Social Recommendation Using Probabilistic Matrix Factorization

Presented by:
Weishi Guo
Tianyang Luo

Date: May 6th, 2021

1.Introduction

Typical Recommender Systems

Recommender systems based on Collaborative Filtering(CF):

- User-based CF: Predict user i’s rating on item j by analyzing users that are similar to user i
- Item-based CF: Items that have received similar ratings previously from users are likely to receive similar ratings from future users

Problems of typical Memory-based Collaborative Filtering

1.Sparsity of the user-item rating information:

- Memory-based Collaborative Filtering relies on the previous rating information, however the density of available ratings in commercial recommendation systems is often less than 1% [1].
- CF method can not handle the case when users who have never rated any items

2. In reality, the users' relationships are very important information for the item recommendation. However, traditional CF methods do not consider users' relations.

Social Recommendation

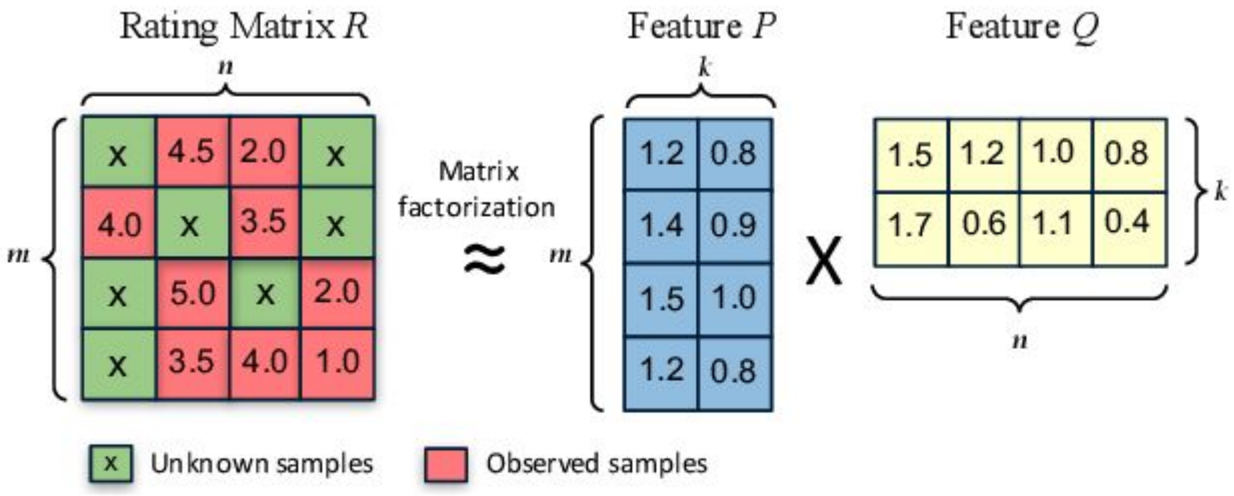
In order to solve previous two problems of traditional memory based CF method, we need to consider both social network structure and user-item rating matrix together in order to make recommendation model to be accurate.

This is called “Social Recommendation”

2.Related Work

Matrix factorization method

We can implement matrix factorization method to predict missing value by R=P*Q



Problems of Matrix Factorization method

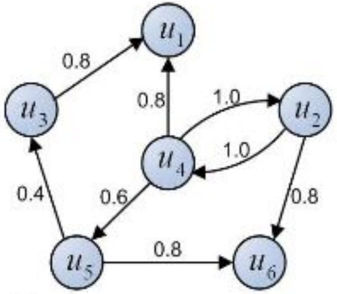
Matrix Factorization method is based on the assumption that users are independent and identically distributed and ignores social activities between users.

In the paper[2] that we read, the authors proposed a **probabilistic** matrix factorization method which can be used to solve the problem.

