 20150204040GX, and Grant 20150101057JC, in part by the Jilin Province Development and Reform Commission Project of China under Grant 2015Y055 and Grant 2015Y054, and in part by the Open Fund of the State Environmental Protection Key Laboratory of Wetland Ecology and Vegetation Restoration, Northeast Normal University, under Grant 130028813.

ABSTRACT Deep learning has become a recent, modern technique for big data processing, with promising results and large potential. For recommender systems, user and item information can be used as input vectors to perform prediction tasks. However, augmenting the number of layers to improve feature extraction will increase the computational complexity considerably and may not achieve the desired results. This paper proposes a method called attention convolution collaborative filtering (Att-ConvCF), which integrates an attention mechanism with a collaborative filtering model to improve the effectiveness of the feature extraction by reassigning the weights of feature vectors. Descriptive documents for the items are used to enrich the background information through a convolutional neural network. Finally, extensive experiments with real-world datasets were performed, and the results showed that Att-ConvCF could effectively extract the feature values of the data and significantly outperform the existing recommendation models.

INDEX TERMS Attention mechanism, collaborative filtering, recommender system.

I. INTRODUCTION

A recommender system is an advanced intelligent method based on machine learning for Big Data. Its main function is to make personalized recommendations according to user needs. It also plays a significant role in enabling feature learning of user preferences and item particulars. In the era of Big Data, users normally feel helpless when confronted with such a large amount of data. The recommender system can effectively improve processing efficiency by predicting ratings or other indicators to reflect user preferences. It aims to provide users with recommendations about items that people with similar tastes and preferences have liked in the past [1]. For instance, such a system can predict the degree of preference of a user for a new movie based on ratings of movies that the user has watched and, based on the predicted rating, decide whether to recommend the movie to the user. It is reported that 80% of movies watched on Netflix are suggested by their recommender system [2].

For such regression prediction problems, collaborative filtering (CF) is the key technique to build a personalized recommender system, which infers a user's preference not only from his behavior data, but also from those of other users [3]. CF mainly includes memory-based methods and model-based methods. The memory-based methods use the connections between users or items to make recommendations. More precisely, similar users or items are matched according to similarity measures. Model-based methods, on the other hand, use a machine learning algorithm to learn feature vectors and then obtain ratings for recommendation.

Many excellent methods used in recommender systems are based on CF techniques. CF analyzes the relationships between users and the interdependencies among items to identify new user-item associations [4]. It aims to use the similar preferences of people in their ratings history to predict what users might prefer. CF can explore the underlying reasons for the common features of different users in the ratings. The general approach is to represent the information of existing users and items as a one-hot vector and to send this vector into the deep learning framework to be trained [5]. Feasible conventional methods include LibFM [6], wide and deep learning [7], NCF [8], and others.

Model-based collaborative filtering has been widely used in recent studies. The method uses machine learning algorithms to train the embedding of users and items and then to build a model to predict the ratings of the users about new items. The method of predicting ratings based on matrix factorization (MF) is quite extensive [9]–[11]. In particular, He *et al.* [8] proposed a neural collaborative filtering (NCF) method to incorporate nonlinear functions into matrix factorization. This is a special method to integrate embedded features into MF and includes a multi-layer perceptron (MLP) to assign a nonlinear structure to the model. For the task of rating prediction on explicit feedback, it has been shown that MF model performance can be improved by incorporating user and item bias terms into the interaction function [12]. The generalized matrix factorization discovers the potential feature interaction between users and items, and the addition of nonlinearity can more accurately capture the complex interactions between users and items.

Most of the time, only one-hot vectors are used to behalf of the users and items. Upon training, the input data will become sparser and contain less information, which is inefficient for a training model that predicts ratings. To enhance accuracy, several recommendation techniques have been proposed that consider not only rating information, but also auxiliary information such as user demographics, social networks, and item description documents [13]. Some recent advances have applied CF to recommendation tasks and shown promising results. They used MF mostly to model auxiliary information, such as textual descriptions of items, audio features of music, and visual content of images [14]–[16]. Kim *et al.* [13] proposed a convolutional matrix factorization method for document context-aware recommendation (ConvMF), which adds document information on items to the probabilities matrix factorization (PMF) so that the document information can convey the features of items more accurately. To process the item documents, this paper uses a convolutional neural network (CNN), which is a state-of-the-art machine learning method that has shown high performance in various domains such as computer vision [17], natural language processing (NLP) [18]–[20], and information retrieval [21], [22]. The document can be transformed into a vector matrix and learned according to the principles of the CNN. The recommendation method based on document context-awareness with CNN processes the item documents as input and is effective in solving sparse data problems, creating a more sophisticated relationship between users and items and increasing the accuracy of ratings.

For the sake of accurate ratings prediction, a convolutional collaborative filtering method with attention mechanism was chosen in this work. An attention mechanism is a means of focusing on specific parts of the input, in other words, giving more weight to more valuable information, in the expectation that a higher attention value could make the model abstract more useful information. Moreover, the attention machine, as a method that copies human behaviors, should have different effects on different sites. Extensive experiments were conducted on the MovieLens and Amazon Instant Video datasets for personalized recommendations. Three methods were suggested to determine which part of the model should be combined with the attention mechanism. The results showed that it is most obvious to combine attention with the hidden layer to enhance prediction accuracy. Finally, several state-of-the-art recommender system methods, such as PMF, CTR, CDL, and ConvMF, were included to indicate the effect of the proposed method.

