Rating Aggregation in Collaborative Filtering Systems

Florent Garcin[1], Boi Faltings[1],
Radu Jurca[2], Nadine Joswig[1]

[1]Artificial Intelligence Lab
Ecole Polytechnique Fédérale de Lausanne
Switzerland
{firstname.lastname}@epfl.ch

[2]Google Inc.
Switzerland
radu.jurca@gmail.com

ABSTRACT

Recommender systems based on user feedback rank items by aggregating users' ratings in order to select those that are ranked highest. Ratings are usually aggregated using a weighted arithmetic mean. However, the mean is quite sensitive to outliers and biases, and thus may not be the most informative aggregate. We compare the accuracy and robustness of three different aggregators: the mean, median and mode. The results show that the median may often be a better choice than the mean, and can significantly improve recommendation accuracy and robustness in collaborative filtering systems.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Collaborative Filtering*

General Terms

Performance, Reliability, Security

1. INTRODUCTION

Recommendation systems use ratings provided by users to recommend products. Recommendations can be uniform for the entire user population, such as hotel recommendations on Tripadvisor.com or product recommendations on Cnet.com, or personalised to the taste of a specific user, such as product recommendations on Amazon.com. Common to both types of systems is that they collect ratings of items from their users, aggregate these ratings and allow users to filter the items that are ranked highest according to these aggregates.

Any collection of ratings is likely to contain outliers or even ratings that have been inserted with the purpose of manipulating the recommendation, therefore it is desirable that the aggregation function should be as robust as possible against them. The most common way of aggregating ratings is by the arithmetic mean, often weighted to take into account similarity or age of ratings. However, one can also consider aggregation using other functions, such as the median or the mode. In unbiased normal distributions, there is little difference between these aggregators. However, it is known that in reality, reviews are often biased [4]. Writing a review or even just leaving a rating requires effort, and since it is voluntary many of these ratings are left by people who have some ulterior motive or extreme opinion. One can thus observe that the distribution of ratings is far from the normal distribution one would expect from an unbiased population of raters. This means that the different ways of aggregating them can give very different results.

In this paper, we consider how the aggregation method influences the accuracy and robustness of the ranking that is obtained as a result. We measure robustness by the fraction of users whose recommendations are likely to be affected by outliers or malicious ratings (hit ratio). We compare three different ways of aggregating n numerical ratings $r_1, ..., r_n$, ordered in increasing order, using different forms of averaging: the *mean*, the *median* and the *mode*.

We present results of empirical studies. We report on experiments with the MovieLens data that show that the median may also be helpful to defend against outliers and malicious attacks. We observe that the three notions of average differ significantly. In particular, the mode and median tend to be more robust to outliers and biased reviews but also result in much higher recommendation accuracy than the mean, and thus may be more informative for a user.

2. RELATED WORK

There are many recommendation systems such as Netflix[1] that follow the model we assume in this paper. In many cases, recommendations are made based on purchase data, assuming that a purchase counts as a positive vote for an item. Examples of such systems are found at Amazon[2] and many other e-commerce retailers. In such systems, ratings are just binary and so their aggregation is quite straightforward. Other mechanisms, like the one of Slashdot[3], use discrete values for rating information and define clear rules describing how sets of feedback are mapped to rankings. We therefore focus on systems that use explicit user rating.

The robustness of recommendation mechanisms has been an important concern of the research community recently.

RecSys'09, October 22–25, 2009, New York, New York, USA.
Copyright 2009 ACM 978-1-60558-435-5/09/10 ...$10.00.

[1] www.netflix.com
[2] www.amazon.com
[3] www.slashdot.org

Mobasher et al [6] have investigated methods for manipulating recommendations by inserting malicious feedback reports and show that such attacks are surprisingly easy to carry out in collaborative filtering systems. O'Mahony et al [7] have shown that while the impact can be somewhat mitigated by using different similarity metrics, this cannot significantly reduce the effectiveness of attacks. Walsh and Sirer [10] have shown that a collaborative-filtering-like mechanism incentivizes users to report feedback truthfully in order to receive the best possible recommendations themselves.

