Paper One

A Hybrid Collaborative Filtering Model with Deep Structure for Recommender Systems

Motivation

In recent years, with the growing number of choices available online, recommender systems are becoming more and more important. Collaborative Filtering(CF) method is one of the most common used approach in recommender system since its good effect on experiments. However, the rating matrix is often very sparse in reality, causing CF-based methods to perform worse when recommending. Therefore, in order to solve the sparsity problem, the authors want to utilize deep learning method to learn the effective representations of users' and items' through the side information and integrate them into the model.

Related Theory

Collaborative Filtering(CF): It is a method based on the similarity of users-users or items-items. It can be classified into two classes:

- Memory-based CF
- Model-based CF

As for Memory-based CF, take users-users collaborative filtering for example, the algorithm needs users rating matrix as an input.

1. Firstly, computing the similarity of each pair of users based on the Person Coefficient:

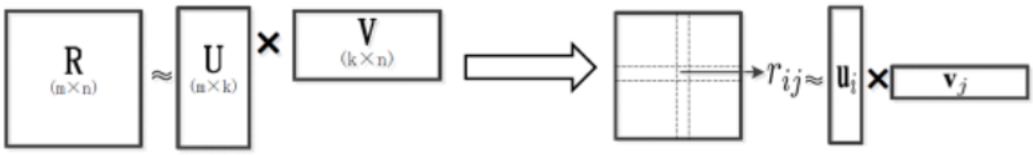
$$\boldsymbol{sim}(\boldsymbol{x},\boldsymbol{y}) = \frac{\sum_{s \in S_{xy}} (\boldsymbol{r}_{xs} - \overline{\boldsymbol{r}_x})(\boldsymbol{r}_{ys} - \overline{\boldsymbol{r}_y})}{\sqrt{\sum_{s \in S_{xy}} (\boldsymbol{r}_{xs} - \overline{\boldsymbol{r}_x})^2} \sqrt{\sum_{s \in S_{xy}} (\boldsymbol{r}_{ys} - \overline{\boldsymbol{r}_y})^2}}$$

2. If we want to recommend one item to the specific user j, we find the top-k users similar to the user j
3. Then, we compute the ratings of the items that user j did not buy:

$$r_{xi} = \frac{\sum_{y \in N} s_{xy} \cdot r_{yi}}{\sum_{y \in N} s_{xy}}$$

4. Finally, we recommend the item with highest rating to user j.

As for Model-based CF, the core is matrix factorization. It utilize matrix factorization method to factorize original rating matrix R into two matrices U and V as the picture shows.(In the training process, we won't consider the missing values.)



The training goal is to minimize the objective function:

$$\arg\min_{\mathbf{U},\mathbf{V}} \mathcal{L}(\mathbf{R}, \mathbf{U}\mathbf{V}^T) + \lambda(||\mathbf{U}||_F^2 + ||\mathbf{V}||_F^2)$$

After finding the optimized U and V, we can compute the missing values according to the cross-product of UV.

Main Theory

Additional Stacked Denoising Autoencoder(aSDAE)

- Autoencoder(AE) is a neural network where f(g(s))=s, s is in the input layer, g is in the hidden layer, f is in the output layer.
- Denoising Autoencoder(DAE) strengthen AE model, introducing noise to the input, which surely makes the result more robust after the encoding.
- Stacked Denosing Autoencoder(SDAE) is a multi-layer neural network based on the Denosing Autoencoder.

Additional Stacked Denosing Autoencoder's goal:

$$\arg\min_{\{\mathbf{W}_l\},\{\mathbf{V}_l\},\{\mathbf{b}_l\}} \alpha||\mathbf{S} - \hat{\mathbf{S}}||_F^2 + (1-\alpha)||\mathbf{X} - \hat{\mathbf{X}}||_F^2 + \lambda(\sum_l ||\mathbf{W}_l||_F^2 + ||\mathbf{V}_l||_F^2),$$

Where S is a sample set and X is the side information set, and W and V are weight matrix.

And latent factor is from the L/2 layer, which is computed as the following equation:

$$\mathbf{h}_l = g(\mathbf{W}_l\mathbf{h}_{l-1} + \mathbf{V}_l\tilde{\mathbf{x}} + \mathbf{b}_l),$$

Hybrid Collaorative Filtering model

Combing the original Model-based CF algorithm with the new method, aSDAE, introduced by this paper, the authors propose a hybrid CF model as the following picture shows.

The general idea is to learn the latent factor representations U and V of users' and items' respectively from the rating matrix and side information matrix. What's more, matrix factorize the rating matrix into U and V. Thus the objective function:

$$\arg\min_{\mathbf{U},\mathbf{V}} \mathcal{L}_R(\mathbf{R}, \mathbf{U}\mathbf{V}^T) + \lambda(||\mathbf{U}||_F^2 + ||\mathbf{V}||_F^2) + \beta\mathscr{L}(\mathbf{S}^{(u)}, \mathbf{X}, \mathbf{U}) + \delta\mathscr{L}(\mathbf{S}^{(i)}, \mathbf{Y}, \mathbf{V}),$$

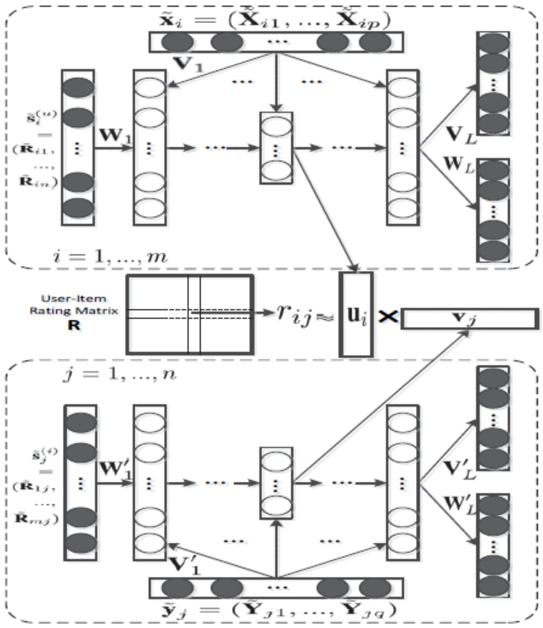
Where R is the rating matrix, X and Y are the side information matrices, U and V are latent factors of users' and items'.