The main contributions of this research can be summarized as follows:

1. A CF framework called *convolutional collaborative filtering with attention* (Att-ConvCF) has been proposed for ratings prediction. The method combines traditional CF with an attention mechanism to assign proper weights to the prediction model and improve feature extraction.
2. Use of the attention mechanism in different parts of the proposed framework has been investigated. The best parts for more focused attention may vary according to the data characteristics. The attention mechanism focuses on the low-dimension model space for dense data and on the high dimension model space for sparse data.
3. Methods of combining different feature vectors are also discussed. Concatenation is suggested to merge miscellaneous feature vectors.

The rest of the paper is organized as follows. Section 2 briefly reviews the preliminaries of CF and of the representative matrix factorization, CNN, and attention mechanism methods. Section 3 describes the architecture of the proposed model and learning algorithm in detail. Section 4 introduces the experimental method, reports the experimental results, and presents discussions. Section 5 summarizes the conclusions and gives directions for future work.

II. BACKGROUND AND RELATED WORK

Recently, most state-of-the-art predictive rating recommendations have been based on CF techniques, which mainly include memory-based methods and model-based methods. The memory-based CF methods use a rating matrix and data on connections between users and items to recommend a new item that has not been rated by users. The model-based methods are more extensive [23], [24] and use machine learning to construct a recommendation model trained for predictive rating on the basis of the history data of users.

A. MATRIX FACTORIZATION

To use the information about users and items more efficiently as well as to calculate the interaction relationships between them, this study used MF to calculate second-order interaction features. MF can reduce the two-dimensional rating matrix $\mathbb{R}^{u*i}$ to the feature vectors of the user u and the item i. For instance, the basic MF model [25] maps the user and item into a joint latent space. As a recommender system method, MF has been successfully used in various applications. The latent factor model (LFM) [26] takes into account the impact of latent factors on the ratings and issues recommendations using the latent vectors of users and items. ConvMF [13] uses information about users and descriptive documents about items as latent vectors to predict ratings.

B. CONVOLUTIONAL NEURAL NETWORK

A convolutional neural network (CNN) is a typical category of multilayer feedforward neural network. Its goal is to combine lower-level features into high level representations following a given network architecture [27]. Two important structures in a CNN are the convolution layer and the pooling layer. The function of the convolution layer is to use the convolution kernel with the weights and a fixed scale to move in the feature matrix according to a fixed step size. In this way, the computational complexity can be reduced by sharing weights and features. In the pooling layer, the most commonly used method is the max-pooling method, which works to reduce the dimensionality of the data without changing the features. Because of these characteristics of CNN, it is good for solving problems that are location-invariant, where each feature is extracted in its input space and also depends on the compositional relationships between local and global features. This is the reason why CNNs have succeeded in computer vision [27]. Even though CNN was originally developed for that purpose [16], the key idea of CNN has been actively applied to information retrieval and NLP in contexts such as search query retrieval [21], [22], sentence modeling and classification [19], [20], and other traditional NLP tasks [18]. It can capture the compositional semantics of an entire sentence according to a set of features and compress these valuable semantics into feature maps [28]. For instance, in the field of dialogue topic tracking, CNN can be used to analyze the dialogue content. Reference [27] uses bag-of-words to form feature vectors from the dialogue content and uses CNN to extract the feature vectors. This is also a classic way to process documents. However, there are many alternative methods for document processing in practical applications. The choice among them depends on the actual situation in the dataset.

C. ATTENTION MECHANISM

The attention mechanism has been a popular model in recent years and has been widely used in many tasks such as recommendation, information retrieval, and computer vision [29]. The main principle is to imitate the attention mechanism of human beings. When our brain processes sensory signals, it will focus on certain representative areas. People have been found to identify things more quickly by using these areas. This rapid screening ability of humans demonstrates the efficiency and accuracy of the attention mechanism. The deep learning attention model can be regarded as a weight matrix with the same scale as the input data. In the beginning, each weight corresponds to an input value; then each of them is multiplied by the input value to give the result with attention. Finally, the degree of attention to the input data depends on the value of the weights. In addition, the weights are proportional to the degree of attention, meaning that a larger weight means stronger attention. In a nutshell, the key idea of attention is to learn how to assign attentive weights to a set of features: higher (lower) weights indicate that the corresponding features are more (less) informative for the task [4]. The attention mechanism has been applied in many recommendation tasks. Considering the interaction relationship of the inputs, the attention network is generated by the pair-wise interaction layer [29] and combined with each interaction vector. Moreover, in a recent issue of [30], Google has proposed a self-attention model that can replace CNN and RNN. It considers the parallelization of model training based on the original attention model and achieves good results in the field of machine translation. The attention mechanism is an excellent method for NLP that improves feature extraction efficiency.

III. The Att-ConvCF MODEL

To begin with, the overall structure of the Att-ConvCF will be introduced. Afterwards, the application and details of the methods, including MLP, CNN, the attention mechanism, and the related optimization method, will be introduced as well.

A. GENERAL FRAMEWORK

Fig. 1 shows the main components of the model. The one-hot vectors are input as user vectors along with the item vectors represented by the document information into the generalized matrix factorization (GMF). Concatenation is then used to combine the two feature vectors into MLP to predict $\hat{y}$. Note that the attention mechanism combines these three parts of the model as comparative experiments.

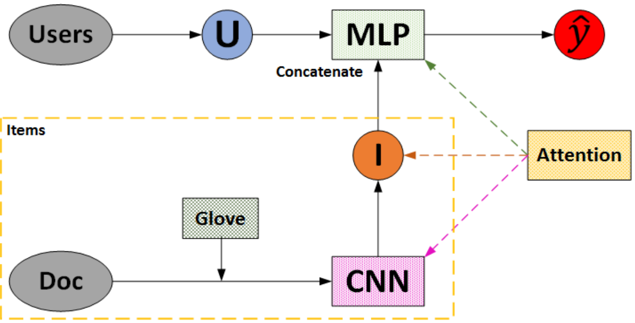

FIGURE 1. Overall architecture of Att-ConvCF.

