Finding Group Shilling in Recommendation System[1]

Xue-Feng Su
Computer Science and Technology
Beijing University of Posts and Telecommunications
Beijing 100876, P.R.China
suxuefeng8@hotmail.com

Hua-Jun Zeng
Microsoft Research Asia
49 Zhichun Road,
Beijing 100080, P.R.China
hjzeng@microsoft.com

Zheng Chen
Microsoft Research Asia
49 Zhichun Road,
Beijing 100080, P.R.China
zhengc@microsoft.com

ABSTRACT

In the age of information explosion, recommendation system has been proved effective to cope with information overload in e-commerce area. However, unscrupulous producers shill the systems in many ways to make profit, and it makes the system imprecise and unreliable in a long term. Among many shilling behaviors, a new form of attack, called group shilling, appears and does great harm to the system. Because group shilling users are now well organized and become more hidden among various normal users, it is hard to find them by traditional methods. However, these group shilling users are similar to some extent, for they both shill the target items. We bring out a similarity spreading algorithm to find these group shilling users and protect recommendation system from unfair ratings. In our algorithm, we try to find these cunning group shilling users through propagating similarities from items to users iteratively. The experiment shows our similarity spreading algorithm improves the precision of the system and provides the system a reliable protection.

Categories and Subject Descriptors

H.3.5 [**INFORMATION STORAGE AND RETRIEVAL**]: Online Information Services – *Commercial services, Data sharing, Web-based services.*

General Terms

Algorithms, Experimentation.

Keywords

Collaborative filtering, recommendation system, group shilling

1. INTRODUCTION

Nowadays the e-customers may become confused to find what they need when facing so many commodities on the website. The emergence of recommendation system is of great help to solve this information overload problem and steer customers toward products that might interest them [5] [6]. Many popular online websites like amazon.com have taken good advantage of recommendation system to assist customer effectively.

Recently much attention has been focused on the recommendation system algorithms, but little has been devoted to the protection of the system under attacks. However, some people begin to pay more attention to this problem. J. Konstan [3] thought that "shills" can have a bad influence upon the system and that recommendation systems will take longer than previously expected to self-correct. Dellarocas [2] suggested using controlled anonymity and cluster filtering to avoid unfairly ratings and discrimination. Canny [1] proposed a system in which users' rating information is both kept privacy from the web administrator and other users. He believed this can eliminate the discriminatory from the system side. O'Mahony [4] performs empirical studies of the resistance of the KNN user-user algorithm based on injecting shilling users into the system, and proved the KNN algorithm to be successful in resisting attack. Based on O'Mahony's work, K. Lam [7] added an analysis of item-based algorithm [6] and studied the attack impact on recommendation as well as prediction.

2. GROUP SHILLING AND ALGORITHM

With some simple filtering rules, the system can eliminate most obviously abnormal rating users based on statistics. For example, we can filter most exceptional users considering of their rating times and average rating score. However, the attack tricks keep developing. Shilling users are not isolated yet today, they are well organized and commit swarm and massive attacks after a premeditated planning. We call this multi-members shilling attack as *group shilling*. Group shilling users are different from traditional attackers, for they make some normal ratings besides attacks to conceal their intentions. As to each shilling user's rating record, the target will be adulterated with some normal ratings on other irrespective items. But after groups of users' attack, the target's rating statistics will be changed unconsciously. We call this scenario as *scenario 1* in our paper.

Another more complicated scenario is that users only attack some items in the target set. We observe that some new crime-specialized companies and click-clubs even provide such "services" which can raise sellers' ratings in a short time without being detected by the website administrators. These criminal companies or clubs firstly scramble the collection of shilling targets into different subsets, and then send to their employees or members for attacking. This is *scenario 2* in our paper.

For the 2 scenarios, we propose a similarity spreading algorithm to find these group shilling users. Although each shilling user acts like a normal customer, they have some relations with each other for they all rate the target set abnormally. Based on this fact, we can distinguish group shilling users. Firstly we construct a bipartite graph for the users and items. Each user could be represented by the vector of items he/she rated and vice versa. Thus, we can calculate each two items' similarity with the Pearson measure, and then we spread items' similarity to the users' rating vectors according to formula 1. If two users in the same shilling group both rate target items in a similar way, the similarity of the two users will rise greatly compared with the

WWW 2005, May 10-14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

[1] This work was carried out when the author was visiting at Microsoft Research Asia.

similarity without spreading procedure. Then we make a user-clustering process based on the new similarities. In the last step, we collect each cluster as an abnormal group, and filter those groups whose cluster size and average similarity of the cluster are smaller than the threshold. Now the remaining clusters of users are our group shilling suspects. If we remove these suspects' rating record from data set, we can protect the recommendation system from group shilling attack.

