Collaborative Filtering Based Spammer Detection and User Reputation Estimation in Rating System

Chaojun Liu, Junyu Niu, Fangjun Liu

[1.]Software School, Fudan University, Shanghai, China

[2.]Shanghai Key Laboratory of Intelligent Information Processing, Fudan University, Shanghai, China

huner2011@foxmail.com

Keywords: Rating system, User reputation, Collaborative filtering, Spammer detection

Abstract. Online rating systems play vital role in the recommendation systems and deeply influence the following user choice. It's common occurrence that spammers contaminate the rating systems with malicious rates. However, most of the researchers pay more attention to the accuracy and capability of user preference prediction, rather than the authenticity to rating systems, which provide basis and resource to the recommendation systems. Taking advantage of recommendation algorithm research achievement, we propose a collaborative filtering based spammer detection and user reputation estimation method which perfectly resolves the sparsity problem in huge rating data and promotes poor user preference property of item-based user reputation algorithms. The experiment shows effectiveness of the algorithm both in traditional mode of spammer attack and newly proposed one in this paper, which highly simulates the behavior of real word spammers.

Introduction

In recent years, popular e-commerce enterprises proposed and promoted recommendation systems to improve user experience which cause great mass fervor in recommendation system research. Among the research outcome, collaborating filtering is one of the most popular methods to predict user preference [6-7].

Guang Ling and Irwin King[8] proposed wonderful idea that collaborating filtering can be used to detect spammers in rating systems. They also created a framework of spammer detecting based on collaborating filtering model and tried to implement it with PMF model.

In this paper, I improve several stages of Guang's framework and proposed a new collaborative filtering based spammer detection and user reputation estimation method. The excellent collaborating filtering model SVD++ [9] is chosen to solve the accuracy and sparsity problem in massive data. The experiment shows the effectivity of the method to the existed spamming modes.

In addition, the existed spamming modes are improved to be more aggressive in a group form. And then a new kind of spamming method, which is more simulated to real world spammer behavior, is proposed and the experiment proves it impacts a lot to the existed user reputation estimation method.

Reputation Estimation Algorithm

Problem Formulation. A rating system can be denoted as a directed bipartite graph $G = \{U, I, R\}$ in which $U = \{u_1, u_2, \dots, u_N\}$ represents a set of N users and $I = \{i_1, i_2, \dots, i_M\}$ means item set of size M. R is the set of edges pointing from U to I. R can be denoted as a $N \times M$ matrix in which the entry on i^{th} row j^{th} column r_{ij} means u_i's rating on i_j $(0 < r_{ij} < 1)$. Let the set of items rated by u_i be I_i and the set of users who have rated i_j be U_j. Given the graph mentioned above, u_i's reputation c_i can be determined by the reputation estimation algorithm.

Algorithm Description. Guang's framework divides reputation estimation into tree stages: prediction model, penalty function and link function. In this section, the framework is introduced in detail and the improvement made is included as well.

(1)Prediction Model. Prediction model is a collaborative filtering model that used to predict user ratings. Given a directed bipartite graph $G = \{U, I, R\}$, a prediction model can calculate all entries of R that denote all ratings users made to all possible items, no matter if they really exist. Let the predict model be H, $H(i,j)$ is the prediction of rating made by u_i on i_j. r_{ij} is a Gaussian random variable centered at $H(i,j)$ with variance σ^2.

$$r_{ij} \sim N(H(i,j), \sigma^2)$$

Here we choose the famous collaborative filtering model SVD++[10], which once won the champion of NetFlix Prize[10] to play the role of prediction model.

$$\hat{r}_{ui} = \mu + b_i + b_u + {q_i}^T \left(p_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j \right)$$

In the equation above, $\mu + b_i + b_u$ is baseline prediction where μ is the overall average rating and the parameters b_i and b_u indicate the observed deviations of user u and item i, respectively, from the average. ${q_i}^T p_u$ is called latent factor model which try to deduce item properties from user ratings and user preferences on these properties[11]. $R(u)$ is the set of items user u rated and ${q_i}^T |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j$ implicit feedback model, where y_j is the implicit feature vector.