Inspired by MF, it is possible to use the information on U and I obtained by the ratings matrix R to predict ratings ($U_i \in \mathbb{R}^{k*n}$, $I_r \in \mathbb{R}^{m*k}$ and $R \in \mathbb{R}^{m*n}$). In addition, the documents that describe the items can be used as input, and CNN can be used to process them into vectors. Due to the differences in how the user and item vectors are obtained, the two types of vectors are integrated into a new type by concatenating them to retain as much feature information as possible. The new vector is sent as input to the MLP for learning and predicting the ratings. Moreover, the attention mechanism is then introduced to optimize the model. To explain more clearly how the attention mechanism works, three methods were developed to integrate the attention model with the pooling layer, the fully connected layer, and the hidden layer. Once this has been done, the model can be used to predict ratings based on these three approaches to evaluate the most efficient way to use attention mechanisms.

B. CNN MODEL OF Att-ConvCF

The CNN structure is designed to process the documentary information about items, which is a critical step in embedding the words in the document. This study used a tool called the Glove [31], which can transfer words in the document to represent them in vector form at a fixed scale to pre-train the raw data to form a word vector matrix D^{a*l}, where a is the size of the embedding dimension for each word and l is the length of the document. Then CNN can be used to extract document features and to form a latent vector representation of the items. Using a one-dimensional convolutional kernel to process document tasks is common. As shown in Fig. 2, the word vector matrix is extracted by the one-dimensional convolution kernel $w \in \mathbb{R}^{a*h}$ at different scales in the convolution layer. The one-dimensional convolution kernel is an effective tool when convolutional neural networks process document tasks. Its size is consistent with the size of the embedding dimension. The size of the convolutional kernel window determines the number of words included in each step, and the number of steps determines the embedding dimension. The method of convolution layers for feature extraction can be represented as:

$$c_n^m = f\left(w \cdot x_{(n:n+h-1)} + b\right) \tag{1}$$

where n denotes the number of convolution operations, m denotes the number of convolutional kernels, and h denotes the size of the convolutional kernel window. In this study, ReLU was chosen as the activation function f because ReLU can avoid the vanishing gradient problem, which causes slow optimization convergence and may lead to a poor local minimum [32], [33]. The symbol "·" represents the dot product of the shared weights of the convolution kernel and the word vector. The notations w and b represent the weights and the bias.

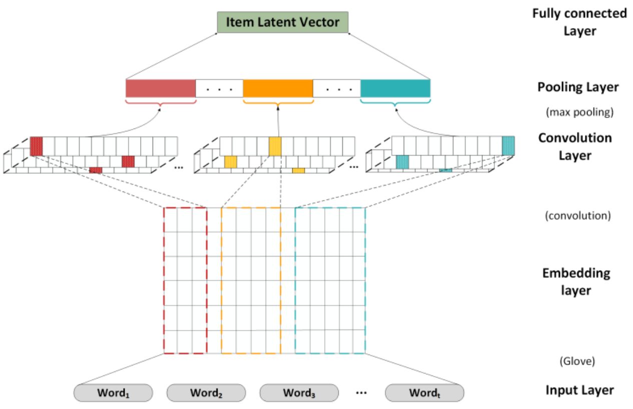

FIGURE 2. The CNN model in Att-ConvCF.

The function of the pooling layer is to reduce the dimension of the word vector further and to extract the feature values on the basis of the convolution operation. In particular, pooling enables the features to be translated without changing their properties. Pooling methods include max-pooling, average-pooling, stochastic-pooling, and others. This study has generally used the max-pooling method, and the features obtained by the convolution layer are further classified through sampling the pooling layer, which can prevent overfitting and enhance structural robustness. The max-pooling method can be expressed as:

$$p_v = max\,[c_n], \tag{2}$$

To generate the item embedding that matches the user embedding, the feature vector is further processed after the pooling layer. The size of the item latent vectors must match that of the user latent vectors so that they can complete the recommendation tasks together [5].

C. ATTENTION MODEL OF Att-ConvCF

Use of an attention mechanism improves the prediction precision of the model and is the main innovative contribution of this research. For a traditional attention mechanism, the attention weight matrix is generated by the encoder structure, and a new feature vector is calculated by multiplying the weight matrix by the corresponding vector. In Fig. 1, the attention mechanism is combined with MLP, the latent vector of items, and CNN respectively to test the prediction accuracy. The attention method in the model can be expressed as:

$$v' = z_{att} \odot v, \tag{3}$$

where the attention weight matrix z_{att} is derived from the corresponding feature vector v and v' denotes the new feature vector for attention. Otherwise, the attention mechanism and CNN are combined in the way that the attention matrix was combined with the pooling layer. In NLP, it has been demonstrated that the use of attention in different layers of CNN causes different effects. The attention mechanism in pooling has mainly been proposed in [34] and [35], but convolution was not affected. Subsequent experiments clearly showed that no matter which part of the prediction model was combined with attention, the predictive ability of the model was improved.

