Robust Evaluation of Binary Collaborative Recommendation under Profile Injection Attack

Qingyun Long
Department of Information and Computer
Shanghai Business School
Shanghai, China
qylong@sbs.edu.cn

Qiaoduo Hu
Labs & Teaching Equipment Center
Shanghai Business School
Shanghai, China
huqd@sbs.edu.cn

***Abstract*—Recommender systems are being improved by every means to be more accurate, more robust, and faster. Collaborative filtering is the mainstream type of recommendation algorithms, and its core is calculating the similarity between users or items based on ratings. Researchers recently found that the binary similarity based solely on who-rated-what rather than actual ratings output more accurate recommendation. We, from robust perspective, evaluated the binary collaborative filtering under multiple types of profile injection attacks on large dataset. Experimental results show binary collaborative filtering is more robust than actual ratings based collaborative filtering in all situations.**

***Index Terms*—recommender system, profile injection attack, binary rating rescaling, robust evaluation, empirical analysis**

I. INTRODUCTION

A. Background

Recommender systems formed from types of information filtering techniques attempt to recommend items (films, television, books, news, images, web pages, etc) that are likely to be interesting to the target users. Due to information overload and fierce business competition, recommendation technology has been the enabling technology even fundamental component of large website. Google, Yahoo, Amazon, Netflix, and many big names, are practitioners of recommender systems. The research of recommendation algorithms has been a common focus for computer science, mathematics, management science, and physics.

For profit, fun, and other intention, recommender systems may be maliciously manipulated. For example, to drive sales, unscrupulous producers may try to influence recommender systems in such a way that their items are recommended to users more often, whether or not they are of high quality [1].Such attacks against recommender systems may cost users time and money by recommending bad items, and system operators profit and reputation by degrading the recommendation accuracy and the users' trust [2]. So the robustness of recommender systems, which is an index of the capacity of defending malicious attacks, is of great significance.

Recently researchers found that the binary similarity based solely on who-rated-what rather than actual ratings output more accurate recommendation [3] [4] [5].If collaborative filtering based on binary similarity can also give fine performance in robustness, current recommendation algorithms and rating scales of recommender systems may need to be re-examined[2] [6]. So, robustness analysis of binary collaborative filtering is very meaningful under this background.

B. Problem Statement

We will evaluate the robustness of binary collaborative filtering under multiple types of profile injection attacks on large dataset, and further explore the meaning and future direction of binary collaborative filtering.

C. Contributions of This Paper

In summary, we make the following contributions:

- Sum up the calculation methods of binary similarity based solely on who-rated-what.
- Evaluate the robustness of binary collaborative filtering under multiple types of profile injection attacks on large dataset.
- Analyze the meaning and future direction of binary collaborative filtering.

D. Paper Organization

The related work is briefly stated in Section Ⅱ .The recommendation algorithms and attack models used in robust evaluation are described in Section Ⅲ. The methods used in experiment design, dataset, parameters setting, and metrics are presented in Section Ⅳ. Then, the detailed experimental results and discussion are given in Section Ⅴ. Finally, the meaning and future direction of binary collaborative filtering are summarized in Section Ⅵ.

II. REALTED WORK

A. Recommender System

Recommender systems are usually classified into three categories: collaborative, content-based and hybrid filtering, based on how recommendations are made [7]. In collaborative filtering, the input of the recommender system is a set of user ratings on items, and the output is the item rating for a given user predicted based upon the ratings given in the user neighborhood or the item neighborhood; so the similarity

978-1-4244-6789-1/10/$26.00 ©2010 IEEE

defined between users or items is crucial, and there are three main approaches: user-based, item-based and model-based approaches. In content-based filtering, recommendation systems recommend an item to a user based upon a description of the item and a profile of the user's interests. In the case of hybrid filtering, both types of information, collaborative and content-based, are exploited, and other information like social and demographic data about users can also be used. Future direction of recommender systems is to improve existing algorithms and ensemble a variety of algorithms that compliment the shortcomings of each other [8] [9].

B. *Profile Injection Attack*

Maliciously manipulations of recommender systems are mainly implemented by profile injection attacks. By the intent of an attacker, attacks can be classified into two categories: "push" attack and "nuke" attack. An attacker may insert profiles to make a product more likely ("push") or less likely ("nuke") to be recommended [10]. Basic strategies include random attack, average attack, bandwagon attack, segment attack, Love/Hate attack, and probe attack [2] [10]. The model of attacks will be described in detail in Section Ⅲ .Profile injection attacks are explored in four directions: capture the attack model and evaluate its power [11] [12]; evaluate the robustness of recommendation algorithms under those attacks [13] [14]; detect those attacks before they cause harm [15] [16] [17] [18] [19] [20]; improve existing algorithms or design new algorithms to be more robust [21] [22].

