CCCFNet: A Content-Boosted Collaborative Filtering Neural Network for Cross Domain Recommender Systems

Jianxun Lian[†], Fuzheng Zhang[‡], Xing Xie[‡], Guangzhong Sun[†]
[†]University of Science and Technology of China, Hefei, China
[‡]Microsoft Research
lianjx@mail.ustc.edu.cn, {fuzzhang, xingx}@microsoft.com, gzsun@ustc.edu.cn

ABSTRACT

To overcome data sparsity problem, we propose a cross domain recommendation system named CCCFNet which can combine collaborative filtering and content-based filtering in a unified framework. We first introduce a factorization framework to tie CF and content-based filtering together. Then we find that the MAP estimation of this framework can be embedded into a multi-view neural network. Through this neural network embedding the framework can be further extended by advanced deep learning techniques.

Keywords

recommendation system; cross domain; neural network

1. INTRODUCTION

With the boosting of online services, recommendation systems (RSs) are playing an increasingly important role in filtering information for customers. Since most users only connect to a small set of items, data sparsity problem, which usually results in overfitting, becomes a major bottleneck for RSs. This is especially true for newly joined users/items, which is known as the cold start problem. To address this problem, researchers have introduced multiple-domain RSs. They assume that there are some consistent patterns across domains, thus the knowledge from the relatively dense auxiliary domains can enhance the prediction on the target domain.

While there are different types of cross domain cases, we focus on the situation where users are fully aligned, like CMF [3], TCF [2] and MV-DNN [1]. However, these RSs only discuss transferring either collaborative filtering or content-based knowledge. In this paper, we consider two questions: (1)Can we learn to transfer both content-based and collaborative filtering across different domains simultaneously? (2)How can deep learning techniques be applied in cross domain RSs?

To answer these questions, we propose a unified cross domain model. The key idea is to learn content embeddings so that the content-based filtering has the same latent factor formulation as model-based CF. Actually we propagate the CF not only to user-item level but also to user-content level. Since neural networks are well known for their hidden representation learning, we further embed this cross domain model into a unified multi-view neural network. We name the novel neural network as Cross-domain Content-boosted Collaborative Filtering neural NETwork (**CCCFNet**). Experimental results reveal that CCCFNet consistently outperforms several baseline methods.

2. CONTENT-BOOSTED CROSS DOMAIN CF MODEL

Suppose we have two domains: the target domain and the auxiliary domain. In the target domain, we denote the rating matrix as $\mathbf{R}^{(1)} = [r_{ui}]_{N\times M^{(1)}}$ with $r_{ui} \in \{[r_{min}, r_{max}], ?\}$, where ? means a missing value, N and $M^{(1)}$ indicate the number of users and items respectively. The item content is denoted as a content matrix $\mathbf{A}^{(1)} = [a_{ik}]_{M^{(1)}\times T^{(1)}}$, where each row represents an item, $T^{(1)}$ is the number of attributes, $a_{ik} \in [0, 1]$ is the normalized weight on attribute k, and a_{ik}=0 indicates item i does not have attribute k. Our goal is to predict the missing values in $\mathbf{R}^{(1)}$. Similarly, we have $\mathbf{R}^{(2)}$ and $\mathbf{A}^{(2)}$ for the auxiliary domain.

A widely used method in collaborative filtering is latent factor model. It factorizes the rating matrix $\mathbf{R}_{N\times M}$ into two low-rank matrices, $\mathbf{U}_{N\times D}$ and $\mathbf{V}_{M\times D}$, as latent factors for users and items. To incorporate items' content knowledge, we embed the attributes into a shared latent space, and then augment the item latent vector V_j with the weighted average of the embedded representation of attributes.

Formally, for each attribute k, $k \in 1..T$, we denote its embedding representation as $B_k = \langle b_{k1}, b_{k2}, ..., b_{kL}\rangle$. So we have a attribute embedding matrix $\mathbf{B}$ with each row indicates an attribute. The augmented latent vector for item j is denoted as $\widetilde{V}_j = \{V_j, P_j\}$ where $P_j = A_j\mathbf{B}$. The user latent vector is simply extended as $\widetilde{U}_i = \{U_{ia}, U_{ib}\}$, where U_{ia} denotes the original CF part while U_{ib} denotes the content preference part. In this way, content-based and collaborative filtering is combined into a unified factorization model. We conduct factorization both on the target domain and auxiliary domain. To bridge the two domains, we assume the users share the same latent factors across these domains. The parameter set is $\Theta = \{\mathbf{U_a}, \mathbf{U_b}, \mathbf{V^{(1)}}, \mathbf{V^{(2)}}, \mathbf{B^{(1)}}, \mathbf{B^{(2)}}\}$ and we want to maximize the following posterior formulation:

$$p(\Theta|\mathbf{R^1}, \mathbf{R^2}) \propto p(\mathbf{R^1}, \mathbf{R^2}|\Theta)p(\Theta) \qquad (1)$$

WWW 2017 Companion, April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4914-7/17/04.
http://dx.doi.org/10.1145/3041021.3054207

Maximizing Eq.(1) is equivalent to minimizing the following loss function:

$$\begin{aligned}\mathcal{L} = &\frac{1}{2}\sum_{i,j}(r_{ij}^{(1)} - U_{ia}\cdot V_j^{(1)} - U_{ib}\cdot\sum_k a_{jk}^{(1)}B_k^{(1)})^2\\ &+\frac{\lambda_1}{2}\sum_{i,j}(r_{ij}^{(2)} - U_{ia}\cdot V_j^{(2)} - U_{ib}\cdot\sum_k a_{jk}^{(2)}B_k^{(2)})^2\\ &+\frac{\lambda_*}{2}\sum_k(\|\Theta\|)^2\end{aligned} \tag{2}$$