3.Probabilistic Matrix Factorization Method

Toy example

Social Network Graph is a directed graph with a weight $w_{i,j}$ which tells how much user i trusts or knows user j. We want to predict missing value of user-item matrix by implementing probabilistic matrix factorization based on user-item matrix and social network graph information



(a) Social Network Graph

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	5	2		3		4		
u_2	4	3			5			
u_3	4		2				2	4
u_4								
u_5	5	1	2		4	3		
u_6	4	3		2	4		3	5

(b) User-Item Matrix

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	5	2	**2.5**	3	**4.8**	4	**2.2**	**4.8**
u_2	4	3	**2.4**	**2.9**	5	**4.1**	**2.6**	**4.7**
u_3	4	**1.7**	2	**3.2**	**3.9**	**3.0**	2	4
u_4	**4.8**	**2.1**	**2.7**	**2.6**	**4.7**	**3.8**	**2.4**	**4.9**
u_5	5	1	2	**3.4**	4	3	**1.5**	**4.6**
u_6	4	3	**2.9**	2	4	**3.4**	3	5

(c) Predicted User-Item Matrix

Figure 1: Example for Toy Data

Method to solve the toy example

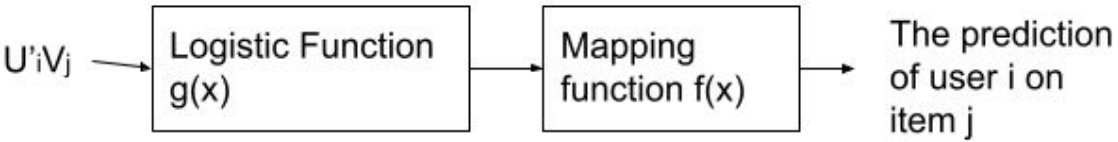
The key idea is that we want to factorize social network graph and user-item matrix by using U^TZ and U^TV.

- U - user latent feature space
  - The column of U denotes the latent feature vector of user i
- V - item latent feature space
  - The column of V denotes the latent feature vector of item i
- Z - factor matrix in the social network graph

$$U = \begin{bmatrix} 1.55 & 1.22 & 0.37 & 0.81 & 0.62 & -0.01 \\ 0.36 & 0.91 & 1.21 & 0.39 & 1.10 & 0.25 \\ 0.59 & 0.20 & 0.14 & 0.83 & 0.27 & 1.51 \\ 0.39 & 1.33 & -0.43 & 0.70 & -0.90 & 0.68 \\ 1.05 & 0.11 & 0.17 & 1.18 & 1.81 & 0.40 \end{bmatrix},$$

$$V = \begin{bmatrix} 1.00 & -0.05 & -0.24 & 0.26 & 1.28 & 0.54 & -0.31 & 0.52 \\ 0.19 & -0.86 & -0.72 & 0.05 & 0.68 & 0.02 & -0.61 & 0.70 \\ 0.49 & 0.09 & -0.05 & -0.62 & 0.12 & 0.08 & 0.02 & 1.60 \\ -0.40 & 0.70 & 0.27 & -0.27 & 0.99 & 0.44 & 0.39 & 0.74 \\ 1.49 & -1.00 & 0.06 & 0.05 & 0.23 & 0.01 & -0.36 & 0.80 \end{bmatrix},$$

Then we can predict the missing value by :



More details of this model is included in [Appendix 1](#)

Matrix Factorization and Posteriors

- Social Network Matrix Posterior:

$$
\begin{aligned}
&p(U, Z|C, \sigma_C^2, \sigma_U^2, \sigma_Z^2) \\
&\quad \propto \; p(C|U, Z, \sigma_C^2)p(U|\sigma_U^2)p(Z|\sigma_Z^2) \\
&\quad = \; \prod_{i=1}^{m}\prod_{k=1}^{m}\mathcal{N}\left[\left(c_{ik}|g(U_i^T Z_k), \sigma_C^2\right)\right]^{I_{ik}^C}
\end{aligned}
$$

- User-item Matrix Posterior:

$$
\begin{aligned}
&p(U, V|R, \sigma_R^2, \sigma_U^2, \sigma_V^2) \\
&\quad \propto \; p(R|U, V, \sigma_R^2)p(U|\sigma_U^2)p(V|\sigma_V^2) \\
&\quad = \; \prod_{i=1}^{m}\prod_{j=1}^{n}\mathcal{N}\left[\left(r_{ij}|g(U_i^T V_j), \sigma_R^2\right)\right]^{I_{ij}^R} \\
&\quad \times \; \prod_{i=1}^{m}\mathcal{N}(U_i|0, \sigma_U^2\mathbf{I}) \times \prod_{j=1}^{n}\mathcal{N}(V_j|0, \sigma_V^2\mathbf{I}).
\end{aligned}
$$