C. *Evaluating Rocommender System*

Recommender systems have been evaluated in many ways: the user tasks being evaluated, the types of analysis and datasets being used, the ways in which prediction quality is measured, the evaluation of prediction attributes other than quality, and the user-based evaluation of the system, and so on [23].The accuracy and robustness are placed high priority and extensively researched, and there are dozen metrics to measure them [23] [24].

III. ALGORITHMS AND MODELS

A. *User based KNN*

User based k-nearest-neighbor algorithm is the first proposed and widely used collaborative filtering technique. A rating of the target user on an unrated item is predicted based on the ratings given to the same item by the users within the target user's neighborhood, as in

$$p_{u,i}' = \bar{r}_u + \frac{\sum_{v \in V} sim_{u,v}(r_{v,i} - \bar{r}_v)}{\sum_{v \in V} |sim_{u,v}|} \tag{1}$$

where u is the target user; i is the target item; $p_{u,i}'$ is the predicted rating given by u to i; V is the k similar neighbors of u that have rated i; $\bar{r}_u$ and $\bar{r}_v$ are the average ratings over all rated items by u and v ;and $sim_{u,v}$ is the similarity between u and v [25]. Different calculation methods of the similarity may result in different prediction outcome. The improvements in user based KNN algorithm are concentrated on similarity calculation.

1) *Pearson KNN*

In Pearson KNN, the similarity is Pearson correlation coefficient [25]:

$$sim_{u,v} = \frac{\sum_{o \in O}(r_{u,o} - \bar{r}_u)(r_{v,o} - \bar{r}_v)}{\sqrt{\sum_{o \in O}(r_{u,o} - \bar{r}_u)^2}\sqrt{\sum_{o \in O}(r_{v,o} - \bar{r}_v)^2}}. \tag{2}$$

where O is the set of items both rated by u and v.

2) *Binary KNN*

In Binary KNN, the similarity is based solely on who-rated what. There are three processing techniques. Firstly, rescale the rating matrix into 0-1 matrix, which replaces every rating with 1 and sets non-rated user-movie pairs to 1; then use prior KNN methods on that 0-1 matrix [3]. Secondly, project the bipartite network into a monopartite network where two users are connected if they have rated at least one common object, then apply random walk with restart algorithm to calculate the similarity between users [5].Thirdly, the similarity $sim_{u,v}$ is computed as in

$$sim_{u,v} = \frac{\sum_{o \in O}(r_{u,o} - \bar{r}_u)(r_{v,o} - \bar{r}_v)}{\sqrt{\sum_{o \in O}(r_{u,o} - \bar{r}_u)^2}\sqrt{\sum_{o \in O}(r_{v,o} - \bar{r}_v)^2}} \tag{2}$$

$$sim_{u,v} = \frac{n_{u,v} n}{n_u n_v} \tag{3}$$

where $n_{u,v}$ is the number of items both rated by u and v; n_u and n_v are the number of items rated by u and v respectively; n is the total number of ratings [4]. In the following robustness evaluation, we choose the third method.

B. *Attack Models*

A profile injection attack against a recommender system consists of a set of profiles added to the system by the attacker. The generic form of these profiles is shown in *Fig. 1*[10].

Rating for selected items	Rating for filler items	Null for unrated items	Rating for target items

Figure 1. The generic form of a user profile for a profile injection attack

Two metrics in profile injection attack are defined as following:

$$Attack\ size = \frac{number\ of\ injected\ profiles}{number\ of\ total\ profiles} \tag{4}$$

$$Filler\ size = \frac{number\ of\ filler\ items\ in\ injected\ profile}{number\ of\ total\ items} \quad (5)$$

All attack models in our robustness evaluation of binary collaborative filtering belong to push attack category and will be introduced as following.

1) Random Attack

The random attack profiles consist of random ratings assigned to the filler items and maximum rating assigned to the target item. The random ratings are drawn from a normal distribution around the mean value of the ratings the target item has already gotten. The maximum rating is the maximum value of the ratings the target item has already gotten. In this attack model, the set of selected items is empty.

