A Probabilistic Matrix Factorization Recommendation Method Based on Deep Learning

To cite this article: Xiaoyue Gong and Xiaojun Huang 2019 J. Phys.: Conf. Ser. 1176 022043

View the article online for updates and enhancements.
A Probabilistic Matrix Factorization Recommendation Method Based on Deep Learning

Xiaoyue Gong¹,*, Xiaojun Huang²
¹Beijing University of Posts and Telecommunications, Beijing, China
²Beijing University of Posts and Telecommunications, Beijing, China

*Corresponding author e-mail: gongxiaoyue@bupt.edu.cn, hxj@bupt.edu.cn

Abstract. In order to improve the accuracy of recommendation, a probabilistic matrix factorization recommendation method based on deep learning (PMFDL) is proposed. The method considers the influence of context information on the implicit feature of items and the influence of time factor on the implicit feature of users. In this paper, a convolutional neural network with attention mechanism is introduced to learn the implicit feature of items, and a long-term and short-term memory network is introduced to learn the implicit feature of users. Finally, we combine probabilistic matrix factorization (PMF) to predict recommendation results. After experimental verification, the experimental results show that the proposed PMFDL method is superior to Probabilistic Matrix Factorization (PMF) and Convolutional Matrix Factorization (ConvMF) in recommendation accuracy.

1. Introduction

With the advent of the big data era, the explosive growth of online information has brought huge troubles to users. Users cannot quickly find the information that they need from massive data. Recommendation system is an effective tool to guide users to discover the information they are interested in. It can provide personalized services for different users, and solve the problem of information overload. Collaborative filtering recommendation is one of the most researched and mature recommended methods. The recommended method of collaborative filtering are mainly divided into neighborhood-based method and model-based method [1]. Compared with neighborhood-based method, the model-based recommendation method such as singular value decomposition (SVD) and PMF can achieve better recommendation results [2]. PMF introduces probability factor on the basis of SVD and improves the recommendation effect. However, the PMF can only use the user-item rating matrix to learn the implicit feature of users and items, and cannot effectively use auxiliary information such as the description information of items and the information of users.

With the arrival of the deep learning boom, deep learning has been applied to the recommendation system by more and more researchers. Although these recommendation system has made up for the shortcomings of traditional recommendation algorithm that cannot effectively use auxiliary information, but there are still some defects. In [3], two parallel convolutional neural networks (CNN) were used to learn the implicit feature of users and the implicit feature of items based on the evaluation information of users. Finally, the two networks were combined through Factorization machines. The impact of time factor on user’s hobbies has not been considered in the model. A hierarchical Bayesian Model (CDL) is proposed in [4]. It combined Stacked Denoising Autoencoders and Latent Factor Model to effectively learn auxiliary information and improved recommend effect.
However, the contextual information is not considered, and the semantics of the content cannot be extracted well. Literature [5, 6] combined CNN and PMF, and used CNN to learn the implicit feature of the items according to the description information of items, but the implicit feature of users cannot be effectively learned. [7] combined CNN and Latent Factor Model, used CNN to extract the implicit feature of items from the text description of learning resources. The model optimized the implicit feature of the items, but didn’t optimize the implicit feature of users. [8] considered the time sequence of user’s browsing history, used the Recurrent Neural Network (RNN) to learn the implicit feature of users, and used the Latent Factor Model to recommend news messages. However, the output dimension was the same as the input dimension in the feature learning of news messages, dimensionality optimization can be performed. A collaborative filtering algorithm based on RNN is proposed in [9], which considered the timing of user consumption when extracting the implicit feature of users. The model did not make full use of the implicit feature of items which are the basis for obtaining the implicit feature of users.

In order to effectively obtain the implicit feature of items and understand the real needs of users, we develop a Probabilistic Matrix Factorization method based on Deep Learning (PMFDL). This method is mainly applied to the recommendation of text description scenario. The method is mainly divided into three parts: learning the implicit feature of items, learning the implicit feature of users and combine PMF to train the implicit feature of items and users. To accurately learn the implicit feature of items, this paper adds the attention mechanism [10, 11] to CNN, focuses user’s attention on a few important feature of items, and filters the interference of useless information of items. In order to describe the real needs of users, this paper adopts LSTM[12, 13] to learn the implicit feature of users. The influence of time factor on user’s hobbies and the connection between historical items are considered in the learning of user’s implicit feature. Finally, PMF is used to linearly combine the implicit feature of the items with the implicit feature of the users to predict the user’s preference for the item. In order to verify the performance of the proposed algorithm, PMFDL is compared with PMF[14] and ConvMF[6] on the ML-1 and AIV dataset. The experimental results show that our method is superior to PMF and ConvMF in recommendation accuracy.

