Recommendation Algorithm for Federated User Reviews and Item Reviews

Xingjie Feng
Civil Aviation University of China
Dongli District,
Tianjin City，China
+86 022 24092621
xjfeng@cauc.edu.cn

Yunze Zeng
Civil Aviation University of China
Dongli District,
Tianjin City，China
+86 022 24092621
540117253@qq.com

Yixiong Xu
Civil Aviation University of China
Dongli District,
Tianjin City，China
+86 022 24092621
1007108634@qq.com

ABSTRACT

The recommendation model based on scoring matrix is widely used. Although it has achieved certain recommendation accuracy, it ignores the large amount of semantic information available in the reviews that reflects the user's interests, and the data sparsity problem still exists. In response to the above problems, a two-channel CNN recommendation algorithm (C-DCNN, Combine-Double CNN) that combines user reviews and item reviews is proposed. First, the user and item review texts are vectorized into word vectors, and then the features of users and the items are extracted by using two CNN networks respectively. Finally, the abstract features are mapped to the same feature space through the dot product in the shared layer which aims at predicting the user's rating for a particular item. Experiments on the public datasets of Amazon, Yelp, and Beer show that the C-DCNN model makes full use of reviews to characterize the deep features of users and items. The MSE of the model on different datasets is smaller than other benchmark algorithms. And C-DCNN effectively alleviates the problem of data sparsity.

CCS Concepts

• Information systems→Recommender systems • Computing methodologies→Neural networks.

Keywords

Word Vector; Recommendation System; Data Sparsity; Dual Channel; Convolutional Neural Network

1. INTRODUCTION

With the rise of the Internet, the amount of data has exploded in the past decade [1]. In order to mine valuable information to customers more efficient from data, many forms of recommendation systems have been born. In previous researches, various recommendation algorithms are discussed and analyzed, such as user similarity recommendation algorithm [2], user implicit factor feedback [3], review text emotion [4-6] and context-aware recommendation system.

AIVR 2018, November 23–25, 2018, Nagoya, Japan.
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-6641-0/18/11…$15.00
DOI: https://doi.org/10.1145/3293663.3293667

Those recommendation methods are all based on explicit scoring whether they are based on item-based recommendations or based on collaborative filtering between items and users. Although they achieve certain recommendation accuracy, they ignore the large amount of semantic information available in the reviews. In general, there are two main types of traditional recommendation methods: content-based recommendations [7-8] and collaborative filtering recommendations [9-10]. The content-based recommendation is to use the item that the user has scored, and the item can be expressed as a series of features, and when the recommended item is close to the item feature scored by the user, it would be recommended to the user. Collaborative filtering recommendations depend on items that have been scored by other users in the system. If user U and user V are very similar in other item ratings, the ratings of new items by users U and V may also be similar. Without having to know a large amount of information of the user or the item, the collaborative filtering method utilizes the user's rating of the item or user behavior to provide personalized recommendations for the user, so it can better capture information that is difficult to automatically identify by the system. For example, the most popular music and movie recommendation systems such as Google News, GroupLens, Today's headlines, etc. are all based on collaborative filtering. To the best of our knowledge, although traditional recommendation method has a satisfying effect in improving the accuracy of the recommendation system, several problems still exists: the sparseness of the scoring matrix, the cold start problem and the recommended system scalability. To solve these problems, several text-based methods have been proposed recently, which would be mainly introduced in the next section.

2. RELATED WORKS

With the rise of deep learning, text-based content mining methods based on neural networks have gradually formed a craze in the tasks of text-based classification, clustering, sentiment analysis and other related natural language processing [11]. Deep learning combines low-level features to form a denser high-level semantic representation, which automatically discovers the implicit feature representation of data, and solves the problem of manual features which required in traditional machine learning, and breakthroughs have been made in the field such as image recognition, machine translation, and speech recognition.