- Social Recommendation Posteriors

$$
\begin{aligned}
&\ln p(U, V, Z|C, R, \sigma_C^2, \sigma_R^2, \sigma_U^2, \sigma_V^2, \sigma_Z^2) = \\
&\quad -\frac{1}{2\sigma_R^2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}^R(r_{ij} - g(U_i^T V_j))^2 \\
&\quad -\frac{1}{2\sigma_C^2}\sum_{i=1}^{m}\sum_{k=1}^{m} I_{ik}^C(c_{ik}^* - g(U_i^T Z_k))^2 \\
&\quad -\frac{1}{2\sigma_U^2}\sum_{i=1}^{m} U_i^T U_i - \frac{1}{2\sigma_V^2}\sum_{j=1}^{n} V_j^T V_j - \frac{1}{2\sigma_Z^2}\sum_{k=1}^{m} Z_k^T Z_k \\
&\quad -\frac{1}{2}\left(\left(\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}^R\right)\ln\sigma_R^2 + \left(\sum_{i=1}^{m}\sum_{k=1}^{m} I_{ik}^C\right)\ln\sigma_C^2\right) \\
&\quad -\frac{1}{2}\left(ml\ln\sigma_U^2 + nl\ln\sigma_V^2 + ml\ln\sigma_Z^2\right) + \mathcal{C}, \qquad (8)
\end{aligned}
$$

Parameters:

- U - user latent feature space
  - Ui denotes the latent feature vector of user i
- V - item latent feature space
- Z - factor matrix in the social network graph
- N(x|μ,σ^2) is the probability density function of Gaussian distribution with mean μ and variance σ.
- C- Social Network Matrix
  - Ci,j denotes the weight of how much user i trust user j
- R - user-item matrix

User-Item Matrix Posterior

- Define the conditional distribution over the observed rating as

$$p(R|U,V,\sigma_R^2) = \prod_{i=1}^{m}\prod_{j=1}^{n}\mathcal{N}\left[\left(r_{ij}|g(U_i^TV_j),\sigma_R^2\right)\right]^{I_{ij}^R},$$

- Place zero -mean spherical gaussian priors on user and movie feature vectors

$$p(U|\sigma_U^2) = \prod_{i=1}^{m}\mathcal{N}(U_i|0,\sigma_U^2\mathbf{I}),$$

$$p(V|\sigma_V^2) = \prod_{j=1}^{n}\mathcal{N}(V_j|0,\sigma_V^2\mathbf{I}).$$

- Finally get :

$$\begin{aligned}
&p(U,V|R,\sigma_R^2,\sigma_U^2,\sigma_V^2)\\
&\quad\propto\ p(R|U,V,\sigma_R^2)p(U|\sigma_U^2)p(V|\sigma_V^2)\\
&\quad=\ \prod_{i=1}^{m}\prod_{j=1}^{n}\mathcal{N}\left[\left(r_{ij}|g(U_i^TV_j),\sigma_R^2\right)\right]^{I_{ij}^R}\\
&\quad\times\ \prod_{i=1}^{m}\mathcal{N}(U_i|0,\sigma_U^2\mathbf{I})\times\prod_{j=1}^{n}\mathcal{N}(V_j|0,\sigma_V^2\mathbf{I}).
\end{aligned}$$

- R - user-item matrix
- V - item latent feature space
- U - user latent feature space
- C- Social Network Matrix
- $N(x|\mu,\sigma^2)$ is the probability density function of Gaussian distribution with mean μ and variance σ.

Recall

Recall that the goal is to predict a missing value in user-item rating matrix

What we calculated are:

- User-item Matrix posterior Pr(U,V| *), where * denotes pre-defined parameters.
- Social Network Matrix posterior Pr(U,Z|*)
- Combine the above together, we calculate Social Recommendation Matrix posterior **Pr(U,V,Z,| C,R,$\boldsymbol{\sigma}^2$)**
    - Reason: In order to reflect that the user's social connections will affect user's rating on items

How can we use Pr(U,V,Z,| C,R,$\boldsymbol{\sigma}^2$) to find U,V,Z?(Next slide)

Matrix Factorization for Social Recommendation

Key ideas:

- After we get the posterior distribution by equation **Pr(U,V,Z,| C,R,$\boldsymbol{\sigma}^2$)**
- Maximizing log-posterior over three latent features with hyperparameters is equivalent to minimize the sum-of-squared-error(MSE)

$$\begin{aligned}\mathcal{L}(R,C,U,V,Z) = \\ \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}^{R}(r_{ij}-g(U_i^T V_j))^2+\frac{\lambda_C}{2}\sum_{i=1}^{m}\sum_{k=1}^{m} I_{ik}^{C}(c_{ik}^{*}-g(U_i^T Z_k))^2 \\ +\frac{\lambda_U}{2}\|U\|_F^2+\frac{\lambda_V}{2}\|V\|_F^2+\frac{\lambda_Z}{2}\|Z\|_F^2, \qquad (9)\end{aligned}$$