2. PMFDL Model

PMF is a probabilistic solution for the matrix factorization model. It maps some potential feature information of users and items to low-dimensional mapping space from the perspective of probability, and then uses a linear combination of low-dimensional feature vector to explain the user’s preference for the item. The main task of PMF is to obtain the implicit feature vector of users and the implicit fea-

![Figure 1. PMFDL model](image-url)
ture vector of items through continuous iterative updating. This paper attempts to improve the acquisition of use’s implicit feature vector and item’s implicit feature vector, and proposes the PMFDL algorithm. The model of PMFDL is shown in Fig.1.

PMFDL uses Attention-CNN to obtain the implicit feature vector of items. Attention-CNN network introduces attention mechanism on the basis of CNN. In recent years, the application of CNN in natural language processing has achieved good results, such as emotional analysis and text classification. CNN can effectively extract potential feature based on global information [3]-[6]. According to the human visual attention mechanism, users only pay attention to one aspect or some aspects of an item. The attention mechanism in deep learning can well interpret this feature. The combination of CNN and attention mechanism can make use of the attention mechanism to filter useless information on the basis of CNN’s extraction of global information, so as to obtain the implicit feature of items better. PMFDL uses LSTM to learn the implicit feature vector of users. LSTM has the memory function, which can remember the user’s past and present interests, discover the implicit connection between items in the historical record, and guarantee the timeliness of user’s hobbies. The internal structure of Attention-CNN and LSTM is described in detail below.

2.1. Attention-CNN Network Structure

The network structure of Attention-CNN is shown in Fig.2. The main function of the structure is to obtain the implicit feature vector of items according to its text description information. The input of Attention-CNN model uses digital quantity. We need to convert the description information of items into digital quantity. Because word embedding can represent complex contexts and dimensions are controllable, this paper uses the word embedding method.

![Figure 2. Attention-CNN Network Structure](image_url)

Language Model: The input of this layer is the text description information of item. The word2vec tool is used to convert each word in the text into a low-dimensional numeric vector representation. We assume that the maximum length of description information is \( d \). The output of the layer is \( X \in \mathbb{R}^{k \times d} \).

\[
X = \begin{bmatrix}
    x_1 & x_2 & \cdots & x_d
\end{bmatrix}
\]

(1)

Where \( k \) is the dimension of each word vector, \( x_i \in \mathbb{R}^k \).

Convolution layer: The main task of this layer is to extract the feature information of text. It uses
multiple convolution kernels with different sharing weights to obtain text feature. We assume that the
window length of the convolution kernel is \( l \), the shared weight matrix can be expressed as \( W^j \in \mathbb{R}^{l \times dl} \).

The features obtained by convolution kernel \( j \) can be expressed as:

\[
t^j = \left[ t'_1, t'_2, \ldots, t'_l, \ldots, t'_{l+1} \right]
\]

(2)

Where \( t'_i = f \left( W^j \otimes X_{(i+l-1)} + b' \right) \), \( t'_i \) represents the eigenvalue of region \( i \) obtained by convolution
kernel \( j \). \( \otimes \) represents convolution operator. \( f(\bullet) \) represents non-linear activation function, and ReLU
activation function is adopted in this paper. \( W^j \) is the mapping matrix, \( X_{(i+l-1)} \) is the \( i \) column to
\( i+l-1 \) column of \( X \), and \( b' \) is a bias.

The output of this layer is:

\[
T = \left[ t'_1, t'_2, \ldots, t'_n \right]
\]

(3)

Where \( n \) is the number of convolution kernel.

**Pooling layer:** This layer uses the overall characteristics of the adjacent output at a location to
replace the output of that location in the network. The output of the layer is:

\[
P = \left[ p_1, p_2, \ldots, p_n \right]
\]

(4)

Where \( p_i = \max \left( t'_i \right) \), \( \max(\bullet) \) is the maximum in \( t'_i \).