Literature [12] introduced the idea that word frequency statistics was first applied to automatic classification, which opened up the research in the field of text mining. The research in [13] shows that the application of review texts in the recommendation system is mainly divided into two categories: user modeling and item modeling. The most common modeling method in the review text comes from the field of information retrieval, in which the user is modeled by directly using the words in his associated review text , correspondingly, the item can also be modeled by directly using its associated review text. Literature [14] utilizes reviews to learn the distribution of item features on different topics and the user's preference for different features of the item, integrating item features and user preferences into the traditional collaborative filtering algorithm to improve the accuracy of recommendation. In literature [15], the LDA model was used to find the subtopics of the Yelp review text before the prediction score. Literature [16] found in the experiment that the sentiment analysis of the review text also helps to improve the prediction accuracy, in which shows that the use of text modeling is the most direct way to solve the sparseness of scoring matrix. However, most of the previous methods are to mine user reviews and item reviews separately, and derive their potential topic distributions without considering the interaction between them. Therefore, this paper proposes a C-DCNN that considers the characteristics of users and items as well as the interaction between them. The C-DCNN will be described in detail below, and the validity of the model is verified by comparative experiments.

3. METHODOLOGY

3.1 Notations

Net_u ： A network that extracts user behavior preference characteristics.

Net_i ： A network for extracting item characteristics.

M_u ： The word vector embedding matrix of the user's reviews.

M_i ： The word vector embedding matrix for the reviews of the item.

t ： The window size of the convolution kernel.

K_j ： The j-th convolution kernel in the convolutional layer.

o_j ： The output of the j-th neuron in the convolutional layer.

F_u ： The output after the full connection layer Net_u .

F_i ： The output after the full connection layer Net_i .

g ： The concatenation vector between F_u and F_i .

$\hat{y}_{ui}$ ： User's predicted score for the item .

3.2 Model Structure

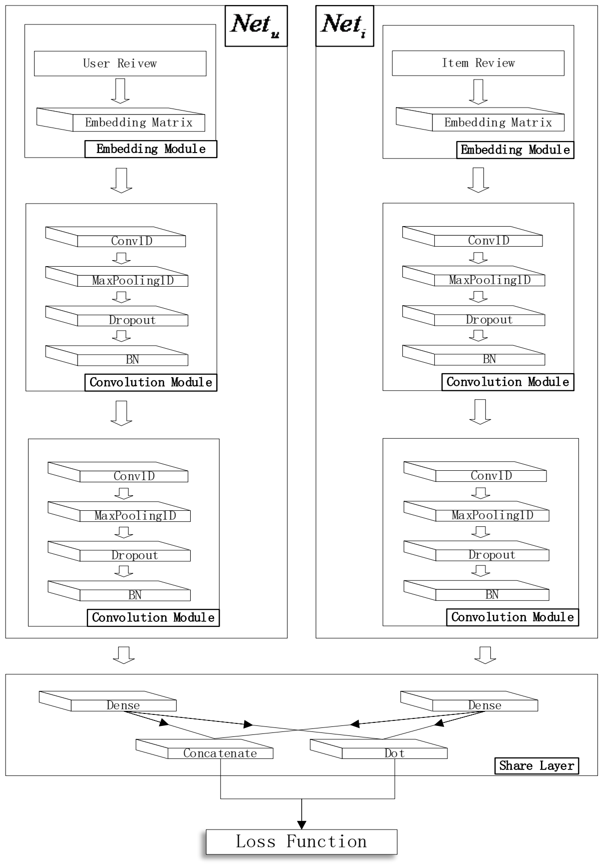
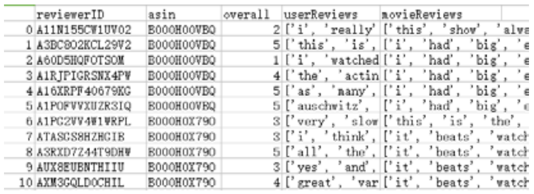