To describe the unexpectedness of r_{ij}, s_{ij} is defined as the following equation.

$$s_{ij} = \left(r_{ij} - H(i,j) \right)^2$$

As SVD++ is used in this paper, the equation can be turned into

$$s_{ij} = \left(r_{ij} - \mathrm{SVDPlusPlus}(i,j) \right)^2$$

(2)Penalty Function. After getting all unexpectedness of every user on every item, we use penalty function to summarize these unexpectedness into one value to represent overall unexpectedness of a user. Let s_i be the average value of all s_{ij} for user i.

$$s_i = \frac{1}{||I_i||} \sum_{j \in I_i} s_{ij}$$

The higher s_i is, the more likely a user can be a normal user, not a spammer.

(3)Link Function. Link function finally calculate the reputation c_i for u_i. To make $0 < c_i < 1$, Guang use a simple equation

$$c_i = 1 - \frac{s_i}{s_{max}}$$

where s_{max} is the maximum possible value of s_i. Since $0 < r_{ij} < 1$, $s_i < 1$. So we assign s_{max} with 1.

But during observation, we found another possible feature of spammer and introduce frequency factor into link function.

Frequency factor is the count of ratings made by a user in fixed period. We think the spammers tend to have lower cost of their rating behavior. For example, a normal user takes two hours to watch a movie then rates it on the website while a spammer only takes two minutes to rate. Another possibility is that spammers only use their account for a little times until they get a new account.

According to the statistics conducted on the public dataset MovieLens[12] 1M, the average of rating counts is 165.6. 68.7% of all 6040 users rate less than the average and 93.3% of them rates less than 496.8 times, which is three times of the average.

Let the average frequency of user rating be f_{avg}, and then the max frequency should be

$$f_{max} = n f_{avg}$$

where n is an integer. For MovieLens, $n = 3$.

To deal with the single-use accounts of spammers, we also have f_{min} which stands for minimum frequency of user rating. f_{min} id dependent on maturity of dataset.
Therefore the final link function becomes

$$c_i = \begin{cases} 0, & f_i < f_{min} \\ 1 - \frac{s_i}{s_{max}}, & f_{min} \le f_i \le f_{max} \\ 1 - \frac{s_i}{s_{max}} - p_i, & x > f_{max} \end{cases}$$

where p_i is the penalty of frequency which can be calculated by

$$p_i = \frac{f_i - f_{max}}{f_{max} \times p_{max}}$$

and p_{max} is the maximum value of p_i among all users.

Attack Models

It is hard to estimate user reputation algorithm because even human can't determine who are spammers in datasets. One popular method is that researchers play as spammers themselves. Spammers are created to attack the existed datasets by attack code.

Existed Attack Models. There are several attack models used by former researchers.

Random spamming: Spammers rate the items in a random way[13].

Exchange spamming: Spammers copy ratings from normal user and exchange the highest score and lowest score in the record.

Optimistic spamming: Spammers assign half of their ratings using the highest possible score and copy randomly from normal users on the other half items[8].

Pessimistic spamming: Spammers assign half of their ratings using the lowest possible score and copy randomly from normal users on the other half items[8].

Group Version of Existed Models. According to an investigation to 30 Taobao vendors, we find the spammers usually act in concert. So we improve the attack models into group version in tree steps.

(1) Build a spammer group. The size of the group should be big enough to influence normal ratings.

(2) Choose the common attack items for the group and rating them with the same attack rating, the highest rating or the lowest rating.

(3) Choose one or a few normal users to copy from and copy their ratings. The size of normal user chosen should be much less than the group size of spammer community.

Mixed Group Attack. Attack behavior in reality can't be simplex. The spammers often rate highest score for their support items and lowest score for the opposed. So we proposed a new attack model called mixed group attack which is also conducted in three steps. The A and C step is just the same as 3.2 but the B step is that choose the common attack items for the group and rating them with the same attack rating. Give half f them the highest rating and the other half rating is the lowest.