**Attention:** Attention mechanism are introduced in this layer. The information weight of each word
is calculated according to the attention matrix. It highlights some important features or ignores useless
information based on the size of the weight value. Calculating the corresponding weight of each word
according to (5).

\[
\alpha_i = g \left( W^{\text{attn}} \ast \left( P^{\text{attn}} \right)^\top + b^{\text{attn}} \right)
\]

(5)

Where \( P^{\text{attn}} = \left[ p_{(i-m-1)}, p_{(i-m)}, \ldots, p_i, \ldots, p_{(i+m-1)} \right] \), \( m \) is the size of the sliding window. \( W^{\text{attn}} \in \mathbb{R}^{1 \times m} \), \( b^{\text{attn}} \in \mathbb{R} \).

\( g(\bullet) \) is a non-linear activation function.

The output of this layer is

\[
\Phi = [\omega_1, \omega_2, \omega_3, \ldots, \omega_k]
\]

(6)

Where \( \omega_i = \alpha_i p_i \).

**Output layer:** Converting the high-dimensional features of the previous layer into a feature vector
of specific dimension as an implicit feature representation of the items through a nonlinear function. The output of the output layer is

\[
v_j = \tanh \left( W^o \Phi + b^o \right)
\]

(7)

Where \( W^o \in \mathbb{R}^{D \times n} \) is the mapping matrix, \( b^o \in \mathbb{R} \) is a bias, \( v_j \in \mathbb{R}^o \) is the implicit feature vector of item
\( j \).

2.2. **LSTM Network Structure**

In this paper, the implicit feature vector of users are obtained according to their historical records.
The chronological order of the items in the history record needs to be considered during the
acquisition process. Therefore, the learning of user’s implicit feature vector can be seen as a time
series problem. Because LSTM can handle sequential problems very well. Thus, this paper uses
LSTM to obtain the implicit feature vector of users. Its structure is shown in Fig.3.

The input sequence of LSTM is the implicit feature vector of historical items in chronological order,
and the output is the implicit feature vector of user.

\[
\begin{align*}
C_t &= \sigma \left( W_c \left[ h_{t-1}, v'_t \right] + h_t \right) \ast \tanh \left( W_s \left[ h_{t-1}, v'_t \right] + h_t \right) + \sigma \left( W_s \left[ h_{t-1}, v'_t \right] + h_t \right) \ast C_{t-1} \\
&= \sigma \left( W_c \left[ h_{t-1}, v'_t \right] + h_t \right) \ast \tanh \left( C_t \right) \\
&= h_t \\
v'_t &= h_t \end{align*}
\]

(8)
Where $W_i$, $W_j$, $W_k$, $W_l$ is the mapping matrix, $b_i$, $b_j$, $b_k$, $b_l$ is a bias, $C_i$ is the cell state at time $t$. $y^i \in \mathbb{R}^d$ is the implicit feature vector of user $i$. $\sigma(\cdot)$ is the activation function of sigmoid.

![LSTM network structure](image)

**Figure 3. LSTM network structure**

## 2.3. PMFDL Model

In Fig.1, $D_i \in \mathbb{R}^d$ is the implicit feature vector of user $i$. $R_j \in \mathbb{R}^n$ is the implicit feature vector of item $j$. $y^i_j$ is the user $i$ rating of the item $j$. $\sigma^2_U$ and $\sigma^2_V$ are the variances of the Gaussian distribution corresponding to $U_i$ and $V_j$, respectively.

According to [14], the conditional probability of the real rating matrix is shown as follows:

$$p(R|U,V,\sigma^2) = \prod_{i=1}^{N} \prod_{j=1}^{M} \left[ N(R_j | U_i^j, \sigma^2) \right]^{y_i^j}$$

(9)

Where $N(x | \mu, \sigma^2)$ is the probability density function of gaussian distribution with mean of $\mu$ and variance of $\sigma^2$. $I_y^i$ is the indicator function. If user $i$ has rated item $j$, the value is 1. Otherwise, the value is 0. Assuming that there are $N$ users and $M$ items, $U \in \mathbb{R}^{N \times d}$ is the implicit feature matrix for all users and $V \in \mathbb{R}^{M \times d}$ is the implicit feature matrix for all items.