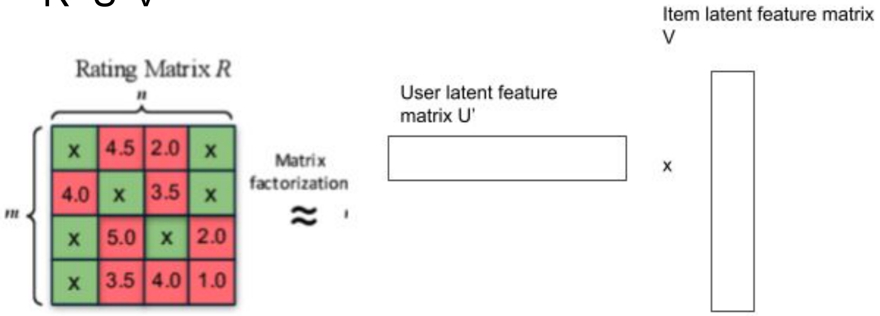
$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial U_i} &= \sum_{j=1}^{n} I_{ij}^{R} g'(U_i^T V_j)(g(U_i^T V_j)-r_{ij})V_j \\ &\quad + \lambda_C\sum_{j=1}^{m} I_{ik}^{C} g'(U_i^T Z_k)(g(U_i^T Z_k)-c_{ik}^{*})Z_k+\lambda_U U_i, \\ \frac{\partial\mathcal{L}}{\partial V_j} &= \sum_{i=1}^{m} I_{ij}^{R} g'(U_i^T V_j)(g(U_i^T V_j)-r_{ij})U_i+\lambda_V V_j, \\ \frac{\partial\mathcal{L}}{\partial Z_k} &= \lambda_C\sum_{i=1}^{m} I_{ik}^{C} g'(U_i^T Z_k)(g(U_i^T Z_k)-c_{ik}^{*})U_i+\lambda_Z Z_k, (10)\end{aligned}$$

- By calculating above gradients, we can train our model to calculate U,V,Z
- Finally we can predict the missing information of user-item rating matrix by calculating U^TV
  - User i's rating on item j is predicted by $U_i^TV_j$

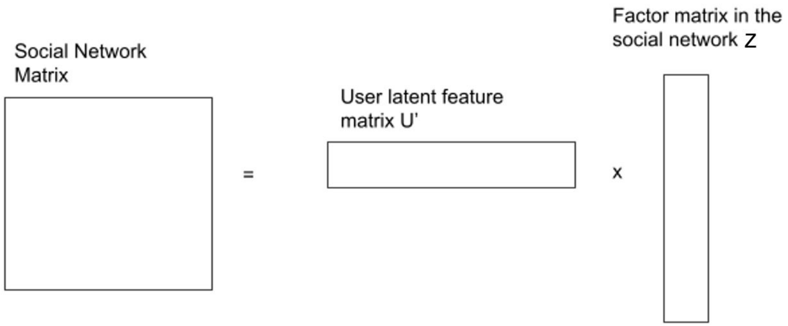
Gradients

$$
\begin{aligned}
\frac{\partial \mathcal{L}}{\partial U_i} &= \sum_{j=1}^{n} I_{ij}^{R} g'(U_i^T V_j)(g(U_i^T V_j) - r_{ij})V_j \\
&+ \lambda_C \sum_{j=1}^{m} I_{ik}^{C} g'(U_i^T Z_k)(g(U_i^T Z_k) - c_{ik}^{*})Z_k + \lambda_U U_i, \\
\frac{\partial \mathcal{L}}{\partial V_j} &= \sum_{i=1}^{m} I_{ij}^{R} g'(U_i^T V_j)(g(U_i^T V_j) - r_{ij})U_i + \lambda_V V_j, \\
\frac{\partial \mathcal{L}}{\partial Z_k} &= \lambda_C \sum_{i=1}^{m} I_{ik}^{C} g'(U_i^T Z_k)(g(U_i^T Z_k) - c_{ik}^{*})U_i + \lambda_Z Z_k, (10)
\end{aligned}
$$

