Research on Computer Information Retrieval Based on Deep Learning

Chenwei Wan
College of Software and Internet of Things Engineering, Jiangxi University of Finance and Economics, Nanchang, Jiangxi, 330013, China

Abstract. With the growth of digital information on the Internet, online stores, online music, video and photo libraries, search engines and recommendation systems have become the main way people are looking for information quickly. Recently, the development of deep learning has achieved many results in the fields of speech recognition, image processing and natural language processing. In addition, deep learning is also applied in the field of recommendation systems and information retrieval. This paper mainly studies a real-time recommendation algorithm for smart TV based on deep learning. The recommendation of TV program based on dnn and rnn is applied to the content-based method in the recommendation system. Experiments show that the model can effectively solve the problem of data sparsity and can alleviate the cold start in collaborative filtering to some extent.

1. Introduction

With the rapid development of the Internet, the TV industry has also shifted from traditional one-way channel programs to Internet-based smart interactive TVs. The explosive growth of the Internet makes it more and more difficult for users to find content of interest from a large number of programs and videos. Therefore, it is necessary to find an efficient and convenient recommendation method, which not only saves the time for users to find video of the program. It can also improve the user experience of using smart TV, so that users are mentally dependent on it. The Internet makes a large amount of new information appear every day. For very large data sets, the traditional recommendation method has not been able to obtain accurate recommendation results, and the recommendation efficiency is extremely low.

In recent years, with the rapid development of hardware technology, more and more methods based on deep learning and analysis of big data have been applied to practice. Because of the huge training data set, the results obtained by the analysis are often more accurate and comprehensive. Therefore, how to apply deep learning technology and smart TV recommendation field will be a research hotspot in the future. At the same time, as a multi-user shared terminal, the same user id may correspond to different family members' preferences, and the user characteristics should be extracted at different times.

2. Overview of the algorithm

In view of the huge amount of data and the recommendation of multi-user terminals, this paper mainly studies a real-time recommendation algorithm for smart TV based on deep learning. Apply a deep learning framework based on dnn and rnn on the content-based approach in the recommendation system. The algorithm steps are as follows:
(1) User information acquisition module: for user behavior: extracting browsing records, on-demand records, and purchase records of the user for a period of time; for user basic information: extracting the user purchase model, the province, the city, and the user activity;

(2) The user feature processing module: first pre-processes the user information data, and embedding the processed data to obtain an input vector input into the deep learning framework;

(3) Model training phase: the processed user input vector is input into the dnn-rnn framework, and the hidden layer is set to three layers, and the user feature vector at a certain moment is obtained through three transformations;

(4) The video feature extraction module is configured to synchronize with the user feature processing module to extract a video id, a type, a release time, a director, and the like into a video feature vector;

(5) The recommended program acquisition module: performs a dot product operation on the user feature vector at a certain moment and the video feature vector, and sorts the user's Top-N recommendation.

3. Algorithm flow details
The following figure shows the flow chart of the real-time recommendation system algorithm based on deep learning.

**Figure 1.** Flow chart of real-time recommendation system algorithm based on deep learning
Step 01: In the data preprocessing stage, information such as user on-demand records, purchase records, and browsing records are extracted from the database, and data cleaning, data protocol, and reconstruction are performed by using Weka.

Step 02: Grab direct data information, user model information such as user model and user activity from the database, as a user auxiliary information feature.

Step 03: The data processed data is mapped into a Vector form required by the model.

Step 04: Perform a vector Embedding operation, including unifying the user portrait information into a vector representation of the same form.

Step 05: In the model training phase, the framework used in the present invention is a deep learning framework based on dnn and rnn. After three-layer vector transformation mapping, the hidden layers are connected to each other, and finally the user feature vector at a certain moment is obtained.

Step 06: The video feature extraction is performed in synchronization with the user information acquisition module, and the video basic feature video id, the video type, the release time, and the like are extracted, and a video or program is represented by the video feature.

Step 07: Adding a video program content introduction in the video feature representation, where text content extraction is involved, the present invention applies the word2vec method to abstractly represent a video program description into a vector form.

Step 08: Input the video feature vector obtained by the video feature extraction into a dot product operating system.

Step 09: Input the user feature vector at time t obtained by the deep learning model training into the dot product operating system.

Step 10: Sort the size by the dot product operation between the vectors to obtain a list of video programs most likely to be of interest to the top n users.

4. Deep learning model

![Image of the real-time recommendation system model structure diagram based on deep learning](image_url)
The structure of the model is the unification of dnn and rnn. On the basis of the multi-layer perceptron, the hidden layer structures are connected to each other, representing the continuity between times, that is, and the time dynamics of displaying the user's interest, real-time. Recommended effect.

01 indicates that the user query record, the viewing record, and the user portrait are expressed as a Vector input into the DNN model after being Embedding.

02 denotes transforming the input vector into a space on another space via a first layer mapping, specifically, using a relu function mapping of one of the excitation functions in the neural network.