Figure 1. C-DCNN structure

The overall structure of the C-DCNN model proposed in this paper is shown in Figure 1. The model consists of two parallel networks. The left network is mainly used to extract the user's behavior preference characteristics while he right network is mainly used to extract the features of the items. Finally, the output of the two networks is integrated through a shared layer to predict the user's score. Overall, user reviews and item reviews are used as input and the final output is the user's rating of the item. The first layer of the model is the word vector embedding layer, in which user reviews and item reviews are mapped into a word vector matrix to capture the semantic information in the reviews. The second layer and the third layer are convolutional neural network modules, which are mainly used to extract user behavior preferences and feature expressions in the high-level abstract space with the specific structures: convolution operation, max pooling, dropout, batch normalization. The fourth layer is the shared layer in which the traditional neural network model directly concatenating the output of different channels into a matrix as the input of the subsequent layer. Besides, this paper has innovated the traditional shared layer structure with adding the dot item of the output from different channels into a matrix which is produced by concatenating the output of different channels. The dot product operation can effectively capture the correlation between any potential factors between the user and the item, so that the model can effectively predict the user's score on items that have never been in contact with his history. Since the only difference between left and right network is the word vector embedding layer of the first layer, the following sections mainly describes the processing details of the left network.

3.3 Words Embedding Layer



	reviewerID	asin	overall	userReviews	movieReviews
0	A11N155CW1UV02	B000H00VBQ	2	['i', 'really'	['this', 'show', 'alwa
1	A3BC8O2KCL29V2	B000H00VBQ	5	['this', 'is',	['i', 'had', 'big', 'e
2	A60D5HQFOTSOM	B000H00VBQ	1	['i', 'watched	['i', 'had', 'big', 'e
3	A1RJPIGRSNX4PW	B000H00VBQ	4	['the', 'actin	['i', 'had', 'big', 'e
4	A16XRPF40679KG	B000H00VBQ	5	['as', 'many',	['i', 'had', 'big', 'e
5	A1POFVVXUZR3IQ	B000H00VBQ	5	['auschwitz',	['i', 'had', 'big', 'e
6	A1PG2VV4W1WRPL	B000H0X790	3	['very', 'slow	['this', 'is', 'the',
7	ATASGS8HZHGIB	B000H0X790	3	['i', 'think',	['it', 'beats', 'watch
8	A3RXD7Z44T9DHW	B000H0X790	5	['all', 'the',	['it', 'beats', 'watch
9	AUX8EUBNTHIIU	B000H0X790	3	['yes', 'and',	['it', 'beats', 'watch
10	AXM3GQLDOCHIL	B000H0X790	4	['great', 'var	['it', 'beats', 'watch

Figure 2. Amazon Reviews Dataset

Figure 2 shows the first 10 rows of the Amazon review data set, where the reviewerID of each row represents the user ID, asin represents the item ID, overall represents the user's rating of the item. Each row is one data point that contains all of the users reviews (excluding the review for the current item), all of the item's reviews (excluding that written by the current user), and the associated rating. In the experiment, the userReviews and movieReviews are mapped to a dense vector of a certain dimension by word2vec technique, and then input them into the two convolution channels of C-DCNN for recommendation.

The word vector map $f: M \rightarrow \Re^n$ represents a function that maps words to n-dimensional vectors according to the word dictionary M, and n takes a value of 50 in the C-DCNN module. The use of word2vec technique can greatly improve the performance of NLP applications. In the word vector embedding layer, reviews are mapped to word vector matrices to mine semantic information. First, all the review word vectors of user u are merged into one document d, and the length of the document is fixed to contain n words, so the user's review word vector matrix M can be expressed by (1):

$$M_u = \theta(d_1) \oplus \theta(d_2) \oplus \cdots \oplus \theta(d_n) \quad (1)$$

Where d_k represents the k-th word in document d, and $\theta(d_k)$ represents the embedding of the word d_k to the corresponding n-dimensional word vector space.

3.4 Convolution Module

The second layer is a conventional convolution module, which includes a convolution layer, a max pooling layer, a dropout layer, and a BN layer. The convolution layer is used to extract the abstract feature of the review word vector matrix M_u of the user u. Assuming that there are a total of m neurons, and one of the neurons j uses a convolution kernel $K_j \in \Re^{c \times t}$ and has a word window size of t. Therefore, for the word vector matrix M_u, the convolution result of each convolution kernel k_j can be expressed by (2):

$$\kappa_j = f(M_u * K_j + b_j) \quad (2)$$

The symbol * indicates the convolution operation, b_j is the offset term, and f is the activation function ReLUs (Rectified Linear Units), and its expression is (3):

$$f(x) = \max\{0, x\} \quad (3)$$

In Figure 3, the max pooling layer in the C-DCNN model uses one-dimensional max pooling and the word window size is set to 2. For example, 9 and 3 in the range of the first word window take the maximum value 9; 5 and 2 in the range of the second word window take the maximum value 5; 4 and 7 in the range of the third word window take the maximum value of 7.