PMF assumes that the implicit feature vector of users and items are independent of each other and the prior probability distribution obeys the gaussian distribution with zero mean value, as shown in the following formula.

$$p(U | \sigma^2_U) = \prod_{i=1}^{N} N(U_i | 0, \sigma^2_U)$$

(10)

$$p(V | \sigma^2_V) = \prod_{j=1}^{M} N(V_j | 0, \sigma^2_V)$$

(11)

The output of Attention-CNN and LSTM can be substituted into (10) and (11).

$$p(U | W_U, X_U, \sigma^2_U) = \prod_{i=1}^{N} N(U_i | y^i, \sigma^2_U)$$

(12)

$$p(V | W_V, X_V, \sigma^2_V) = \prod_{j=1}^{M} N(V_j | v_j, \sigma^2_V)$$

(13)

where $W_U$ represents all mapping matrix and biases in LSTM. $W_V$ represents all the mapping matrices and biases in Attention-CNN.

According to Bayesian formula, we can know that the posterior probability of the implicit feature matrix of users and items satisfy the following formula.

$$p(U,V,W_U,W_V | R, X, \sigma^2, \sigma^2_U, \sigma^2_V) \approx p(R | U,V,\sigma^2) \times p(U | W_U, X_U, \sigma^2_U) \times p(W_U | \sigma^2_U) \times$$

$$p(V | W_V, X_V, \sigma^2_V) \times p(W_U | \sigma^2_U)$$

(14)

Where $p(W_U | \sigma^2_U) = \prod \limits_{i=1}^{N} N(w_i | 0, \sigma^2_U)$ is the priori probability distribution of $W_U$, and $\Omega$ is the overall representation of the mapping matrix involved in LSTM network. $p(W_V | \sigma^2_V) = \prod \limits_{j=1}^{M} N(v_j | 0, \sigma^2_V)$ is the priori probability distribution of $W_V$, and $\Omega$ is the overall representation of the mapping matrix involved in Attention-CNN. $X_U$ is the input set of LSTM, and $X_V$ is the input set of Attention-CNN.

PMFDL model solves the maximum posterior probability estimate of the known evaluation matrix $R$. 

---

5
\[
\max \left( p(U,V,W_r,W_t \mid R,X,\sigma^2,\sigma^2_{\theta},\sigma^2_{\phi},\sigma^2_{\chi}) \right) = \max \left( p(R \mid U,V,\sigma^2) \times p(U \mid W_r, X, \sigma^2_t) \right) \\
\times p(W_r \mid \sigma^2_t) \times p(V \mid W_r, X, \sigma^2_t) \times p(W_t \mid \sigma^2_t)
\]

(15)

For convenience, we take the natural logarithm of equation (15).

\[
\ln \left( p(U,V,W_r,W_t \mid R,X,\sigma^2,\sigma^2_{\theta},\sigma^2_{\phi},\sigma^2_{\chi}) \right) = -\frac{1}{2\sigma^2_t} \sum_{i=1}^{N} \sum_{j=1}^{M} I_{ij} \left( R_{ij} - U_i V_j \right)^2 - \frac{1}{2\sigma^2_{\theta}} \sum_{i=1}^{N} \sum_{j=1}^{M} \| y_i - v_i \|^2 - \frac{1}{2\sigma^2_{\phi}} \sum_{i=1}^{N} \sum_{j=1}^{M} \| u_i - y_i \|^2
\]

\[
-\frac{1}{2\sigma^2_{\chi}} \sum_{i=1}^{N} \sum_{j=1}^{M} \left( w_i \right)^2 - \frac{1}{2\sigma^2_{\chi}} \sum_{i=1}^{N} \sum_{j=1}^{M} \left( v_i \right)^2 + C
\]

\[(16)\]

Where \( \Omega \) and \( \Phi \) is the number of mapping matrices of LSTM and Attention-CNN, respectively. Since the mapping matrices of Attention-CNN and LSTM are randomly initialized according to the Gaussian distribution, the mapping matrix entries in (16) can be ignored.

The maximum value of (16) can be equivalent to the minimization problem of the following formula.

\[
f = \sum_{i=1}^{N} \sum_{j=1}^{M} \left( R_{ij} - U_i V_j \right)^2 + \chi \sum_{i=1}^{N} \sum_{j=1}^{M} \left( y_i - v_i \right)^2 + \beta \sum_{i=1}^{N} \sum_{j=1}^{M} \left( u_i - y_i \right)^2
\]

\[(17)\]

Where \( \chi = \sigma^2/\sigma^2_\theta \), \( \beta = \sigma^2/\sigma^2_\phi \).