D. PREDICTION MODEL OF Att-ConvCF AND OPTIMIZATION ALGORITHM

According to the method proposed above, the feature vectors of users and items were obtained. To predict ratings, the two types of feature vectors must be merged into a new vector and fed into the hidden layers for learning. Based on this concept, the ratings prediction formula can be expressed as:

$$\hat{y}_{ir} = w_{ir} \sum_{i=1}^{N} \sum_{r=1}^{M} U_i \circ I_r + w_0, \tag{4}$$

where w_{ir} and w_0 denote the weights and the bias respectively, $\circ$ denotes the concatenation operation. To capture high order feature values and perform nonlinear computing to achieve the goal of predicting ratings, this study used MLP to evaluate user-item feature vectors, which form a deep neural network with multiple hidden layers, where the neurons between each adjacent layer are fully connected and given a nonlinear activation function for calculation. Formally, the definition of the hidden layers is:

$$\begin{aligned} L_1 &= \sigma_1 \left(w_1 f_{(U \circ I)} + b_1\right) \\ L_2 &= \sigma_2 \left(w_2 L_1 + b_2\right) \\ &\vdots \\ L_j &= \sigma_j \left(w_j L_{j-1} + b_j\right), \end{aligned} \tag{5}$$

where j denotes the number of layers in the MLP and σ_j, w_j, and b_j denote the activation function, weights, and bias for layer j respectively. In general, the number of hidden layers is a crucial parameter for the architecture of multilayer neural networks [36]. However, the number of layers is limited, and blindly increasing them cannot improve the precision of the model indefinitely.

To improve prediction accuracy and efficiency further, certain optimization algorithms were used to perfect the model. The stochastic gradient descent (SGD) method is a general optimization algorithm for neural networks in the deep learning framework, which can update the weights in the model iteratively and obtain the gradient through randomly selected data to update the weights w. Because this method is mostly used in experiments with large amounts of data, it can improve the efficiency of experiments:

$$J(\theta) = \frac{1}{2m} \sum_{j=0}^{m} \left(\hat{y}_{ir}^{(j)} - y_{ir}^{(j)}\right)^2, \tag{6}$$

where $J(\theta)$ denotes the loss function and is used to update the weight values as follows:

$$\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta), \tag{7}$$

where α denotes the learning rate, which determines the step size of the gradient descent. If the value is too small, the rate of function minimization will be affected. By contrast, if the value is too large, the function will overshoot the minimum. In addition, dropout [37], which is a neural network regularization technique to prevent overfitting in the hidden layer while optimizing the model, was used. Specifically, during model training, the neural network in each layer randomly dropped neurons in a certain proportion. Moreover, the dropout parameters were set so that they could be updated, thereby effectively preventing co-adaptation of neurons.

When using the SGD method to train the model, the importance of avoiding vanishing and exploding gradients in the backpropagation should be emphasized. These models contain many fully connected neural network frameworks in the hidden layers, and therefore if the weights are mostly less than 1 in the case of relatively deep network layers, the problem of vanishing gradients will appear in the later stage of calculations. Conversely, if the weight values are greater than 1, the exploding gradient problem can easily occur. In general, all these problems will seriously affect model training. Therefore, batch normalization (BN) was used in this study to solve the problems caused by uncontrollable weight scales. Essentially, the function of BN is to standardize the input data of each layer:

$$L_{j(BN)} = \gamma \odot \left(\frac{L_j - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}}\right) + \beta, \tag{8}$$

where the average of the batch $\mu_\beta = \frac{1}{k} \sum_{i=0}^{k} wx_i$ is subtracted from the original input and the result is then divided by the standard deviation σ_B^2. To prevent the divisor from being zero, a tiny positive number ε is added. γ also serves to adjust the size of the value, and β can shift the normalization value.

IV. EXPERIMENTS

This section first introduces the datasets, evaluation methods, and parameters used in the experiments. Then the results of the experiments are introduced and analyzed from three aspects: 1) adding an attention mechanism can improve the stability and prediction accuracy of the model, but the combination of attention mechanisms is different with different types of datasets; 2) according to the experimental results, combining feature vectors using the concatenation method is better than using the multiplication method; and 3) the Att-ConvCF model proposed here outperforms other state-of-the-art methods.

A. EXPERIMENTAL SETTING

We use three datasets to run in a deep learning environment. And four state-of-the-art methods are used for comparative experiments.

1) DATASET AND EVALUATION METHOD

To demonstrate that the proposed model can effectively predict ratings, three real-world datasets were used to validate the model accuracy: MovieLens-1m (ML-1m), MovieLens-10m (ML-10m), and Amazon Instant Video (AIV). Among these, the MovieLens datasets are commonly used in the field of prediction ratings. They are the full version of the latest MovieLens data published by GroupLens [38]. The ratings of users for movies are contained in these datasets, where users rate movies explicitly on a scale of 1 to 5. However, the MovieLens datasets do not contain descriptions of the movies, and therefore associated documents from the IMDB were used to provide descriptive information about the items.

Reviews of movies are contained as item description information in the AIV. Table 1 lists in detail the statistical information for these three datasets.

TABLE 1. Data statistics on three real-world datasets.

Dataset	#Users	#Items	#Ratings
ML-1m	6,040	3,544	993,482
ML-10m	69,878	10,073	9,945,875
AIV	29,757	15,149	135,188

The dataset preprocessing procedure before the experiment removed users with fewer ratings and item descriptive information that did not match. Moreover, corpus-specific stop words, which occurred at a frequency higher than 0.5 in the documents, were removed, and the maximum number of words per document was set to 300. As showed in Table 1, the number of ML-10m ratings was much greater than for ML-1m, but the density of ML-10m was sparser than that of ML-1m. The numbers of items in AIV were greater than in ML-1m, but AIV contains relatively few ratings data. Therefore, AIV was a sparser dataset for these purposes. To evaluate the predictive accuracy of the model on real-world datasets, the datasets were divided into 80% as a training set, 10% as a validation set, and 10% as a test set after shuffling the order. We processed the data in this way before each experiment. The metrics of the predicted ratings of the model were evaluated by RMSE, which is directly related to an objective function of conventional rating prediction model:

$$RMSE = \sqrt{\frac{\sum_{i,r=1}^{T}\left(y_{ir} - \hat{y}_{ir}\right)^2}{|T|}}, \tag{9}$$

where T denotes the total number of ratings used for training and y_{ir} denotes the real ratings.