Garcin et al [2] analyse in the context of reputation systems how to aggregate feedback ratings into a single value. They consider different ways of aggregating ratings with respects to three criteria: informativeness, robustness and strategyproofness. On all these criteria, they show that the mean seems to be the worst way of aggregating ratings. The median is radically more robust and has the advantage of being strategyproof.

Resnick and Sami [8] apply a reputation mechanism to the problem of manipulation in collaborative filtering systems. They propose an *influence limiter* where users' ratings are weighted by their reputation and only users that gain a reputation for truthful feedback are given significant influence on the rankings. However, in [9] they show that the robustness thus achieved comes at a high cost.

Mehta et al [5] investigate robust collaborative filtering mechanisms using model-based algorithms. They adapt robust statistical methods and present a Robust Matrix Factorisation algorithm that can produce stable recommendations in the presence of spam and noise.

In this paper, we focus on an empirical study of how aggregation of ratings influences properties of the ranking. This issue is complementary to the techniques for obtaining honest ratings themselves, and our analysis is orthogonal to the existing works on robustness of feedback systems.

3. EMPIRICAL STUDY

Our main results are based on an empirical study of a collaborative filtering system using the MovieLens[4] data set. The data set contains 1682 movies rated by 943 users. 100,000 ratings ranging from 1 to 5 were given by these users. Each user rated at least 20 movies. We constructed a user-based collaborative filtering system based on the k-Nearest Neighbour algorithm as outlined in [6].

Given a user u and a target item i for which the system must offer a recommendation, the algorithm first computes the k most similar users to u (neighbours of u) based on the available ratings. The similarity between users u and v is computed using Pearson's correlation coefficient:

$$psim_{u,v} = \frac{\sum_{i=1}^{n}(r_{u,i} - \overline{r}_u) \cdot (r_{v,i} - \overline{r}_v)}{\sqrt{\sum_{i=1}^{n}(r_{u,i} - \overline{r}_u)^2} \cdot \sqrt{\sum_{i=1}^{n}(r_{v,i} - \overline{r}_v)^2}}, \qquad (1)$$

where $r_{u,i}$ and $r_{v,i}$ are the ratings of some item i for u and v respectively, and $\overline{r}_u$ and $\overline{r}_v$ are the average ratings of u and v over the set of items.

We implemented a second similarity metric which takes into account how many items two users rated in common. We used the similarity metric proposed in [1] which is the Pearson's correlation coefficient weighted with the Jaccard similarity measure, defined as the fraction of items rated in common by both users:

$$jpsim_{u,v} = \frac{|\{R_u \cap R_v\}|}{|\{R_u \cup R_v\}|} * psim_{u,v}, \qquad (2)$$

where R_i is the set of all items rated by user i. We call this the *Jaccard-weighted similarity metric.*

We consider the set V of the 20 most similar neighbours of u that have rated at least 5 movies in common with u. The predicted rating for an item i not yet rated by user u is computed by aggregating the ratings of the u's neighbours for item i. We consider the following aggregation rules:

- the mean of the neighbouring ratings weighted by their similarity:

$$pred_{u,i} = \overline{r}_u + \frac{\sum_{v \in V} sim_{u,v}(r_{v,i} - \overline{r}_v)}{\sum_{v \in V} |sim_{u,v}|}, \qquad (3)$$

  where $sim_{u,v}$ is either $psim_{u,v}$ or $jpsim_{u,v}$. This is the aggregator commonly used in collaborative filtering today, and the one used in [6].

- the median of the neighbouring ratings weighted by their similarity, with a tie-breaking rule that prefers items with a higher number of ratings in the overall database. Assuming that the neighbours are ordered by increasing rating, the median is the rating r_i given by the smallest user i such that

$$\sum_{j=1}^{i} sim_{n_j,u} \geq \sum_{j=i+1}^{n} sim_{n_j,u} \qquad (4)$$

  This tie-breaking rule has the disadvantage of favoring very popular items. These have a high chance of being liked, and so result in high accuracy of the recommender, but the recommendation is not as valuable as a more diverse one.

- the weighted median of the neighbouring ratings, but with a tie-breaking rule that prefers less controversial items. We measure controversy by the minimal percentage of additional neighbours with similarity 1 that would be required to change the aggregate. This tie-breaking rule does not favour highly rated items and thus results in more valuable recommendations.