Optimization:

The solution to the objective function is obtained from the stochastic gradient descent method, from which the updating rule is:

$$\mathbf{u}_i = \mathbf{u}_i - \eta \frac{\partial}{\partial \mathbf{u}_i} L(\mathbf{U}, \mathbf{V}),$$
$$\mathbf{v}_j = \mathbf{v}_j - \eta \frac{\partial}{\partial \mathbf{v}_j} L(\mathbf{U}, \mathbf{V}),$$

L(U,V) denotes the objective function.



Overall Hybrid CF Model

Prediction:
After learning the latent factors for each user and item, we get the U and V matrix.
Then we can predict the rating matrix $\overline{\mathrm{R}} = UV^T$, where the missing value can be computed.

Paper Two

Deep Collaborative Filtering via Marginalized Denoising Auto-encoder

Motivation

Similar to the motivation of first paper *A Hybrid Collaborative Filtering Model with Deep Structure for Recommender Systems*, authors want to solve the sparsity problem in CF models. And they also want to learn the effective latent representations through side information. But the difference is that authors of this paper use Probabilistic Matrix Factorization rather than common Matrix Factorization, and they utilize Marginalize Denoising Stacked Autoencoder, which is an enhanced version of SDAE.

Related Theory

Probabilistic Matrix Factorization:

Generally speaking, it introduce probability model into the original Matrix Factorization methods on Collaborative Filtering(CF), making it more robust.

- For a rating matrix R, we assume it obey the Gaussian distribution.
- We also assume matrix R can be factorized into two matrices with the Gaussian distribution.

Therefore, they can be expressed in mathematical formulas as followed:

$$p(R|U,V,\sigma^2) = \prod_{i=1}^{M}\prod_{j=1}^{N}\mathcal{N}(R_{ij}|U_i^T V_j, \sigma^2)^{I_{ij}}$$

$$p(U|\sigma_u^2) = \prod_{i=1}^{M}\mathcal{N}(U_i|0,\sigma_u^2)\ ;\ p(V|\sigma_v^2) = \prod_{j=1}^{N}\mathcal{N}(V_j|0,\sigma_v^2).$$

By introducing the probability models into the matrix R, U and V, the goal of Model-based Collaborative Filtering is:

$$\arg\min_{U,V} E = \|R - UV^\top\|_F^2 + \beta(\|U\|_F^2 + \|V\|_F^2).$$

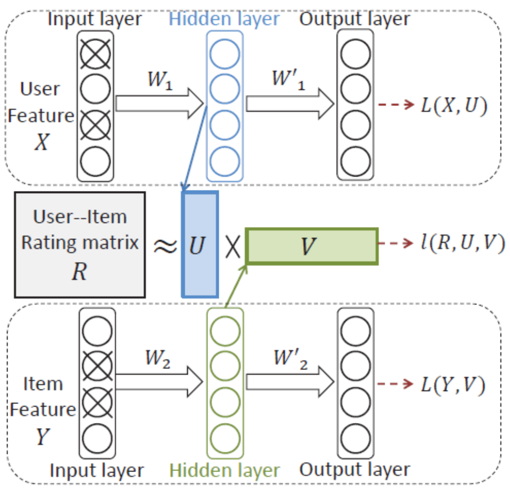
Marginalized Stacked Denoising Autoencoder(mDA)

SDAE has been introduced in the above. Despite their state-of-the-art performance, SDA has the high computational cost of training, as they rely upon the iterative and numerical optimization techniques to learn a large amount of model parameters. Therefore, a method called Marginalized Stacked Denoising Autoencoder is introduced to solve the above problem. Its objective is:

$$\mathcal{L}(W) = \|\bar{X} - W\widetilde{X}\|_F^2,$$

Its main idea is to operate multiple passes of random corruptions over X to obtain $\overline{X}$, where $\overline{X} = [X, ..., X]$ is the c-times repeated version of X, and $\overline{X}$ is the corresponding corrupted version.

Main Theory



As we can see from the above picture, it is very similar to the model in the first paper. The difference is:

1. The matrix factorization is not the common matrix factorization but is introduced with probability model thus becoming probabilistic matrix factorization.
2. The original aSDAE has high computational cost, and now it is replaced as mDAE with lower cost.

Therefore, we can clearly see that the main idea is still similar, thus the objective function and optimization methods are similar as well.
Here are the general algorithms:

***Algorithm 1.** mDA-CF Algorithm*

Input: Rating matrix R, user features X, item features Y, parameters λ, α, β.
Output: Latent factors U, V

Initialize U, V, P_1 and P_2;
while validation error decreases, ***do***
    Update W_1 using (9);
    Update W_2 using (10);
    Update P_1 using (12);
    Update P_2 using (13);
    for each observed R_{ij}, ***do***
        Update u_i using (14);
        Update v_j using (14);
    end for
end while