2) IMPLEMENTATION DETAILS

The procedure proposed by Keras [39] was used for the implementation and experimental framework, and a GeForce GTX 1080 GPU was used for the computations. Glove was used to generate a 100-dimensional embedding for each word. In this way, a word vector matrix $D \in \mathbb{R}^{a*b}$ ($a = 100$, $b = 300$) was obtained for the CNN model. According to previous experimental experience, Glove was used to preprocess the AIV description document into 300-dimensional word vectors. In the convolution layer, each of the feature vectors was extracted as a 100-word vector matrix and one-dimensional convolution kernels with window sizes of 3, 4, and 5. Different sizes of windows can extract the features from different aspects of the word vector matrix and improve prediction accuracy. The concatenation method was used to merge the latent vectors of users and items, taking into account that the inputs of users and items are different properties of the data and that therefore the feature values of the users and items retain their maximum extent after concatenation.

The attention mechanism was implemented to integrate three parts of the model: the hidden layer, the fully connected layer, and the CNN pooling layer. Among these, the attention vector of the pooling layer was obtained by reshaping the matrix from the convolution layer. The dropout procedure and a learning rate of 0.2 and batch normalization were implemented to prevent overfitting of the model.

3) BASELINES

Three Att-ConvCF models were used to compare the following baselines:

PMF: probabilistic matrix factorization [24] is a standard collaborative filtering predictive ratings model using only user ratings.

CDL: collaborative deep learning [15] uses auto-encoders and PMF to predict ratings and improves ratings prediction accuracy by analyzing documents using SDAE.

CTR: collaborative topic regression [40] is a state-of-the-art recommendation model. It combines PMF and latent Dirichlet allocation (LDA) to predict whether a user is interested.

ConvMF: convolutional matrix factorization [13] is a recent representative recommendation model that combines PMF and CNN methods to predict ratings by using document information for items.

Among them, the most competitive method is ConvMF, which has a similar structure to this paper. But it adopted different collaborative filtering methods.

B. EXPERIMENTAL RESULTS

1) EFFECT OF THE ATTENTION MECHANISM ON MODEL PREDICTION RESULTS

As a method of simulating human behavior, the attention mechanism is applied to the field of deep learning, which is more and more popular. This experiment was intended to verify whether the attention mechanism has a positive impact on prediction accuracy. First of all, the ML-1m and AIV datasets were used to compare the models with and without the attention mechanism; the results are shown in Fig. 3. The results from both datasets reflect that the attention mechanism as an important auxiliary method can improve feature extraction by the model and obtain better performance. Because the original model cannot effectively extract features, an appropriate attention mechanism will play an active promoting role.

The advantage of the attention mechanism can make the model achieve the best prediction accuracy quickly and even improve the prediction accuracy effectively. In the sparser AIV dataset, the model that combined an attention mechanism with the latent vector scored 2.87% lower in the RMSE metrics than the model without the attention mechanism. This shows that it is a good approach to use an attention mechanism to process sparser data. Moreover, Fig. 3 illustrates that the model with an attention mechanism has higher stability.

According to the principle of the attention mechanism, efforts were made to merge it into different parts of the model

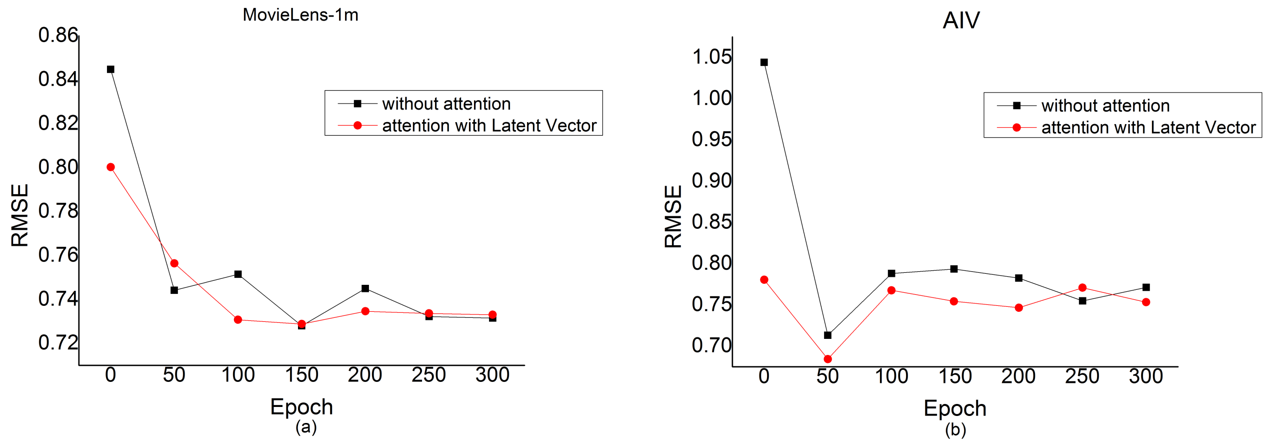

FIGURE 3. RMSE values of validation without attention and with attention in fully connected layer.