Experiments

Dataset. In this section, I conduct practical experiments to validate algorithms mentioned in this paper. To compare them, I choose public dataset MovieLens [13] as normal ratings and create spammers with the models introduced in section 3.

Evaluation method. The Evaluation method we used in this paper is the Area under the ROC Curve (AUC). ROC curve is abbreviation of receiver operating characteristic curve which use the false positive rate (FPR) and true positive rate (TPR) as x-axis and y-axis value. And every run of the algorithm forms a point on the curve. The bigger AUC is, the better performance is.

Results

Since random attack and exchange attack make no sense in reality, so I only choose pessimistic attack and optimistic to compare with mixed group attack proposed in this paper.

Table-1: AUC of Algorithms against Single Version Attacks

	L1AVG	L2AVG	Guang's	Ours
Pessimistic	0.9745	0.9791	0.9798	0.9879
Optimistic	0.9327	0.9433	0.9632	0.9802

As we can see from Table-1, our method performs well against single version attacks.

Table-2: AUC of Algorithms against Group Version Attacks

	L1AVG	L2AVG	Guang's	Ours
Pessimistic	0.8956	0.8834	0.9237	0.9456
Optimistic	0.8803	0.8815	0.9095	0.9378
Mixed Group	0.8431	0.8479	0.8862	0.9124

Concluded from Table-2, the group version of existed attacks do a much better job than the single one and our mixed group attack model definitely shows greatest threat to all existed algorithms. Moreover, our algorithm performs the most stable in all kind of attacks.

Conclusion

We propose a new SVD++ based spammer detection and user reputation estimation method which perfectly resolves the sparsity problem in huge rating data and promotes poor user preference property of item-based user reputation algorithms.

We also propose group version of existed attack models and mixed group attack model on the basis of improvement. The experiment shows both the effectiveness of the algorithm and the power of the new attack model.

There are some directions worthy of consideration for future study. One direction is to detect the relationship of different spammers and find out the community of them. The second direction is to consider the increasing growth of rating data in real rating systems to find out how to update user reputation real time.

References

[1] Resnick P, Kuw abara K, Zeckhauser R, et al.Reputation System s. Comm unications of the A CM ,2000, 43 (12) : 45258.

[2] Jøsang A, Golbeck J. Challenges for robust trust and reputation systems[C]//Proceedings of the 5th International Workshop on Security and Trust Management (SMT 2009), Saint Malo, France. 2009.

[3] B.C. Chen, J. Guo, B. Tseng, and J. Yang. User reputation in a comment rating environment. In KDD,2011.

[4] R.H. Li, Jerry Yu Xu, X. Huang, et al. Robust Reputation-Based Ranking on Bipartite Rating Networks. In Proceedings of SDM. 2012: 612-623.

[5] R.Guha, Ravi Kumar, Prabhakar Raghavan, and Andrew Tomkins. Propagation of trust and distrust. In Proceedings of the 13th international conference on World Wide Web, WWW'04, pages 403-412, New York, NY, USA, 2004. ACM.

[6] Goldberg D, Nichols D, Oki BM,Terry D，et al. Using collaborative filtering to weave an information tapestry[J].Communications of the ACM.December,1992,35(12):61-70.

[7] Resnick P,Iacovou N,Suchak M,Bergstrom P, Riedl J, et al. GroupLens:an open architecture for collaborative filtering of netnews.In: Proceedings of the ACM CSCW' 94 Conference on Computer-Supported Cooperative Work, 1994, 175-186

[8] L. Guang, Irwin K, Michael R.Lyu. A Unified Framework for Reputation Estimation in Online Rating Systems, IJCAI 2013.

[9] Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model, KDD 2008.

[10] http://www.netflixprize.com.

[11] KOREN Y，BELL R，VOLINSKY C．Matrix factorization techniques for recommender systems[J]．IEEE Computer，2009，42f81：30—37．

[12] http://www.cs.umn.edu/Research/GroupLens.

[13] Robin D. Burke, Bamshad Mobasher, Chad Williams, and Runa Bhaumik. Classification features for attack detection in collaborative recommender systems. In KDD, pages 542-547, 2006.