2) Average Attack

The average attack profiles consist of average rating assigned to the filler items and maximum rating assigned to the target item. The average rating is the average value of the ratings the target item has already gotten. In this attack model, the set of selected items also is empty.

3) Bandwagon Attack

The goal of the bandwagon attack is to associate the attacked item with a small number of frequently rated items. The bandwagon attack profiles consist of random ratings assigned to the filler items and maximum rating assigned to the selected high rated items and the target item.

IV. Experimental Methodology

A. Dataset

In the research field of recommender system, there are several public datasets widely used like EachMovie dataset, MovieLens smaller dataset, MovieLens medium dataset, MovieLens larger dataset, Netflix dataset. The binary collaborative filtering was claimed to especially do well in addressing sparse dataset [5]. Moreover, we have limited computation resources. So, we choose to do experiment on MovieLens larger dataset. This dataset consists of 10 million ratings for 10681 movies by 71567 users, and the density is 1.3%. All ratings are integer values between one and five, where one is the lowest and five is the highest. All the users have rated at least 20 movies.

The ratings data are 80%/20% split into training and test data. The split is conducted by running the script named split_ratings.sh which is provided with dataset by GroupLens. The rating is predicted with 5 fold cross validation where we repeat experiment with each training and test set and average the results.

The set of attacked items consists of 10 movies whose ratings distribution matches the overall ratings distribution of all movies. Each movie is attacked as a separate test, and the results are averaged. In each test, a number of attack profiles are generated and inserted into the training set.

B. Metrics

Robustness measures the performance of the system before and after an attack to determine how the attack affects the system as a whole [10]. We are interested not in raw performance, but in the change in performance induced by an attack. This change is measured by Prediction Shift which is defined as in

$$Prediction\ Shift = \frac{\sum_{i \in I_T} \sum_{u \in U_T} (p'_{u,i} - p_{u,i})}{|I_T||U_T|} \quad (6)$$

where I_T is the set of items and U_T is the set of users in the test set; $p'_{u,i}$ represents the prediction after the attack and $p_{u,i}$ before.

C. Parameters setting

In user based KNN, K, the size of neighborhood, need be preset. It is reasonable to set K between 10 and 50 [25].In our experiment, K is preset as 50. Reference [10] found, in profile injection attack, the attack power increased with attack size, but increased very slowly even hardly when attack size was more than 10%; and the attack power also increased sharply with filler size, but decreased slowly when filler size was more than 20%. So, in our experiment, the attack size varies from 2% to 10% and filler size from 5% to 25%. When we explore how the prediction shift varies with the attack size, the filler size is set 5%; and when we explore how the prediction shift varies with the filler size, the attack size is set 4%.In bandwagon attack, we associate the attacked item with top 10 most rated items.

V. Experimental Results and Discussion

Fig. 2, Fig. 3, and *Fig. 4* show the results of comparative experiments examining Pearson KNN and Binary KNN algorithms under average attack, random attack and bandwagon attack, at different attack sizes and filler sizes.

The average attack was shown to be highly effective in prior work [2] [10] [14] and our experimental result also indicated that this was the case. The predict shift of Pearson KNN rose sharply to around 1.9 as the attack size increased to around 5%, but after this point it climbed up gradually; the predict shift of Binary KNN also rose sharply as the attack size increased to around 4%, but after this point it went up slowly (see *Fig. 2 Left*). The predict shift of Pearson KNN decreased gradually as the filler size increased; but the predict shift of Binary KNN rose sharply as the filler size increased to around 10%, nevertheless after this point it dropped off slowly (see *Fig. 2 Right*). Overall, Binary KNN changed moderately and was absolutely robust than Pearson KNN under average attack.

As for random attack, the predict shift of Pearson KNN rose sharply to around 1.8 as the attack size increased to around 6%, but after this point it climbed up gradually; it decreased gradually as the filler size increased to around 10%, nevertheless after this point it dropped off very slightly; the predict shift of Binary KNN changed as same as for average

attack (see *Fig. 3*). Overall, Binary KNN changed moderately and was absolutely robust than Pearson KNN under random attack as attack size varied; but when filler size was more than 10%, robustness of both were very close.

As for bandwagon attack, the predict shift of Pearson KNN and Binary KNN changed as similarly as for average attack (see *Fig. 4*). But the absolute predict shift of Binary KNN is less than Pearson KNN under bandwagon attack; the robustness gap between Pearson KNN and Binary KNN for bandwagon attack is smaller than for average attack.