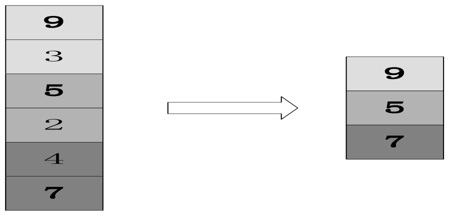


Figure 3. MaxPooling with Word Window 2

Through the max pooling layer, the maximum value in the feature map is found, ie the most important features are captured. At the same time, it also plays a role in reducing the dimension of the feature map, reducing the network parameters, and speeding up the training, which can relieve the problem of over-fitting to some extent. After the max pooling layer, the output k_i of the convolution operation will be reduced to a fixed-size vector:

$$o_j = \max\{\kappa_1, \kappa_2, ..., \kappa_{(n-t+1)}\} \quad (4)$$

Equation (4) is the processing result of the max pooling layer. This model uses a total of k convolution kernels to extract a variety of different features jointly, so the output vector of k convolution kernels can be expressed by equation (5):

$$O = \{o_1, o_2, ... o_k\} \quad (5)$$

Dropout is an effective mechanism to prevent over-fitting of neural networks. Each round of training inactivates some hidden layers of neurons randomly so that each training network is different, which is equivalent to randomly adding some noise in each training to enhance the generalization ability and robustness of the network. In this experiment, the probability of taking dropout is 0.5, which means that each connection has a 50% probability of being removed during training, and all connections are retained during testing.

The main role of the BN layer is to allow the training network to use a large learning rate to accelerate the network convergence. At each gradient descent, the corresponding activation function is normalized by mini-batch so that the result (each dimension of the output signal) has a mean of 0 and a variance of one.

After passing through the second convolution layer, the data is input to the fully connected layer, and finally the high-level abstract feature vector $F_u \in \Re^{d \times 1}$ of the user u is obtained. the fully connected layer can be specific expressed by equation (6):

$$F_u = f(W \times O + b') \quad (6)$$

Where matrix W is the weight parameter of the fully connected layer and b' is the offset term of the fully connected layer. Finally, we can respectively get the output F_u, F_i of the user convolution network Net_u and the item convolution network Net_i.

3.5 The Shared Layer

F_u, F_i is respectively a user feature and an item feature in different feature spaces. In order to accurately predict the user's score on the item, they need to be mapped into the same feature space. First, the user feature vector F_u and the item feature vector F_i are concatenated to obtain a vector $g = (F_u, F_i)$. Although F_u, F_i is in different feature spaces, g is an intersection between the two spaces, that is, their potential interaction. In order to further capture the interaction, we add F_u, F_i's dot product item $F_u^T F_i$ to the score prediction function, then user u's score prediction function $\hat{y}_{ui}$ of the item i can be expressed by (7):

$$\hat{y}_{ui} = \hat{w}_0 + \sum_{i=1}^{|g|} \hat{w}_i g_i + F_u^T F_i \quad (7)$$

Where w is the global offset term and $\hat{w}_i$ is the weight of the i-th variable g_i in vector g.

3.6 Network Training

In this paper, the loss function C of the model is proposed to minimize the error between the score prediction function $\hat{y}_{ui}$ and the true score y_{ui} in the data set. The expression (8) of the loss function C is given below:

$$\min_{w_i^*, w_0^*} \sum_{(u,i)\in\kappa} (\hat{y}_{ui} - y_{ui})^2 + \lambda \|W\|^2 \quad (8)$$

The constant λ controls the degree of regularization. The loss function uses the $L2$ regular term. The advantage of $L2$ normalization is that the network is more likely to learn a smaller weight, and this method can also be seen as a eclectic way to minimize the original cost function and find smaller weights.