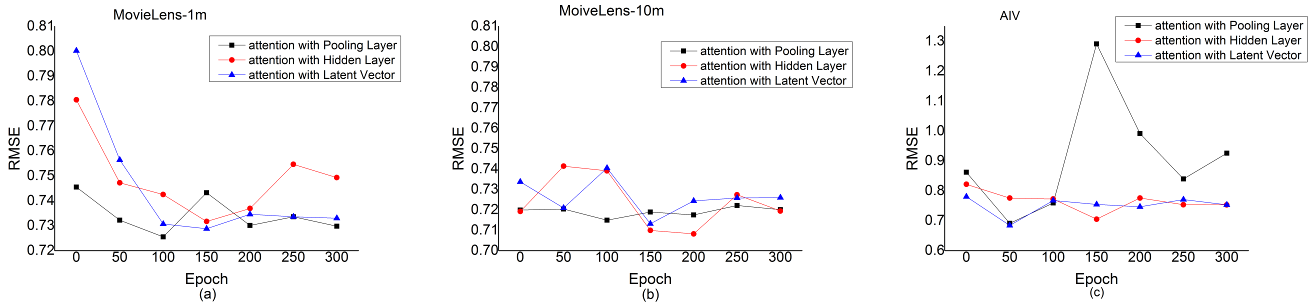

FIGURE 4. RMSE values of validation of three methods to combine attention.

so as to compare the effects of different parts of the attention mechanism on model prediction accuracy. Table 2 shows the results of experiments on the three datasets. The table shows the best RMSE values for each of the three parts of the model where the attention mechanism was implemented for 300 epochs. In the ML-1m dataset, the model that combined the attention mechanism with the pooling layer performed the best in prediction accuracy. However, there was only a slight difference in prediction accuracy between this model and the model combining the item latent vector with the attention mechanism. On the ML-10m dataset, the prediction accuracy of all models was improved. Unlike the results on the ML-1m dataset, the best prediction accuracy on the ML-10m dataset was achieved by the model combining the hidden layer with the attention mechanism. For the sparsest AIV dataset, the three methods all performed well in prediction accuracy, and the results were very close.

TABLE 2. RMSE values for validation of each attention method model.

Model	Dataset		
	ML-1m	*ML-10m*	*AIV*
Att-ConvCF(Hidden)	0.7317	**0.7085**	0.7051
Att-ConvCF(Latent)	0.7288	0.7132	**0.6843**
Att-ConvCF(Pooling)	**0.7255**	0.7149	0.6914

However, as shown in Fig. 4, the model combining the attention mechanism with the pooling layer seemed to show an overfitting phenomenon on the AIV during the training procedure. The main function of the attention mechanism is to help the model assign higher weight to valuable features, so that when data are sparse, effective feature extraction is particularly important. Combining the results from the ML-10m dataset, which was also sparse, the model prediction accuracy was higher when the attention mechanism was combined with the hidden layer that contained more feature information (including both user and item features). In contrast, the model was able to extract feature vectors from a plentiful description document for the dense ML-1m dataset. No matter what combination of attention mechanism is used, an attention mechanism is good at processing Big Data, as shown by Fig. 4. On the ML-10m dataset, the evaluation results were more stable, and no overfitting occurred.

2) COMPARING THE EFFECTS OF THE MULTIPLICATION AND CONCATENATION METHODS

In a multiple-layer deep learning neural network, a small change can affect the overall network outcome. It was noted from the experimental results that the input user vectors included only label information, whereas the input item vectors were formed from the descriptive documents. The user and item vectors need to be merged into a multiple-layer neural network to train the model. But these two types of input are not the same in the nature of the datasets. Keras provided several different methods for merging these two types of vectors. Two commonly used methods, multiplication and concatenation, were used to perform the comparisons. The multiplication method merges two types of vectors of the same dimension into a new vector by one-to-one multiplication. The concatenation method merges two types of vectors into a new vector by connecting them end to end. The best-performing model on the ML-1m dataset was selected and used with its structure and parameters remaining unchanged, and the two methods described above were used for comparison.

Fig. 5 shows the RMSE values obtained by the two methods during 300 epochs on the test set. It can be seen intuitively from the figure that the concatenation method is superior to the multiplication method in prediction accuracy. Moreover, the model using the concatenation method is more stable in prediction ability and more generalized during the 300-epoch training process. The experimental results show that the concatenation method is better suited to the proposed model. It can retain the maximum amount of the information contained in the two feature vectors.

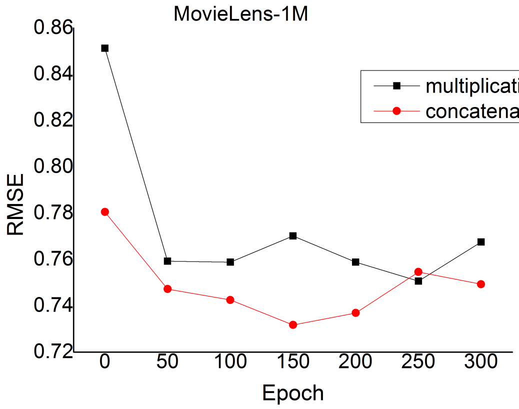

FIGURE 5. RMSE values for validation of the multiplication and concatenation methods.

3) COMPARATIVE EXPERIMENT ON PREDICTION ACCURACY

In this study, four representative experiments were performed to compare the proposed model with ConvMF, which is the most competitive method. As shown in Table 3, all values of RMSE on the test set were enumerated for the five models, including Att-ConvCF. Based on the datasets mentioned above, the models with the highest prediction accuracy on the three datasets were selected to compare with the four other models. It can be intuitively observed that the RMSE of the Att-ConvCF test set was superior to the other models.

TABLE 3. Overall test RMSE.