- the mode of the neighbouring ratings weighted by their similarity, with a tie-breaking rule that prefers items with a higher number of ratings overall. Again assuming that neighbours are ordered by increasing rating, the mode is the smallest rating r such that

$$\sum_{\{j|r_j=i\}} sim_{n_j,u} \geq \sum_{\{j|r_j=k\}} sim_{n_j,u} \quad \forall k \neq i \qquad (5)$$

- the weighted mode of the neighbouring ratings, with a tie-breaking rule that prefers less controversial items. Similar to the median, we measure controversy by the minimal percentage of additional neighbours with similarity 1 that would be required to change the aggregate.

[4] http://www.grouplens.org/node/73

Table 1: The mean-average error (MAE) for different aggregators and similarities.

Aggregator	MAE Pearson	MAE Jaccard
weighted mean	0.5810	0.6216
weighted median	0.6310	0.6707
weighted mode	0.6930	0.7238

3.1 Recommendation accuracy

We evaluate the quality of the different algorithms for computing the predictions using precision, recall and the resulting F1 metric [3]. The mean-average error (MAE) is difficult to compare since both the median and mode use only a restricted set of values. Table 1 shows the mean-average error for the three aggregators. As expected, the MAE is lower for the mean since this aggregator minimises this measure. Therefore, the MAE is not indicative of the actual performance of the recommender system.

We consider as the set T of relevant target items the items that a user has rated at least as high as the 20th best (i.e. the 20 top-rated items plus any others that are tied for membership in that set). Let k be the size of this set.

To evaluate recommendation recall, we adopt a leave-one-out method where we iteratively consider one of the target items as unrated, compute a set of k recommendations excluding already rated items, and then check if the target item is among them. Since the ranking of items stays the same, we do this in a single run by considering the set R of the top l recommendations, where $l+1$ is the rank of the kth non-target item. This works because an item will appear in the top k recommendations when all other target items are excluded if and only if it has at most $k-1$ non-target items that are ranked higher than itself, i.e. if it is in the maximal set of recommendations that just excludes the kth non-target item.

Using this "expanded" set of recommendations, we can then compute the recall as the fraction of the target set contained in the recommendations, and the precision as the fraction of the recommendations that are in the target set.

We define:

- *precision* as the fraction of R that is also in T, i.e. $\frac{|R \cap T|}{|R|}$.
- *recall* as the probability that an item in the target set will be found within the k highest ranked items when all other already rated items are excluded, given as $\frac{|R \cap T|}{|T|}$.
- the *F1 metric* as in [3]:

$$F1 = \frac{2 * precision * recall}{precision + recall} \qquad (6)$$

Tables 2 and 3 show that the mean is by far not the best aggregator of recommendation scores; both precision and recall are significantly higher when the median is used. The difference is more marked for the Pearson similarity metric, which is more likely to include outliers among the neighbours, and it seems that the robustness of the median allows the recommender to take advantage of the Pearson metric in a stronger way than when the mean is used. The mode has the best precision and recall when the Pearson similarity and the controversy tie-breaking rule of ratings is used.

Table 2: Recommendation accuracy for the Pearson similarity metric.

Aggregator	Precision	Recall	F1
weighted mean	0.0962	0.0656	0.0780
weighted median, #ratings	0.3090	0.2761	0.2916
weighted median, controversy	0.2320	0.1848	0.2057
weighted mode, #ratings	0.2916	0.2526	0.2707
weighted mode, controversy	0.2535	0.2024	0.2251

Table 3: Recommendation accuracy for the Jaccard-weighted similarity metric.

Aggregator	Precision	Recall	F1
weighted mean	0.1228	0.0905	0.1042
weighted median, #ratings	0.3340	0.3342	0.3341
weighted median, controversy	0.2683	0.2486	0.2581
weighted mode, #ratings	0.3072	0.2950	0.3010
weighted mode, controversy	0.2660	0.2222	0.2421

However, it is not the case for the Jaccard-weighted similarity metric. The median outperforms the mode whatever the tie-breaking rule.