4. EXPERIMENTS

4.1 Datasets and Evaluation Metric

In the experiment, the following three common datasets were used to evaluate the performance of the C-DCNN model:

Yelp: This is a large dataset about restaurant reviews, with a total of about 1 million reviews and ratings.

Amazon: The Amazon Review Dataset contains information about Amazon item reviews and metadata. There are about 140 million reviews from May 1996 to July 2014, covering a total of 21 items of reviews.

Beer: This is a dataset about beer reviews from ratebeer.com. There are about 3 million reviews from 10 years in total.

MSE and MAE were used to evaluate the C-DCNN model during the experiment. The reason for the use is because the C-DCNN model can be classified as a regression algorithm, and the recognized indicators for the regression algorithm are MSE and MAE, and the expressions are as shown in formula(9)-(10):

$$MSE = \frac{1}{N} \sum_{n=1}^{N} (\hat{y}_n - y_n)^2 \quad (9)$$

$$MAE = \frac{1}{N} \sum_{n=1}^{N} |\hat{y}_n - y_n| \quad (10)$$

Suppose there are N samples in the data set, where y_n is the true score of the n-th sample and $\hat{y}_n$ is the predicted score of the n-th sample.

4.2 Experimental Procedure

Input: User Reviews and Item Reviews.

Output: User's predicted rating on the item.

Step1: Data cleaning of datasets. Due to the different lengths of each user's reviews and item reviews, the first 20 words with the highest frequency are retained in the experiment.

Step2: Use Word2Vec to map user reviews and item reviews after data cleaning into 50-dimensional word vectors.

Step3: Input the trained user review word vector and the item review word vector into two convolution modules for feature extraction.

Step4: Concatenating the output of the two convolutional modules,and adding the dot product item of the two output word vectors, the equation (8) is trained by the gradient descent method to minimize the loss function.

4.3 Hyper-Parameter

Table 1. Comparison of Relevant Indicators Under Different Word Vector Dimensions

	50	100	150	200	300
MSE	0.924	0.951	0.946	0.937	0.977
iterations	30	45	60	70	90

It can be seen from Table 1 that the best result can be obtained when the word vector dimension is 50. The reason is that the reviews are mostly positive, neutral and negative vocabulary. After the 20 most frequent words in each review set are mapped by word2vec, the distance in the high-dimensional space is mainly concentrated in these three parts. The word vectors between each part are very close. When the word vector dimension is increased, the word vector in the high-dimensional space becomes sparser and weakens the association between words and words.

Table 2. Comparison of Relevant Indicators Under Different Convolution Layers

	1-layer	2-layer	3-layer
avg-MSE	1.132	0.924	1.241
iteration	50	30	20
parameters	837,473	208,897	51,873

The size of 1 to 3 layer's convolution kernel windows is 3, 4, and 5 respectively. On the Amazon dataset, the 1, 2, and 3 layers were used, and the number of convolution kernels was 16, and 3 sets of comparison experiments were performed. Each group of experiments was performed 10 times, and 80% of the data was randomly selected as the training set while 20% as a test set. The avg-MSE corresponds to the average MSE of the 10 experiments, and the iteration is the number of iterations required for the current model structural loss value to stabilize.The table have shown that the best results can be achieved with a 2-layer convolution module. The analysis found that the number of convolution parameters was the highest compared to the use of 2-layer and 3-layer convolution modules. This is due to the 1-layer convolution module which the number of dimensionality reduction is small and the output matrix is large, resulting in more fully connected neural cells that the extraction level of text features is not sufficient. On the contrary, under the 3-layer, the convolution module extracts features excessively abstract and loses a lot of information.