In a word, Binary KNN is more robust than Pearson KNN under all conditions in our experiment. Compared with experimental results in [10] where the dataset is MovieLens smaller dataset, we infer that Binary KNN is more robust than Pearson KNN especially when the dataset is sparse.

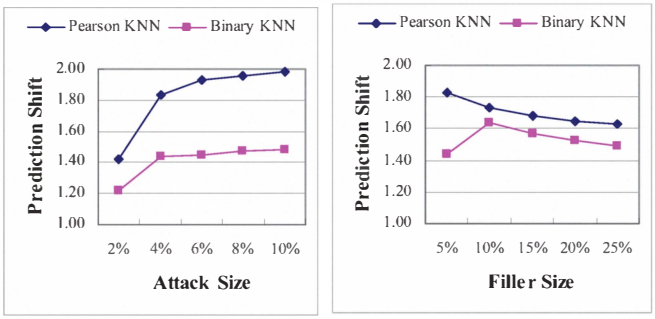

Figure 2. Prediction shift for average attack

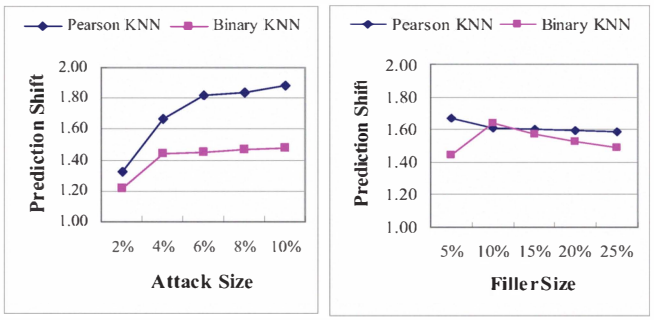

Figure 3. Prediction shift for random attack

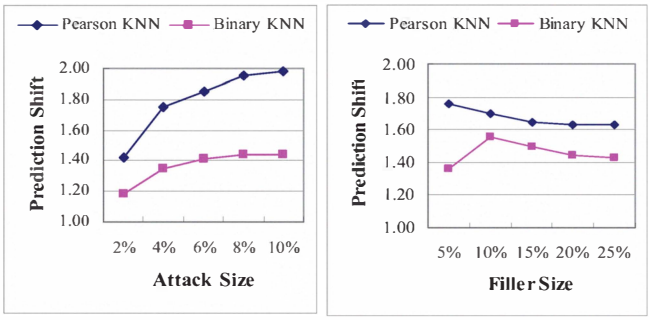

Figure 4. Prediction shift for bandwagon attack

VI. CONCLUSION

Binary collaborative filtering is significant to recommender system. Firstly, it is generally much faster to calculate binary similarity based solely on who-rated-what rather than to calculate prior similarity based on actual ratings, and thus binary collaborative filtering algorithms can save time [5].Secondly, binary collaborative filtering may get more accurate predictions than prior similarity based collaborative filtering especially for sparse data and the datasets that real world recommender systems process are usually very sparse [5].Thirdly, as our experimental results show binary collaborative filtering is more robust than other similarity based collaborative filtering. To be faster, more accurate, more robust are primary objectives of recommender system development. So, binary collaborative filtering gives promising choice.

We will deepen and extend our research in several directions. Firstly, repeat our experiments on multiple datasets with different scales to verify if our findings are general. Secondly, besides the Pearson similarity , compare binary similarity with other similarity like Spearman similarity, vector similarity, refined correlation based similarity. Thirdly, evaluate other metrics of binary collaborative filtering such as coverage, learning rate.

ACKNOWLEDGMENT

This research was supported by Key Disciplines Fund of Shanghai Business School for Business Information Management.

REFERENCES

[1] S. K. Lam, J. Riedl, "Shilling recommender systems for fun and profit," in *WWW'06*, New York, USA.,2004, pp.393-402.

[2] B. Mobasher, R. Burke, R. Bhaumik, J.J. Sandvig, "Attacks and remedies in collaborative recommendation" *IEEE Intelligent Systems*, vol. 22, pp. 56-63, May. 2007.

[3] R. M. Bell, Y. Koren, C. Volinsky. (2010, May 10). *The BellKor solution to the Netflix Prize* [Online]. Available: http://www.netflixprize.com.