3. NEURAL NETWORK EMBEDDING

Neural networks are well known for their non-linear representation learning ability. Thus we are interested in connecting the proposed cross domain model with neural network. First we represent the original matrix factorization model with a neural network as shown in Figure 1. The two input vectors, i.e. $\mathbf{x}_u$ and $\mathbf{x}_v$, represent the current user and item encoded using 1-of-N and 1-of-M coding separately. In the hidden layer, we use the identity function as the activation function:

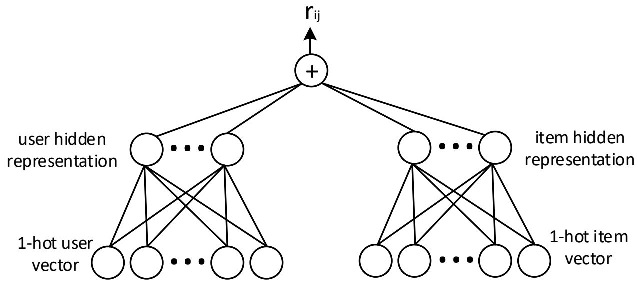

Figure 1: Neural network formulation for matrix factorization.

$$\mathbf{h}_u = f(\mathbf{x}_u\mathbf{W}_u) = \mathbf{x}_u\mathbf{W}_u \quad , \quad \mathbf{h}_v = f(\mathbf{x}_v\mathbf{W}_v) = \mathbf{x}_v\mathbf{W}_v$$

and the output unit is the dot product of two hidden layers:

$$y = \mathbf{h}_u\cdot\mathbf{h}_v = \sum_i h_{ui}\times h_{vi} \tag{3}$$

The learning objective is to minimize reconstruction error between y and r_{ij} as well as the regularization of weights $\mathbf{W_u}, \mathbf{W_v}$. By comparing it with the factorization model, we can see that for this kind of neural network, the weights between hidden layers and input layers are equivalent to the latent factors in the matrix factorization model:

$$\mathbf{W}_u = \mathbf{U}, \quad \mathbf{W}_v = \mathbf{V}$$

Similarly, we can formulate the item attribute latent factorization in a traditional feed-forward neural network. The input for the user side is unchanged, while the input for item set is extended by content features: $\widetilde{\mathbf{x}}_v = \{\mathbf{x}_v, \mathbf{a}_v\}$. We use hyperbolic tangent function as the activation function of attribute hidden nodes:

$$\mathbf{h}_v = f(\mathbf{a}_v\mathbf{W}_v) = tanh(\mathbf{a}_v\mathbf{W}_v)$$

Then the jointly cross domain RS can be formulated as a multi-view neural network learning framework illustrated in Figure 2. Different domains share the user-side sub-network.

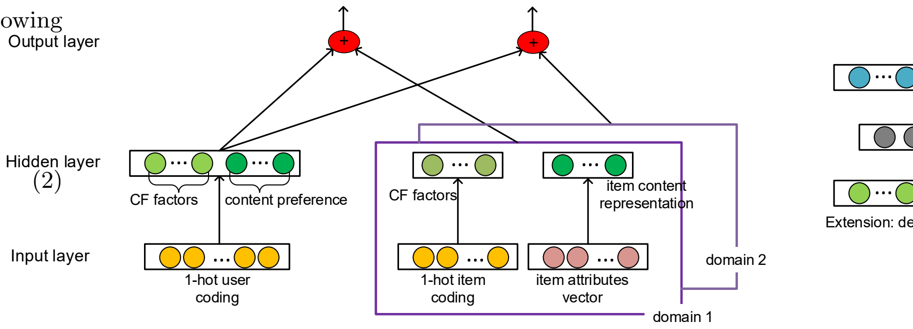

Figure 2: Neural network formualtion for cross domain RS.

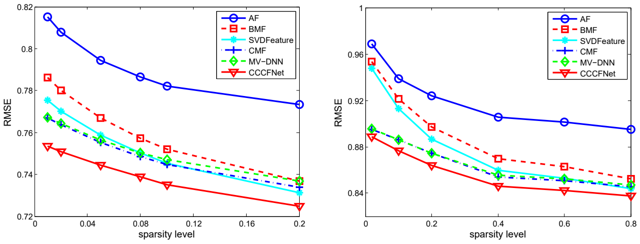

Figure 3: RMSE performance on Douban (left) and MovieLens (right).

4. EXPERIMENTS

We evaluate the performance of CCCFNet on two rating datasets: Douban and MovieLens 20M. Douban is a leading Chinese online community which allows users to record information related to multiple domains, including movies, books, music and activities. For MoiveLens, we split the movies into two disjointed parts to simulate two different domains. We compare our model with Average Filling (AF), Bias Matrix Factorization (BMF), CMF, MV-DNN, and SVDFeature. The results are shown in Figure 3. Our proposed model consistently outperforms the baselines. In the future, we are going to study exploiting advance deep learning techniques, such as RNN for next basket prediction, CNN or auto-encoder for image data embedding.

5. REFERENCES

[1] A. M. Elkahky, Y. Song, and X. He. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pages 278–288, New York, NY, USA, 2015. ACM.

[2] W. Pan, N. N. Liu, E. W. Xiang, and Q. Yang. Transfer learning to predict missing ratings via heterogeneous user feedbacks. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Three*, IJCAI'11, pages 2318–2323. AAAI Press, 2011.

[3] A. P. Singh and G. J. Gordon. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, pages 650–658, New York, NY, USA, 2008. ACM.