Table 3. Comparison of Relevant Indicators Under Different Convolution Kernel Size

	4	8	16	32	64
avg-MSE	1.09	1.12	0.924	1.15	1.32
iteration	9	16	30	50	60
parameters	52,609	104,449	208,897	420,865	857,089

It can be seen from Table 2 that the use of the 2-layer convolution module is the best, so the comparison experiment of the number of different convolution kernels here uses a 2-layer convolution structure. As is shown in Table 3, five sets of contrast experiments were performed using 4, 8, 16, 32, and 64 convolution kernels, and each set of experiments was performed 10 times. Each time, 80% of the data was randomly selected as the training set and 20% was used as the test set. The avg-MSE corresponds to the average MSE of the 10 experiments, and the iteration is the number of iterations required for the current model structure loss value to stabilize. It have shown that the number of optimal convolution kernels is 16 for the reason that under 4 and 8 convolution kernels, the abstract feature has limited capabilities and the convolution kernel is not fully utilized to extract the deep features of users and items. In contrast, when 32 and 64 convolution kernels are used, the extraction features are too detailed, resulting not only in overfitting but also in longer training time.

4.4 Baselines

In the comparative experiment, this paper is compared with the following four baselines:

• **Singular Value Decomposition (SVD):** SVD is one of the most commonly used matrix decomposition in collaborative filtering recommendation algorithms. It only uses the user's scoring matrix for the item as input, utilizing the interaction of hidden spaces in the low rank matrix to predict the score. The experiment retains this method for comparison with the traditional recommendation algorithm that relies solely on the scoring matrix.

• **Factorization Machine (FM):** FM is a generalization of linear regression. Compared to linear regression which only the weight of a single independent variable can be calculated, FM can calculate the weight between different independent variables. It is predicted by the user's scoring matrix score and does not utilize the textual information of the review.

• **Topic Model (LDA):** LDA is a well-known topic generation model that can be used to learn the implicit topic distribution from a review dataset. Taking the reviews corresponding to the user and the item as input, the learned topic distribution is used as the item hidden feature and the user hidden feature to predict scores which are optimized by the gradient descent algorithm. The experiment retains this method in line with the literature [15] as a baseline method for comparison.

• **Collaborative Topic Regression(CTR):** CTR is a collaborative filtering algorithm that combines topical probability models, which uses textual information to recommend articles and has superior performance. The experiment retained this method to verify whether the C-DCNN model made full use of the review information.

4.5 Results

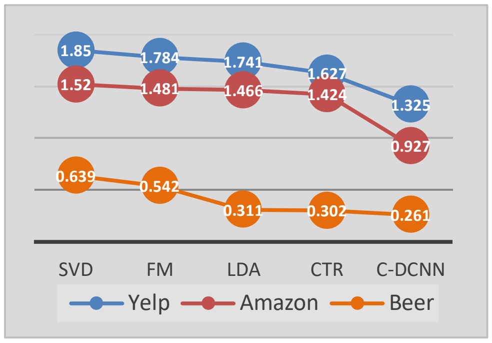


Figure 3. MSE of C-DCNN and Baselines on Three Datasets

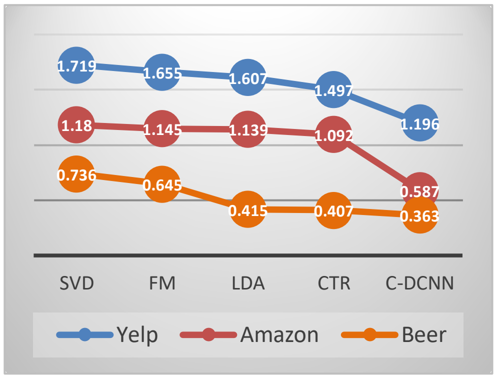


Figure 4. MAE of C-DCNN and Baselines on Three Datasets

It can be seen from Fig. 3 and Fig. 4 that the experimental results of C-DCNN on different datasets are better than the baselines. The reason is that C-DCNN model uses two convolution channels to extract the connection between user reviews and item reviews, so the level of abstraction of C-DCNN is more detailed than that of only one convolution channel. On the other hand, rich semantic information as input can reflect the user's emotional tendency more realistically than a single scoring matrix.