[4] M. Piotte , M. Chabbert. (2010, May 10). *The Pragmatic Theory solution to the Netflix Grand Prize* [Online]. Available: http://www.netflixprize.com.

[5] M. Shang, L. Lu, W. Zeng, Y. Zhang, T. Zhou, "Relevance is more significant than correlation: Information filtering on sparse data," *Europhysics Letters*, vol. 88, pp.1-4, Dec. 20049.

[6] D. Cosley, S. K. Lam, I. Albert, J. A. Konstan, J. Riedl, "Is seeing believing? how recommender interfaces affect users' opinions," in *CHI '03*, Florida, USA.,2003, pp. 585-592.

[7] G. Adomavicius, A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, pp.734-749,Jun. 2005.

[8] R. M. Bell, Y. Koren, "Lessons from the Netflix Prize challenge," *SIGKDD Explorations*, vol.9, pp. 75-79, Dec. 2007.

[9] Y. Koren, "Tutorial on recent progress in collaborative filtering," in *RecSys '08*, Lausanne, Switzerland,2008, pp.333.

[10] B. Mobasher, R. Burke, R. Bhaumik,C. Williams, "Toward trustworthy recommender systems:an analysis of attack models and algorithm robustness," *ACM Transactions on Internet Technology*, vol. 7, pp. 23-60, Oct. 2007.

[11] R. Burke, B. Mobasher, R. Bhaumik, "Limited knowledge shilling attacks in collaborative filtering systems," in *IJCAI'05*, Edinburgh, UK.,2005.

[12] R. Burke, B. Mobasher, R. Bhaumik, C. Williams, "Segment-based injection attacks against collaborative filtering recommender systems," in *ICDM'05*, Houston,USA.,2005, pp.577-580.

[13] B. Mobasher, R. Burke, J.J. Sandvig, "Model-based collaborative filtering as a defense against profile injection attacks," in *AAAI'06*, Boston,USA.,2006, pp.1388-1393.

[14] B. Mehta,T. Hofmann,W. Nejdl, "Robust collaborative filtering," in *RecSys'07*, Minneapolis,USA.,2007, pp.49-56.

[15] B. Mobasher, R. Burke, C. Williams, R. Bhaumik, "Analysis and detection of segment-focused attacks against collaborative recommendation," *Lecture Notes in Artificial Intelligence*, vol. 4198, pp. 96-118,2006.

[16] R. Burke, B. Mobasher, C. Williams, R. Bhaumik, "Classification features for attack detection in collaborative recommender systems," in *KDD'06*, Philadelphia, USA.,2006, pp.542-547.

[17] R. Burke, B. Mobasher, C. Williams, R. Bhaumik, "Detecting profile injection attacks in collaborative recommender systems," in *CEC'06*, San Francisco, USA.,2006, pp.23.

[18] R. Bhaumik, C. Williams, B. Mobasher, R. Burke, "Securing collaborative filtering against malicious attacks through anomaly detection," in *ITWP'06*, Boston, USA.,2006.

[19] C. Williams, R. Bhaumik, R. Burke, B. Mobasher, "The impact of attack profile classification on the robustness of collaborative recommendation," in *WEBKDD'06*, Philadelphia, USA.,2006.

[20] B. Mehta, W. Nejdl, "Unsupervised strategies for shilling detection and robust collaborative filtering," *User Modeling and User-Adapted Interaction*, vol. 19, pp. 65-97,Jul. 2008.

[21] J.J. Sandvig, B. Mobasher, R. Burke, "Impact of relevance measures on the robustness and accuracy of collaborative filtering," in *EC-Web'07*, Regensburg, Germany,2007, pp.99-108.

[22] B. Mehta T. Hofmann, "A survey of attack-resistant collaborative filtering algorithms," *IEEE Data Eng. Bull.*, vol. 31, pp. 14-22, 2008.

[23] J. L. Herlocker, J. A. Konstan, L. G. Terveen, J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, pp.5-53, Jan.2004.

[24] M. Mahony, N. Hurley, N. Kushmerick, G. Silvestre, "Collaborative recommendation: A robustness analysis," *ACM Transactions on Internet Technology*, vol. 4, pp.344-377, Nov.2004.

[25] J. L. Herlocker,J. A. Konstan,A. Borchers,J. Riedl, "An algorithmic framework for performing collaborative filtering," in SIGIR'99, New York,USA.,1999,pp.230-237.