03 denotes that the vector obtained by the first layer is transformed into a vector on another space by a second layer mapping, specifically, by using a relu function mapping of one of the excitation functions in the neural network.

04 indicates the hidden layer vector representation of the t-1 moment of the first layer of hidden layer. The neural network is like a black box. How to realize the complex internal interior is impossible to know. It can only be obtained by continuously adjusting parameters or increasing data set training. An optimal interpretable model.

05 denotes the hidden layer vector representation of the t+1 moment of the first layer of the hidden layer. Specifically, it is obtained by the hidden layer learning rule in the RNN network, and the final state transition function can be obtained by training.

06 indicates the hidden layer vector representation of the t-1 moment of the second hidden layer. The neural network is like a black box. How to realize the complex internals is not known. It can only be obtained by continuously adjusting or increasing the data set training. An optimal interpretable model.

07 denotes the hidden layer vector representation of the t+1 moment of the second layer hidden layer. Specifically, it is obtained by the hidden layer learning rule in the RNN network, and the final state transition function can be obtained by training.

08 indicates the hidden layer vector representation of the third layer hidden layer at t-1. The neural network is like a black box. How can we realize the complex internals? We can only get it by constantly adjusting or increasing the data set training. An optimal interpretable model.

09 denotes the hidden layer vector representation of the t+1 moment of the third layer hidden layer. Specifically, it is obtained by the hidden layer learning rule in the RNN network, and the final state transition function can be obtained by training. The training rules for the three hidden layers are the same.

10 represents the user vector that is finally obtained at time t after training by the model.

5. Experimental results

In this paper, three sets of comparison models are selected for comparison, which are probability matrix decomposition model pmf, collaborative improvement recommendation model based on label improvement, itcf, and collaborative filtering recommendation model sdelm based on stack automatic encoder. The probability matrix decomposition model belongs to the classical algorithm in collaborative filtering recommendation; the itcf model introduces the tag word frequency dynamic weight based on the traditional collaborative filtering algorithm, which is still a shallow model in essence; sdelm belongs to the depth model, using the stack The automatic encoder extracts features and predicts the score by using the nearest neighbor optimization algorithm.

Four sets of algorithms were run on the Movielens-1M test set to obtain MSE comparisons of different models. The smaller the MSE, the better the model performance. It can be seen from the following table: In the MovieLens-1M, the performance of the ConvHR model proposed in this paper is also significantly improved compared with the collaborative filtering recommendation model ITCF, which is about 44.89% improved; and the feature is also extracted using the stacked automatic encoder. In this case, ConvHR has improved performance by about 36.97% compared with SDELM. It can be seen that the model can effectively improve the performance of the model by adding auxiliary information of users and movies.
Table 1. Performance comparison of different algorithm models (mse)

| Algorithm name | MovieLens-1M MSE | MovieLens-10M MSE |
|----------------|------------------|-------------------|
| ITCF           | 1.321            | 1.219             |
| SDELM          | 1.175            | 1.098             |
| Model of this paper | **0.792**        | **0.732**         |

6. Conclusion
The concept of deep learning stems from the study of artificial neural networks. Deep learning combines low-level features to form more abstract high-level representations of attribute categories or features to discover valid representations of data, and such relatively short, dense vector representations are called distributed feature representations (also known as embedded representations). This paper mainly studies a real-time recommendation algorithm for smart TV based on deep learning. Recommendations for TV programs based on the deep learning framework of dnn and rnn are applied to the content-based approach in the recommendation system. Although the model currently has a larger performance improvement than other benchmark models, it does not take into account the user and the movie. More links and other information about movie properties. Subsequent work considers from the perspective of the user watching the movie, and adds it as part of the user's feature; in terms of movie attributes, the user comment text, movie picture or even the movie summary part can be used as the movie feature, thereby further improving the accuracy of the model.

References
[1] A. Singhal and J. Srivastava, “Research dataset discovery from research publications using web context,” Web Intell., vol. 15, no. 2, pp. 81–99, 2017.
[2] A. Singhal, R. Kasturi, and J. Srivastava, “DataGopher: Context-based search for research datasets,” in Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration, IEEE IRI 2014, 2014, pp. 749–756.
[3] A. Singhal, “Leveraging open source web resources to improve retrieval of low text content items,” ProQuest Diss. Theses, p. 161, 2014.
[4] A. Singhal, R. Kasturi, V. Sivakumar, and J. Srivastava, “Leveraging Web intelligence for finding interesting research datasets,” in Proceedings-2013 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2013, 2013, vol. 1, pp. 321–328.
[5] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, “Recommender system application developments: A survey,” Decis. Support Syst., vol. 74, pp. 12–32, 2015.
[6] S. Lakshmi and T. Lakshmi, “Recommendation Systems: Issues and challenges,” Int. J. Comput. Sci. Inf. Technol., vol. 5, no. 4, pp. 5771–5772, 2014.