Through comparison experiments, it is also found that the performance based on the deep learning method is obviously superior to the traditional machine learning method. This is because the traditional machine learning method is greatly influenced by human factors. The machine learning models based on feature classification are mostly shallow models, and the structure is relatively simple leading to the training data cannot be fully fitted. The deep learning model can automatically extract features from the original data, reducing the incompleteness of manual features extraction in traditional methods. In the future, as the amount of data increases, the advantages of the deep learning model will gradually emerge, and the performance of the recommendation method based on deep learning will be more satisfying.

4.6 Model Structure Analysis

In order to verify the rationality of the C-DCNN dual-channel structure, effectively alleviate the data sparsity and the necessity of using the BN layer to improve the training speed, the C-DCNN with dual-channel structure and the SC-DCNN with single-network structure are respectively compared below.

The differences between C-DCNN and SC-DCNN are described below:

- **C-DCNN:** This model has two networks Net_u and Net_i, Net_u which extracts the hidden features of the user mainly from the user's reviews, and Net_i mainly extracts the hidden features of the item from the items' reviews. Finally, a shared layer is used to calculate the output of the two networks into the same feature space for score prediction.

- **SC-DCNN:** The SC-DCNN uses only one network Net, and the specific convolution layer structure of the network is the same as Net_u. The user review matrix M_u and the item review matrix M_i are concatenated into a matrix M as an input to Net. Since there is only one network, the shared layer is cancelled, namely, the loss function without dot product.

As is shown in Table 4 below, C-DCNN and SC-DCNN were respectively applied to the datasets Yelp, Amazon, and Beer with the aim at comparing the score prediction MSE between two models. It can be seen that the average MSE of the C-DCNN with dual channel structure on the three data sets is 30% lower than the SC-DCNN of the single network structure.

Table 4. Comparison of MSE between SC-DCNN and C-DCNN

DataSet	SC-DCNN	C-DCNN
Yelp	1.726	1.325
Amazon	1.573	0.982
Beer	0.412	0.261
avg-MSE	1.237	0.856

One of the most straightforward explanations is that the parameters of the dual channel are twice as large as those of the single network. The C-DCNN can capture the hidden features of users and items more meticulously, and the dual-structure Net_u and Net_i are parallel structures that can't interfere with each other while focusing on extracting the hidden features of the user and the item's high-level abstraction. Finally, this model maps the hidden feature information of the two networks to the same feature space through an additional sharing layer, so it can fully exploit the potential affinity of the user's hidden features and hidden features of the item and the accurate prediction scores is achieved effectively.

The cold start problem is currently existing in the recommendation system, particulaly, when new users join the system, they available scores are limited , and it is not easy for the system to learn the implicit features from their ratings. It is found in the experiment that when the user group and the scoring amount are sparse, the MSE of the C-DCNN model proposed in this paper is obviously superior to the traditional single-channel structural model and effectively alleviates the data sparsity.

4.7 Stability Analysis

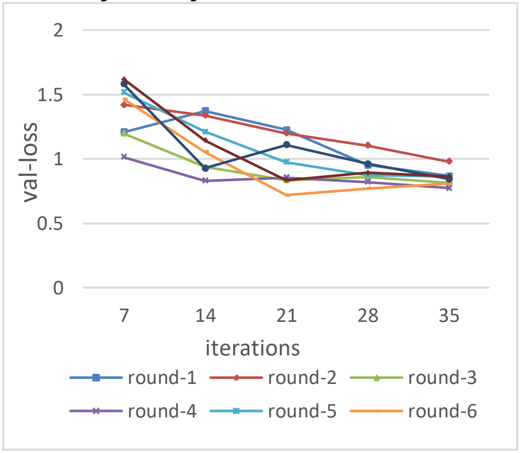


Figure 5. C-DCNN training process

In order to verify the stability of the C-DCNN model, 80% of the data set was randomly selected as the training set, 20% was used as the verification set, and 8 rounds of experiments were performed. As a result, as shown in Fig. 5, the loss of the validation set in the 8-round experiment is negatively correlated with the number of training rounds. And there is no large fluctuation in the overall training process, which can be reached with a large learning rate to the lowest loss value. The experimental results show that the C-DCNN model has good stability.

