Tridimensional Regularization with Analysis in Social Network for Recommender Systems

Chaofan Lin
520021911042
ACM Class 2020
Zhiyuan College, Shanghai Jiao Tong University
siriusneo@sjtu.edu.cn

Abstract

Matrix factorization with social regularization is a simple but effective social recommending approach. However, there is still much information in social networks waiting for us to extract. In this paper, we try to improve this model by weighing friends with similarity, influence and familiarity. We perform empirical studies on three different datasets to compare their effectiveness.

1 Introduction

Recommender systems have become a hot research topic recently due to its commercial value and its role in satisfying users' need of information. By the fact that the dimension of truly essential features of an item is small, **Matrix Factorization (MF)** [4] is proposed as a traditional method.

It is intuitive that social network information in websites (e.g. "Follow" network in Douban) can be used to boost recommender systems. Based on the matrix factorization framework, a **Social Regularization (SR)** [7] term is added to utilize the information of friend relationships (edges in the network), based on the idea that it is more possible for friends to have similar ratings.

However, this method does not make the most of information in the network. A basic observation is that people treat their friends or followees differently in the real world. In this paper, we adopt two possible factors abstracted from real social relationship: influence and familiarity, which is able to be analyzed in the network. With these adjustments, **Tridimensional Social Regularization (TriSR)** is proposed by using a tridimensional distance $Tri = (similarity, influence, familiarity)$ to weigh the friendship in social network. The experimental result shows that TriSR has a slight improvement on most of our datasets averagely.

The rest of this paper is organized as follows: Sec.2 provides an overview of related approaches which our work is based on, including MF and SR. Sec.3 mainly presents our TriSR method. Sec.4 shows the empirical research we perform on our self-implemented benchmark. The conclusions are drawn in Sec.5.

2 Related Work

2.1 Matrix Factorization

MF adopts the idea that we can use some latent features (e.g. "Character" and "Plot" for movies) to describe an item. Under this assumption, different tastes of users and various items are all regarded as weigh vectors.

This model introduces a user-matrix $U_{m\times d}$ and an item-matrix $V_{n\times d}$, with U_{ij} being the attention user i pays in feature j, and V_{kj} be the score of item k earns in feature j. It predicts the rating

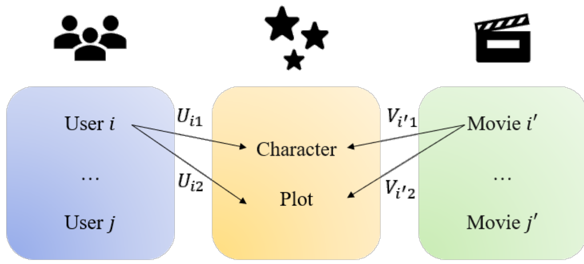


Figure 1: A Matrix Factorization Example with Dimension of Latent Features = 2

matrix $R_{m\times n}$ (i.e. R_{ij} : the rating of user i on item j) by $R \approx UV^T$. The model can be trained by minimizing the distance

$$\min_{U,V} \mathcal{L} = \frac{1}{2}\|R - UV^T\|_F^2$$

Here $\|\cdot\|_F$ denotes the Frobenius norm. By adding regularization, due to the sparsity of R, it is often rewritten as

$$\min_{U,V} \mathcal{L} = \frac{1}{2}\{\sum_{i=1}^{m}\sum_{j=1}^{n} I_{ij}(R_{ij} - U_i V_j^T)^2 + \lambda_1\|U\|_F^2 + \lambda_2\|V\|_F^2\} \tag{1}$$

where m is the number of users, n is the number of items, $\mathcal{N}(i)$ is the neighbours of user i, $I_{ij} \in \{0, 1\}$ indicates whether user i has rated item j. Many social recommending approaches [7, 3, 6] are based on this framework.

2.2 Social Regularization

SR improves MF by utilizing information in social network. Fig 2. shows an exmaple in real world rating website. The key idea is that a movie received many positive feedbacks among your friends will more probably attract you. Therefore it is rational that the user in Fig 2. gives "Star Wars" a high score.