In this paper, the coordinate descent algorithm is used to solve the minimum value of \( f \). The updated expression of \( U_i \) and \( V_j \) is:

\[
U_i \leftarrow 2 \left( 2\beta I_i + \sum_{j=1}^{M} I_j v_j^T \right)^{-1} \beta \ast y_i + \sum_{j=1}^{M} R_{ij} v_j
\]

(18)

\[
V_j \leftarrow 2 \left( 2\chi I_j + \sum_{i=1}^{N} I_i U_i^T \right)^{-1} \chi \ast v_j + \sum_{i=1}^{N} R_{ij} U_i
\]

(19)

PMFDL steps are as follows:

Step1: Construct rating matrix \( R \), randomly initialize parameters \( U \) and \( W_r \).

Step2: Initialize \( V \) through Attention-CNN according to \( R \) and \( U \).

Step3: Trained \( V \) by Attention-CNN and \( U \) by LSTM according to PMF algorithm;

Step4: Execute 2, 3 steps in a loop until convergence.

3. Experimental Analysis

3.1. Data Set

This paper adopted the Amazon Instant Video (AIV) data set (http://jmcauley.ucsd.edu/data/amazon/) and MovieLens-1m (ML-1) data set (https://grouplens.org/datasets/movielens/) as the experiment- al data set. ML-1 description information from http://www.imdb.com/. The data set contains the user’s rating information for items, as well as the user’s evaluation of items and the description information of items. The user’s evaluation information are taken as the description information of the items. Before the training, the information in the data set was preprocessed. The item without description information were removed. The information of processed data set is shown in Table 1.

| Data Set | Users | Items | Ratings |
|----------|-------|-------|---------|
| ML-1     | 6040  | 3544  | 993482  |
| AIV      | 29577 | 15149 | 135188  |

To avoid over-fitting, 80% of the data were randomly selected as training set, 10% as validation set, and 10% as test set.

In the paper, we need to measure the accuracy of the forecast based on the deviation between the predicted rating and the true rating. As root mean square error (RMSE) is more sensitive to the abnormal values in the prediction, RMSE is selected as the evaluation index in this paper. The smaller
the RMSE obtained by the recommendation algorithm means the more accurate the prediction. The calculation method of RMSE is as follows.

\[
RMSE = \sqrt{\frac{1}{T} \sum_{(i,j) \in T} (R_{ij} - \hat{R}_{ij})^2}
\]  

(20)

Where \( R_{ij} \) represents the user \( i \) actual rating for the item \( j \). \( \hat{R}_{ij} \) represents the user \( i \) predictive rating for the item \( j \), and \( T \) represents the number of test set.

3.2. Results Analysis

After the data set was processed, the average length of item’s description information in AIV was 200, and that of ML-1 was 230. We assume that the longest description information of the item is 300 in this paper. In the pre-training of word vector, the dimension of word vector is set to 100. The mapping matrix and bias in Attention-CNN and LSTM networks are randomly initialized by gaussian distribution.

![Figure 4. The influence of implicit feature dimension on RMSE](image)

![Figure 5. The influence of convolution kernel number on RMSE](image)

To verify the performance of the model in this paper, we discuss the impact of implicit feature dimensions and the number of convolution kernel in the ML-1 dataset. It can be seen from Fig.4 and Fig.5. The implicit feature dimension is too large, the model will remember the feature of each item, leading to overfitting and reducing the prediction accuracy. Too few convolution kernel will lead to insufficient feature extraction, and too many will over-refine each feature of an item and lead to remember accurately. In order to verify the accuracy of the model prediction, the dimension of implicit features is set to 60 and the number of convolution kernel is set to 60 in the subsequent experimental verification.
The number of neurons inside the cell in the LSTM structure will affect the learning of user’s implicit feature. We set different numbers to train the model in the ML-1 data set, and the results are shown in Fig.6. When the number of neurons in the Cell is small, it cannot effectively extract user’s implicit feature. When there are too many neurons in the cell, it will accurately learn the feature of each user and lead to overfitting. According to the results in Fig.6, we set the number of neurons inside the cell in LSTM structure as 60. The parameter settings are shown in Table 2.

![Figure 6. The influence of unit quantity on Model](image)

Table 2. Parameter Settings.

| Model   | $\chi$ | $\beta$ | $\alpha$ |
|---------|--------|---------|----------|
| PMF     | 0.1    | 0.1     | 1        |
| ConvMF  | 100    | 10      | -        |
| PMFDL   | 100    | 8.5     | -        |

PMF: Probabilistic Matrix Factorization uses ratings for collaborative filtering.
ConvMF: Convolutional Matrix Factorization uses CNN to extract item information.
PMFDL: The algorithm is proposed in this paper.