4.8 Conclusion and Future Work

This paper proposes a two-channel CNN recommendation algorithm (C-DCNN) that combines user and item reviews. The C-DCNN model maps the user and item review texts in the form of word vector respectively, and then represents them to a highly abstract feature space by using CNN, and finally uses the dot product item to find the interaction between the user and the item to complete the recommendation. Compared with the model using the scoring matrix, the C-DCNN using the text word vector effectively mitigates the impact of data sparsity and greatly improves the accuracy of prediction. In addition, the experiment proved the rationality of using user reviews and item reviews with two networks respectively, and showed that C-DCNN training process has good stability.

This article just focuses on the use of review texts for recommendation, however, factors such as user and item characteristics would change over time, so the weight of potential features and the influence of context in the recommendation system will be considered in future research.

5. ACKNOWLEDGMENTS

This work is supported in part by the National Natural Science Foundation of China and China Civil Aviation Administration Joint Fund Project under Grant No.U1233113,National Natural Science Youth Foundation of China under Grant No.61301245 and 61201414.

6. REFERENCES

[1] Huang Liwei, Liu Yanbo, Li Deyi. Recommendation System Based on Deep Learning[J]. Chinese Journal of Computers, 2017, 40: 1-29.

[2] Xu Zhiming, Li Dong, Liu Ting, Li Sheng, Wang Gang, Yuan Shulun.The similarity measure of Weibo users and its application[J].Chinese Journal of Computers,2014.3 27(3):207-218.

[3] Yu Gang, Wang Zhiyan, Shao Wei, Hu Shuyue, Cai Yi. Personalized review recommendation based on singular value decomposition[J].Journal of University of Electronic Science and Technology of China,2015,44(04):605-610.

[4] Shen Chang, Yan Junzhong. Text sentiment classification algorithm based on two-channel convolutional neural network[J]. Pattern Recognition and Artificial Intelligence, 2018, 31(02): 158-166..

[5] Li Hanzhen, Qian Li, Zhou Pengfei. Affective Analysis and Mining for Commodity review Texts[J].Information Science,2017,35(01):51-55+61.

[6] Li Yong, Zhou Xueguang, Sun Yan, Zhang Huanguo. Research and Implementation of Sentiment Analysis in Chinese Weibo[J].Journal of Software,2017,28(12):3183-3205.

[7] Leng Yajun, Lu Qing, Liang Changyong. Overview of Collaborative Filtering Recommendation Techniques[J]. Pattern Recognition and Artificial Intelligence, 2014, 27.8: 720-734..

[8] Shan Jingjing. Research on content-based personalized recommendation system [D]. Northeast Normal University, 2015.

[9] Wang Ruiqin, Jiang Yunliang, Li Yixiao, Lou Jungang. A Collaborative Filtering Recommendation Algorithm Based on Multiple Social Trusts[J]. Journal of Computer Research and Development, 2016, 53(06): 1389-1399..

[10] LU Kun, XIE Ling, LI Mingchu. A Collaborative Filtering Recommendation Algorithm Based on Implicit Trust[J].Microcomputer Systems,2016,37(02):241-245.

[11] Silver D, Huang A, Maddison C J, et al. Mastering the game of Go with deep neural networks and tree search[J]. Nature, 2016, 529（7587）: 484-489.

[12] Yang Xia, Huang Chenying. Summary of text mining [J]. Science and Technology Information. 2009 (33).

[13] Chen L，chen G,Wang F.Recommender systems based on user reviews:the state of the art[J].User Modeling and User-Adapted Interaction,2015,25(2):99-154.

[14] Tan Yunzhi, Zhang Min, Liu Yiqun, Ma Shaoping. Collaborative Recommendation Framework Based on User Rating and review Information[J]. Pattern Recognition and Artificial Intelligence, 2016, 29(04): 359-366.

[15] Huang J, Rogers S and Joo E.Improving restaurants by extracting subtopics from yelp reviews// Proceedings of the iConference, Berlin,Germany,2014:1-5.

[16] Mukherjee S, Basu G, and Joshi S. Incorporating author preference in sentiment rating prediction of reviews[C].Proceedings of the 22nd International World Wide Web Conference(WWW), Rio de Janeiro,Brazil, 2013: 47–48.