Figure 2: A Social Network Constraint Example

However, the friendship does not always bring consensus, which inspires us to introduce a similarity to weigh our friends. The most widely used method is Pearson Correlation Coefficient (PCC) [1].

$$Sim(u, v) = \frac{\sum_{I_{ui}I_{vi}=1}(R_{ui} - \overline{R_u})(R_{vi} - \overline{R_v})}{\sqrt{\sum_{I_{ui}I_{vi}=1}(R_{ui} - \overline{R_u})^2} \cdot \sqrt{\sum_{I_{ui}I_{vi}=1}(R_{vi} - \overline{R_v})^2}} \tag{2}$$

And the model is proposed by minimizing the loss

$$\min_{U,V} \mathcal{L} = \frac{1}{2}\{\sum_{i=1}^{m}[\sum_{j=1}^{n} I_{ij}(R_{ij} - U_i V_j^T)^2 + \beta \sum_{v\in\mathcal{N}(i)} Sim(i, v)\|U_i - U_v\|_F^2] + \lambda_1\|U\|_F^2 + \lambda_2\|V\|_F^2\} \tag{3}$$

3 Method

Still there are some limitations in SR as it weighs friends only by similarity. We improves SR by further analyzing the network and adding two dimensions: influence and familiarity. To be specific, our TriSR uses $Tri(i,u)$ to weigh the relationship between i and u which is defined as

$$Tri(i,u) = \sqrt{(\alpha Inf(u))^2 + (\beta Sim(i,u))^2 + (\gamma Fam(i,u))^2} \tag{4}$$

where $Inf(u)$ is the influence of u and $Fam(i,u)$ is the familiarity between i and u.

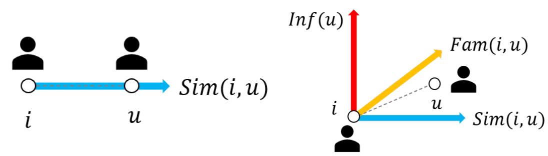


Figure 3: Comparsion between $Sim(i,u)$ (1 dimension) and $Tri(i,u)$ (3 dimensions)

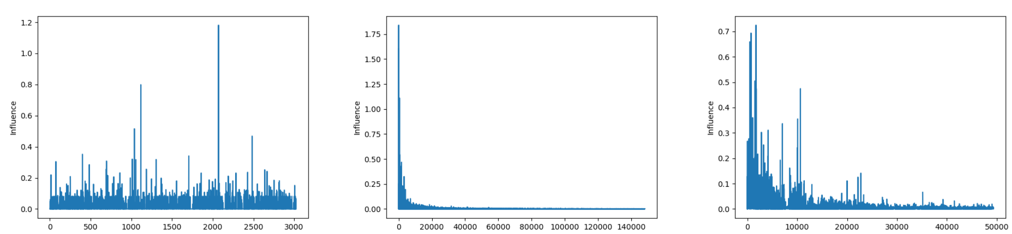
3.1 Influence Analysis

The influence in a network obeys the Power Law Distribution, meaning only a small group of people are the celebrities of this network. Usually a recommendation from a big movie blogger is more convincible than a casual user, which inspires us to adjust our model by influence.

Influence Analysis is a traditional topic and there are many recent approaches like VoteRank [11]. But we choose the classical PageRank [2] due to its high effectiveness in online situation. The influence can be analyzed iteratively through a Markov Process.

$$Inf^t(i) = p \sum_{v \in \mathcal{N}(i)} \frac{Inf^{t-1}(v)}{\deg(v)} + (1-p) \cdot \frac{1}{m} \tag{5}$$

where p is a hyperparameter usually set to 0.85. Fig 4. illustrates the distribution of the influence in our datasets which factually obeys power law distribution.



(a) Douban (b) Dianping (c) Epinions

Figure 4: The Distribution of Influence

3.2 Familiarity

Another noteworthy weight is the familiarity between users in real world because familiars not only have a greater chance to recommend but a higher credibility based on the idea that "familiars will not cheat us".

We adopt a direct familiarity measure [10] according to the proportion of common friends.