Model training and prediction are conducted according to the parameters set in Table 2. The RMSE results are shown in Table 3. According to the experimental results in Table 3, the PMFDL algorithm is superior to the CDL and ConvMF models in terms of recommendation accuracy.

Table 3. RMSE Result.

| Model   | ML-1 | AIV  |
|---------|------|------|
| PMF     | 0.9534 | 1.3827 |
| ConvMF  | 0.8522 | 1.1235 |
| PMFDL   | 0.8430 | 1.1202 |

ConvMF can achieve higher recommendations than PMF because ConvMF takes into account the contextual information of item’s description information. PMFDL filters out the interference information through the attention mechanism. At the same time, it considers the connection between items and items in the user’s historical record and the timeliness of user’s interest. Experimental results show that our model can effectively obtain the implicit feature of users and items.

4. Conclusion
In order to obtain the implicit feature of items and users and improve the recommendation efficiency, the PMFDL algorithm model is proposed in this paper. The model considers the contextual information of item’s description information and filter out the interference information. The
Experimental results show that PMFDL algorithm model can effectively obtain the implicit feature of items and users and improve the accuracy of recommendation. Although this paper considers the context information of item’s description information, it still needs to improve the semantic information extraction of content. In future work, we will consider using RCNN-HW network to extract the feature of description information.

References
[1] Zhang S, Yao L, Sun A. Deep Learning based Recommender System: A Survey and New Perspectives. 2017.
[2] Koren, Yehuda. "Factorization meets the neighborhood: a multifaceted collaborative filtering model." ACM SIGKDD International Conference on Knowledge Discovery and Data Mining ACM, 2008, pp. 426-434.
[3] Zheng L, Noroozi V, Yu P S. Joint deep modeling of users and items using reviews for recommendation. Proceedings of the Tenth ACM International Conference on Web Search and Data Mining ACM, 2017, pp. 425-434.
[4] Wang H, Wang N, Yeung D Y. Collaborative deep learning for recommender systems. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining ACM, 2015, pp. 1235-1244.
[5] Kim D, Park C, Oh J, Lee S, Yu H. Convolutional matrix factorization for document context-aware recommendation. Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016, pp. 233-240.
[6] Kim D, Park C, Oh J, Yu H. Deep hybrid recommender systems via exploiting document context and statistics of items. Information Sciences, 2017, 417, pp. 72-87.
[7] Shen X, Yi B, Zhang Z, Shu J, Liu H. Automatic Recommendation Technology for Learning Resources with Convolutional Neural Network. International Symposium on Educational Technology. IEEE, 2016, pp. 30-34.
[8] Okura S, Tagami Y, Ono S, Tajima A. Embedding-based News Recommendation for Millions of Users ACM SIGKDD International Conference on Knowledge Discovery and Data Mining ACM, 2017, pp. 1933-1942.
[9] Devooght R, Bersini H. Collaborative Filtering with Recurrent Neural Networks. 2017.
[10] Seo S, Huang J, Yang H, Liu Y. Representation Learning of Users and Items for Review Rating Prediction Using Attention-based Convolutional Neural Network. International Workshop on Machine Learning Methods for Recommender Systems (MLRec)(SDM’17). 2017.
[11] Chen J, Zhang H, He X, et al. Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval. ACM, 2017, pp. 335-344.
[12] Twardowski B. Modelling contextual information in session-aware recommender systems with neural networks. Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016: 273-276.
[13] Soh H, Sanner S, White M, Jamieson G. Deep sequential recommendation for personalized adaptive user interfaces. Proceedings of the 22nd International Conference on Intelligent User Interfaces. ACM, 2017, pp. 589-593.
[14] Mnih A, Salakhutdinov R R. Probabilistic matrix factorization. Advances in neural information processing systems. 2008, pp. 1257-1264.
[15] Wen Y, Zhang W, Luo R, Wang J. Learning text representation using recurrent convolutional neural network with highway layers. arXiv preprint arXiv:1606.06905, 2016.
[16] He R, McAuley J. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. proceedings of the 25th international conference on world wide web. International World Wide Web Conferences Steering Committee, 2016, pp. 507-517.