Dataset	Model					Improvement
	PMF	CDL	CTR	ConvMF	Att-ConvCF	
ML-1m	0.8971	0.8879	0.8969	0.8549	**0.7402**	11.47%
ML-10m	0.8311	0.8186	0.8275	0.7930	**0.7601**	3.29%
AIV	1.4118	1.3594	1.5496	1.1279	**0.7752**	35.27%

The ConvMF model, which was the best competitor, uses a similar attention mechanism to Att-ConvCF. In particular, it seamlessly integrates CNN into PMF, and user labels and item documents are used for ratings prediction. On the ML-1m dataset, the improvement of Att-ConvCF over ConvMF was 11.47%, which is a very significant improvement. Compared with CDL, which also uses collaborative filtering, the model proposed here used an attention mechanism, which is a more effective method for feature extraction, and the prediction accuracy was greatly improved. On the ML-10m dataset the proposed model achieved only a 3.29% improvement compared to ConvMF, the reason that massive datasets are more conducive to feature extraction for models. However, in comparison, Att-ConvCF has a higher predictive ability. This also indicates that the proposed model is more effective in processing sparse datasets. Compared with CTR on the ML-10m, which also uses information from descriptive documents, the proposed model was improved by 6.74%, which was mainly due to data preprocessing. Att-ConvCF, unlike ConvMF, uses Glove as its word embedding model. It is well-known that Glove can transform words into alternative vectors with different dimensions. In this study, the appropriate dimensions for the Glove embedding method in the preprocessing procedure were selected by experimental comparison. This is in accord with the principle of multiple-layer neural networks and provides enough feature values for deep learning to predict ratings. On the AIV dataset, Att-ConvCF showed the ability to process relatively sparse data. The improvement of Att-ConvCF over ConvMF was 35.27%, which was a great improvement. Moreover, it was demonstrated once again that Att-ConvCF can process sparser data than other models.

V. CONCLUSIONS

This paper has proposed a novel deep learning model for recommender systems, called Att-ConvCF. The main idea was to combine an attention mechanism into a collaborative filtering recommendation model. Descriptive document information was used for feature extraction instead of the sparse input vector of items, and the matrix formed by word embedding was included into CNN by means of the attention mechanism. The new vectors formed by concatenating the feature vectors of users and items were sent to the nonlinear hidden layer to predict ratings.

This study has demonstrated through extensive experiments that adding an attention mechanism improves the capability of the model. The effect of integrating the attention mechanism with different parts of the model has also been discussed and evaluated by predictive results. This study is a useful attempt to understand attention mechanisms more clearly. It has also been demonstrated that Att-ConvCF is generally superior to other models compared with control experiments, which shows that the attention mechanism plays an important role and also that using concatenation to process feature vectors in the proposed model is a notably successful approach.

In future research, the authors will continue to consider integration of the attention mechanism with the recommender system. It is well-known that Google published a paper referred to here as [30], which introduced a self-attention mechanism and achieved good results in NLP. Efforts will also be made to use the self-attention model instead of the CNN to process descriptive document information. In these experiments, adding an attention mechanism to user input data was not considered because the user input only contained the label, which provides very limited information. Rather, enriching the original user data to provide more features was explored. In this way, the issue of using an attention mechanism can be considered in terms of user data, which is another future project. At the same time, it can be expected that the self-attention model can improve feature extraction capability and achieve more accurate recommendations.

REFERENCES

[1] A. Gupta and B. K. Tripathy, "A generic hybrid recommender system based on neural networks," in *Proc. IEEE IACC*, Gurgaon, India, Feb. 2014, pp. 1248–1252.

[2] F. Strub, J. Mary, and R. Gaudel, "Hybrid collaborative filtering with neural networks," presented at the ICML, New York, NY, USA, Jun. 2016.

[3] X. He *et al.*, "Outer product-based neural collaborative filtering," in *Proc. 27th IJCAI*, Melbourne, VIC, Australia, 2018, pp. 2227–2233.

[4] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T.-S. Chua, "Attentive collaborative filtering: Multimedia recommendation with item- and component-level attention," in *Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Tokyo, Japan, 2017, pp. 335–344.

[5] X. He and T.-S. Chua, "Neural factorization machines for sparse predictive analytics," in *Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Tokyo, Japan, 2017, pp. 355–364.

[6] S. Rendle, "Factorization machines with LibFM," *ACM Trans. Intell. Syst. Technol.*, vol. 3, no. 3, pp. 1–57, May 2012, doi: 10.1145/2168752.2168771.

[7] H.-T. Cheng *et al.*, "Wide & deep learning for recommender systems," in *Proc. DLRS*, Boston, MA, USA, 2016, pp. 7–10.

[8] X. He *et al.*, "Neural collaborative filtering," in *Proc. Int. WWW Conf. Committee*, Perth, WA, Australia, 2017, pp. 173–182

[9] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proc. UAI*, Montreal, QC, Canada, 2009, pp. 452–461.

[10] X. He, H. Zhang, M.-Y. Kan, and T.-S. Chua, "Fast matrix factorization for online recommendation with implicit feedback," in *Proc. 39th Int. ACM SIGIR*, Pisa, Italy, 2016, pp. 549–558.

[11] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T.-S. Chua, "Discrete collaborative filtering," in *Proc. 39th Int. ACM SIGIR*, Pisa, Italy, 2016, pp. 325–334.

[12] A. J. Smola. (2012). *Recommender Systems Lecture*. [Online]. Available: http:// alex.smola.org/ teaching/berkeley2012/slides/8_Recommender.pdf

[13] D. Kim, C. Park, J. Oh, S. Lee, and H. Yu, "Convolutional matrix factorization for document context-aware recommendation," in *Proc. ACM Conf. Recommender Syst.*, Boston, MA, USA, 2016, pp. 233–240.