$R=U^TV$



$C=U^TZ$



Train the R,S together by gradients

4.Results

Experiments Set Up

- Data: Epinions Dataset which is a review site
- Experiments:
  - Dataset was splitted into training and testing set
  - Loss function= Mean Absolute Error
- Goal: Compared the testing results with other Matrix Factorization Methods such as: Maximum Margin matrix Factorization(MMMF)

Results

- Test results: Accuracy was improved by around 10% relative to other matrix factorization methods.
- $ƛ_C = \boldsymbol{\sigma}^2_R / \boldsymbol{\sigma}^2_C$ impacts MAE loss
    - $\boldsymbol{\sigma}^2_R$ is the pre-defined variance of user-item rating, $\boldsymbol{\sigma}^2_C$ is the predefined variance of users' trust weight
- When users have more no rating records, the model increase the performance by more than 36.75%
- More efficient compared with other matrix factorization methods.

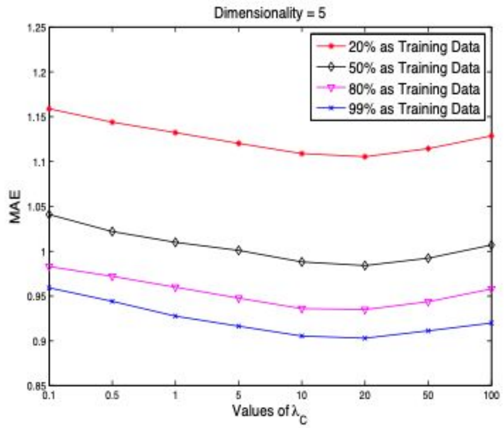
R- User-item rating matrix
C- Social Network Matrix

$\lambda_C = \sigma^2_R / \sigma^2_C$ impacts MAE loss

Recall that: there are some missing information in user-item matrix R and Social Network matrix C. σ_R is the pre-defined variance for R.

In paper, authors found that as λ_C increases, the prediction accuracy increases at first and then decreases.

This coincides with the intuition that purely using the user-item matrix or purely using the social network cannot generate better performance than combine them together



(a) Dimensionality=5

5.Conclusion

Advantages and disadvantages

Advantages:

- Accuracy is higher than other state-of-the art CF recommendation system
- The model considers about the social network influence on user-item matrix
- The model is applicable to other fields such as social search, information retrievals
- Time complexity is very efficient.

Disadvantage:

- Time series: the model does not consider that users' social network may change with time.

Questions?

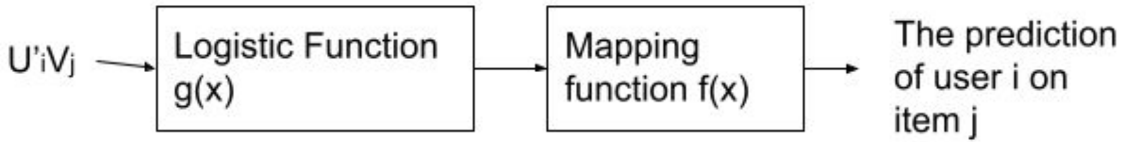
Citations

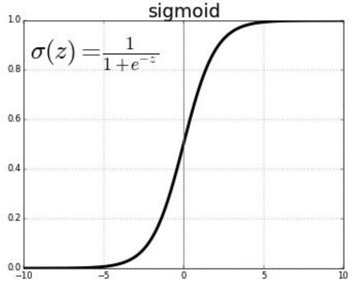
[1] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-based collaborative filtering recommendation algorithms. In WWW ’01: Proceedings of the 10th international conference on World Wide Web, pages 285–295, New York, NY, USA, 2001. ACM.

[2]https://dl.acm.org/doi/10.1145/1458082.1458205

7. Appendix

Appendix 1 Model





The logistic function(sigmoid) looks like below figure. Recall that the rating of user i on item j is in range of [0,5]. However, after the matrix factorization, we get U^TZ and U^TV, however we can not make sure $U_i^TV_j$ is in range of [0,5] as required. But we can know that if $U_i^TV_j$ is high, then the rating should be very high and by using the logistic function g(x), we can narrow the range of $g(U_i^TV_j)$ in the range of (0,1). The higher $g(U_i^TV_j)$ the higher rating will be.

Finally, we can use a mapping function to make $f(g(U_i^TV_j))$ in the range of [0,5]