$$weight_i' = weight_i + \sum_{j=1}^{n} sim(\text{item}_i, \text{item}_j) * weight_j (j \neq i) \quad (1)$$

3. EXPERIMENT AND EVALUATION

3.1 Data Set

In our experiment, we use classical KNN user-based algorithm [5] in our recommendation system, and experiment on the Each Movie data set which consists of 259,233 ratings from 2,000 users and 1,623 movies, with every user rating at least 40 movies. We select the first 200 users' rating data as the training set, and the remaining 1,800 users' as the test set. Furthermore, we select 10 ratings of testing user as the user's profile to compare similarity and make prediction. We use *MAE*, mean absolute error, to evaluate how our generation of simulative shilling users affects the recommendation system and the effect of our similarity spreading algorithm.

3.2 Experiment Design

We evaluate our similarity spreading algorithm by comparing 3 results of MAE score. Firstly, we evaluate test set with respect to original train set, and this is the baseline prediction. Secondly, we inject simulated group shilling users with their ratings record into the train set to affect the system, and also get an evaluation of the system. In the third test, we apply our similarity spreading algorithm on the shilling train set, detect and remove the group shilling suspects with their ratings, and then we evaluate the test set based on the filtered train set to see the improvement of the recommendation system's prediction.

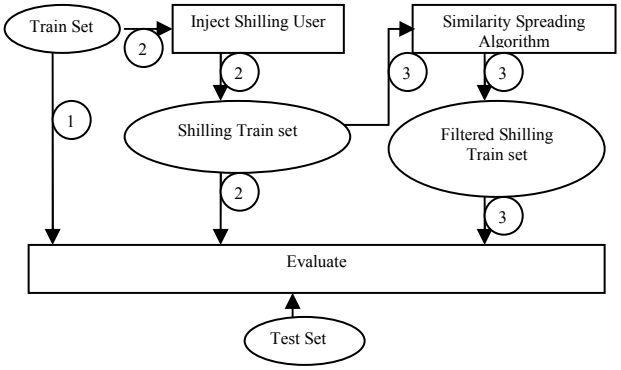

Figure 2. Experiment design

3.3 Group Shilling User Generation

As we know, if too few items are rated by shilling users, the similarities will be scale down too far to be considered by KNN algorithm [4]. Therefore, we suppose every shilling user will rate the first 100 movies at a random score from 1 to 6. Besides these normal 100 ratings, every shilling user in one group will rate 10 movies selected from totally 1,623 movies as the targets. In scenario 1, each group attacks 10 movies and the targets between groups won't be overlapped, while in scenario 2 each group member only attacks 5 movies in target set sized of 10.

3.4 Experimental Results

For scenario 1, we apply our similarity spreading algorithm on the data set adulterated with group shilling users, and get an average MAE score of 1.049, while the MAE without our algorithm, which means the system under attack, is 1.088. For scenario 2, we also lower the MAE score from 1.085 to 1.047.

4. CONCLUSION AND FUTURE WORK

In this paper, we definite a new form of online shilling to the recommendation system, and then present a novel algorithm based on iterative similarity spreading to detect and prevent group shilling. Experiments proved our similarity spreading algorithm can find these abnormal users successfully and improved the prediction of the recommendation system. Furthermore, we verified the similarity spreading procedure plays an important role in our similarity spreading algorithm.

In our next plan, we will study a new similarity spreading method to control the computational complexity of our algorithm and make it adapted for large scaled data sets. Furthermore, the considering of time dimension is another direction of our next research.

5. REFERENCES

[1] J. Canny. Collaborative filtering with privacy via factor analysis. In IEEE Conference on Security and Privacy, May 2002.

[2] C. Dellarocas. Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior. In ACM Conference on Electronic Commerce. 2000.

[3] J. Konstan and J. Riedl. Good ratings gone bad: study shows recommender systems can manipulate users' opinions. CHI 2003 Conference on Human Factors in Computing Systems. April 2003.

[4] M. P. O'Mahony, N. Hurley, N. Kushmerich, and G. Silvestre. Collaborative recommendation: a robustness analysis. ACM Transactions on Internet Technology, 2003.

[5] P. Resnick, N. Iacovou, M. Sushak, P. Bergstrom, and J.Riedl. GroupLens: An open architecture for collaborative filtering of netnews. In Proceedings of CSCW 1994. ACM SIG Computer Supported Cooperative Work. 1994.

[6] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th International World Wide Web Conference (WWW10). Hong Kong. May 2001.

[7] K. Lam, John. Riedl Shilling recommender systems for fun and profit. In Proceedings of the 13th International World Wide Web Conference (WWW2004), New York, May 2004