[14] A. van den Oord, S. Dieleman, and B. Schrauwen, "Deep content-based music recommendation," in *Proc. NIPS*, Lake Tahoe, NV, USA, 2013, pp. 2643–2651.

[15] H. Wang, N. Wang, and D.-Y. Yeung, "Collaborative deep learning for recommender systems," in *Proc. KDD*, 2015, pp. 1235–1244.

[16] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma, "Collaborative knowledge base embedding for recommender systems," in *Proc. KDD*, San Francisco, CA, USA, 2016, pp. 353–362.

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 306–351, Dec. 1998, doi: 10.1109/5.726791.

[18] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *J. Mach. Learn. Res.*, vol. 12 pp. 2493–2537, Aug. 2011.

[19] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," in *Proc. 52nd Annu. Meeting ACL*, vol. 1, 2014, pp. 655–665.

[20] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. EMNLP*, 2014, pp. 1746–1751.

[21] J. Gao *et al.*, "Modeling interestingness with deep neural networks," in *Proc. EMNLP*, 2014, pp. 1–12.

[22] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil, "A latent semantic model with convolutional-pooling structure for information retrieval," in *Proc. 23rd ACM Int. Conf. Conf. Inf. Knowl. Manage. (CIKM)*, 2014, pp. 101–110.

[23] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, Las Vegas, NV, USA, 2008, pp. 426–434.

[24] R. Salakhutdinov and A. Mnih, "Probabilistic matrix factorization," in *Proc. NIPS*, vol. 20, no. 2, 2007, pp. 1257–1264.

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

[26] Y. Koren, "Factor in the neighbors: Scalable and accurate collaborative filtering," *ACM Trans. Knowl. Discovery Data*, vol. 4, no. 1, pp. 1–24, Jan. 2010.

[27] S. Kim, R. Banchs, and H. Li, "Exploring convolutional and recurrent neural networks in sequential labelling for dialogue topic tracking," in *Proc. Annu. Meeting ACL*, Berlin, Germany, 2016, pp. 963–973.

[28] Y. Chen *et al.*, "Event extraction via dynamic multi-pooling convolutional neural networks," in *Proc. Annu. Meeting ACL*, Beijing, China, vol. 1, 2015, pp. 167–176.

[29] J. Xiao *et al.*, "Attentional factorization machines: Learning the weight of feature interactions via attention networks," in *Proc. IJCAI*, Melbourne, VIC, Australia, 2017, pp. 3119–3125.

[30] A. Vaswani *et al.*, "Attention is all you need," in *Proc. 31st Conf. NIPS*, Long Beach, CA, USA, 2017, pp. 5998–6008.

[31] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proc. EMNLP*, Doha, Qatar, 2014, pp. 1532–1543.

[32] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. 30th ICML*, Beijing, China, vol. 28, 2013, pp. 1–6.

[33] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," *J. Mach. Learn. Res.*, vol. 15, no. 4, pp. 315–323, 2011.

[34] W. Yin and H. Schütze, "MultiGranCNN: An architecture for general matching of text chunks on multiple levels of granularity," in *Proc. ACL*, Beijing, China, 2015, pp. 63–73.

[35] C. D. Santos, M. Tan, B. Xiang, and B. Zhou, "Attentive pooling networks," *CoRR*, vol. abs/1602.03609, pp. 1–10, Feb. 2016.

[36] G. Brightwell, C. Kenyon, and H. Paugam-Moisy, "Multilayer neural networks," *J. Indian Inst. Sci.*, vol. 1, no. 1, pp. 192–194, 2003.

[37] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[38] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, p. 19, Jan. 2016.

[39] F. Chollet. (2015). *Keras*. [Online]. Available: https://github.com/fchollet/keras

[40] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2011, pp. 448–456.

BANGZUO ZHANG was born in 1971. He received the bachelor's degree in computer science from Northeast Normal University, in 1995, and the master's and Ph.D. degrees in computer application technology from Jilin University, in 2002 and 2009, respectively.

He has been a Visiting Scholar with the University of Illinois at Chicago, Chicago, from 2011 to 2012. He is currently an Associate Professor with the School of Information Science and Technology, Northeast Normal University. His current research interests include information retrieval, recommender systems, machine learning, and data mining.

HAOBO ZHANG received the B.Eng. degree in computer science and technology from the Jilin University of Finance and Economics University, Changchun, China, in 2016.

He is currently pursuing the master's degree with Northeast Normal University, Changchun. His current research interests include recommender system, machine learning, and data mining.

XIAOXIN SUN received the B.S. degree in computer science and technology and the M.S. degree in computer application technology from Northeast Normal University, Changchun, China, in 2002 and 2005, respectively.

He is currently pursuing the Ph.D. degree with Northeast Normal University. He is currently a Lecturer with Northeast Normal University. His current research interests include recommender system and deep learning.

GUOZHONG FENG was born in Kunshan, Jiangsu, China, in 1982. He received the bachelor's degree in mathematics and applied mathematics and the Ph.D. degree in probability theory and mathematical statistics from Northeast Normal University, Changchun, in 2004 and 2011, respectively.

From 2012 to 2015, he was a Postdoctoral Research Fellow with the Key Laboratory of Applied Statistics of MOE, Northeast Normal University. He is currently an Associate Professor with Northeast Normal University. His research interests include text mining and statistical machine learning.

CHUNGUANG HE was born in 1972. He received the bachelor's, master's, and Ph.D. degrees in environmental science from Northeast Normal University in 1996, 1999, and 2008, respectively.

He is currently a Professor with the School of Environment, Northeast Normal University. His current research interests include wetland ecology and ecological restoration of water pollution.