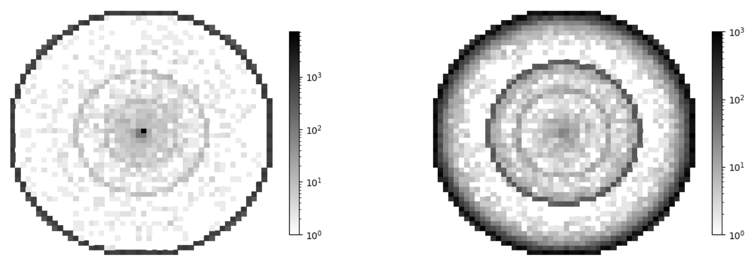
$$Fam(u,v) = \frac{|\mathcal{N}(u) \cap \mathcal{N}(v)|}{\sqrt{|\mathcal{N}(u)|}\sqrt{|\mathcal{N}(v)|}} \tag{6}$$

3.3 Our Model

By combining (2), (4), (5) and (6), the objective function of TriSR can be formulated as

$$\min_{U,V} \mathcal{L} = \frac{1}{2}\{\sum_{i=1}^{m}[\sum_{j=1}^{n} I_{ij}(R_{ij} - U_i V_j^T)^2 + \sum_{v \in \mathcal{N}(i)} Tri(i,v)\|U_i - U_v\|_F^2] + \lambda_1\|U\|_F^2 + \lambda_2\|V\|_F^2\} \quad (7)$$

with three hyperparameters α, β, γ indicates the importance of each dimension in Tri. To visualize the difference between similarly and Tri, we choose the user with the most friends in Dianping, drawing the distribution of the distance between him and his friends. Fig 5. shows that Tri has a smoother distribution.



(a) Similarity (b) Tri

Figure 5: The Distribution of the Distance to the User with the Most Friends in Dianping

4 Experiments

4.1 Datasets

Our experiments are performed on three different datasets: Douban[1] [12], Dianping [2] [5] and Epinions [3] [8]. Each dataset consists of two parts: historical ratings and a undirected social network. All rating scores are scaled to $(0, 5]$.

Table 1: Statistics of Datasets

Dataset	Douban	Dianping	Epinions
Users	3022	147918	49289
Items	6977	11123	139738
Ratings	195493	2149675	664824
Social Relationships	1366	629618	487181

4.2 Metrics and Competitors

We mainly use two popular metrics in the field of recommender systems: Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). They are defined as follows.

$$MAE = \frac{\sum_{(i,j,R_{ij})\in T} |\hat{R}_{ij} - R_{ij}|}{|T|}$$

$$RMSE = \sqrt{\frac{\sum_{(i,j,R_{ij})\in T} (\hat{R}_{ij} - R_{ij})^2}{|T|}}$$

[1]https://www.douban.com
[2]https://www.dianping.com
[3]https://www.epinions.com

where T is the test set with each test rating being a triple $(user, item, score)$.

We evaluate the performance in our self-implemented benchmark (see the code attached). To provide a conprehensive view, we adopt the following methods as competitors.

- **UserMean** predicts the rating $(user, item, score)$ by the average rating scores of $user$.
- **ItemMean** predicts the rating $(user, item, score)$ by the average rated scores of $item$.
- **UserCF** is a Collaborative Filtering algorithm focusing on users. It predicts the rating $(user, item, score)$ by the average rating scores of Top-N most similar users of $user$.
- **ItemCF** [9] is the same with UserCF but it focuses on items.
- **MF** is the matrix factorization method.
- **SR** is a matrix factorization with social regularization based on similarity.
- **TriSR** improves SR with a more detailed three-dimensional social regularization.

After performing 5-fold cross-validation, the parameters are set as follows: On Douban and Dianping, $\lambda_1 = \lambda_2 = 0.01$ for all methods. $\beta = 5$ for SR and $\alpha = 2, \beta = 5, \gamma = 3$ for TriSR. On Epinions, $\lambda_1 = \lambda_2 = 10^{-5}$, $\beta = 0.55$ and $\alpha = 0.25, \beta = 0.5, \gamma = 0.25$. For the dimension of latent features d, we adopt $d = 10$ on Douban and $d = 20$ on Dianping and Epinions.

We test each model 5 times in the benchmark and take the average as the result. On Douban and Dianping, we take 60% data for training and the rest for testing while the rate is 90% in Epinions. We use Adam (at first) and Mini-batch SGD (close to convergence) to training.

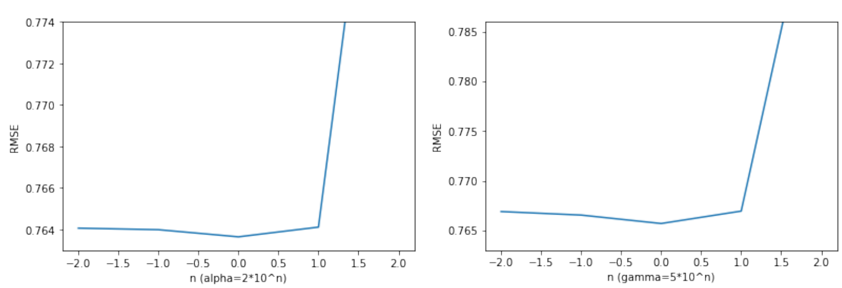
4.3 Results

Table 2: Experiment Results

Dataset	Metrics	UserMean	ItemMean	UserCF	ItemCF	MF	SR	TriSR
Douban	MAE	0.6961	0.6265	0.6874	0.7462	0.5984	0.5973	**0.5959**
	RMSE	0.8675	0.7875	0.8743	0.9440	0.7625	0.7614	**0.7597**
Dianping	MAE	0.5314	0.5127	0.5351	0.5464	0.5048	0.5034	**0.5033**
	RMSE	0.7216	0.6702	0.7044	0.7835	0.6700	**0.6674**	0.6677
Epinions	MAE	0.9369	0.9421	0.9827	0.9873	0.9251	0.9183	**0.9180**
	RMSE	1.2072	1.2243	1.2942	1.3314	1.1920	1.1832	**1.1824**

4.4 Impact of Parameters

Since β in our model is similar with β in SR, here we only do some experiments in α and γ on Douban dataset. As is shown in Fig 6. , a large parameter ($\sim 10^2$) leads to a catastrophe but a small one ($\sim 10^{-2}$) does not lose too much performance because it approximates to the SR model. And it also shows that $\alpha = 2, \gamma = 5$ ($\sim 10^0$) is relatively optimal.



(a) Impact of α (b) Impact of γ

Figure 6: Impact of Parameters

5 Conclusion

The result shows that TriSR has a slightly better performance in most cases, except the RMSE metric in Dianping. A possible explanation might be that Dianping is a platform concentrating more on items, which also explains that ItemMean performs better in this test point.

Based on this idea, we can classify different recommender systems into two categories: user-oriendted and item-oriented, by comparing the performance of UserMean and ItemMean. TriSR has a better performance in a user-oriented platform like Douban and Epinions due to our utilization of users' social information.

However, the improvement is not so much as we except. Further work on item-oriented platforms and directed social network is needed.

References

[1] J. S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI'98, page 4352, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[2] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1):107–117, 1998. Proceedings of the Seventh International World Wide Web Conference.

[3] M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, RecSys '10, page 135142, New York, NY, USA, 2010. Association for Computing Machinery.

[4] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):3037, aug 2009.

[5] H. Li, D. Wu, W. Tang, and N. Mamoulis. Overlapping community regularization for rating prediction in social recommender systems. In *RecSys*, pages 27–34, 2015.

[6] H. Ma, I. King, and M. R. Lyu. Learning to recommend with social trust ensemble. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, page 203210, New York, NY, USA, 2009. Association for Computing Machinery.

[7] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, page 287296, New York, NY, USA, 2011. Association for Computing Machinery.

[8] P. Massa and P. Avesani. Trust-aware recommender systems. In *Proceedings of the 2007 ACM Conference on Recommender Systems*, RecSys '07, page 1724, New York, NY, USA, 2007. Association for Computing Machinery.

[9] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *WWW '01*, 2001.

[10] L. Xiang. Recommendation system practice. page 162, 1 2012.

[11] J.-X. Zhang, D. Chen, Q. Dong, and Z.-D. Zhao. Identifying a set of influential spreaders in complex networks. *Scientific Reports*, 6, 01 2016.

[12] J. Zheng, J. Liu, C. Shi, F. Zhuang, J. Li, and B. Wu. Dual similarity regularization for recommendation. In J. Bailey, L. Khan, T. Washio, G. Dobbie, J. Z. Huang, and R. Wang, editors, *Advances in Knowledge Discovery and Data Mining*, pages 542–554, Cham, 2016. Springer International Publishing.