3.2 Resilience against attack

Since recommendations are different for each user, we can no longer give a single number of ratings required to change this ranking. Instead, we consider the robustness of the average recommendation received by each user. We considered scenarios where an attacker wants to push an item into users' recommendations by inserting fake user profiles that provide high ratings for that item. In particular, we implemented the *average attacks* described in [6] with 144 (15%) attack profiles. We characterise robustness by the *hit ratio*, defined as the percentage of times that the promoted item is recommended in a recommendation list as a result of the attack. We randomly select 20 target items on which we perform the attack.

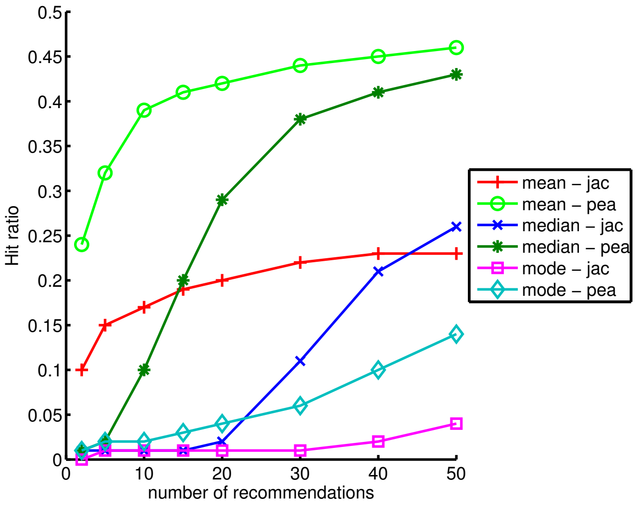

Figure 1: The hit ratio as a function of the size of the number of recommended items; median and mode use the number of ratings as a tiebreaking rule.

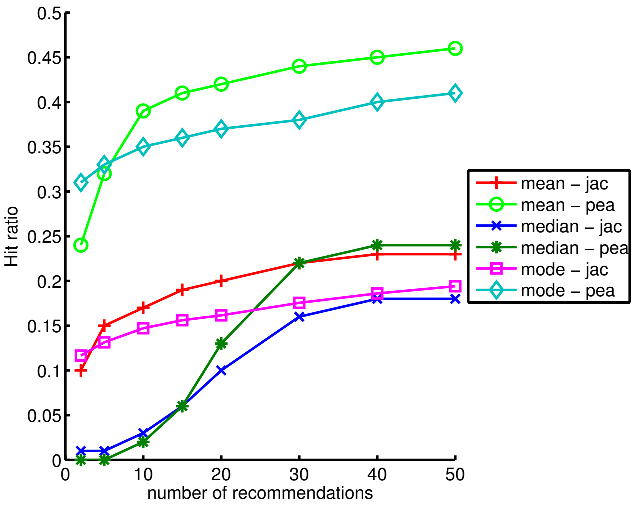

Figure 2: The hit ratio as a function of the size of the number of recommended items; median and mode use the controversy of ratings as a tiebreaking rule.

Figures 1 and 2 show the hit ratios as a function of the number of recommendations. We can see that when the number of recommended items is small, aggregating ratings by the median makes the recommender very resilient against attack, and it remains more resilient than the mean even when many recommendations are given. This again shows the much greater robustness of the median as a rating aggregator. Interestingly, the mode is the most robust aggregator when the tie-breaking rule is based on the number of ratings. However, it performs poorly when the tie-breaking rule uses the controversy of ratings. Since this tie-breaking rule may be more desirable in practice to obtain novel recommendations, the median seems to be the best choice.

4. CONCLUSION

Most recommendation systems aggregate user ratings to establish a ranking of alternatives and recommend the highest-ranked items. It has been common to use a weighted or unweighted arithmetic mean as an aggregation function. However, there are other choices that may produce better results in certain circumstances, and it is surprising that this question has not attracted more attention so far.

We considered three different ways of aggregating ratings: the mean, median and mode, with appropriate weighting when required. If ratings were unbiased and normally distributed, the different notions would not differ much. However, when ratings are collected from a population of users there are many biases [4], and the three methods give very different results.

Theoretical analysis [2] of the breakdown point already points to higher robustness of median and mode to outlier ratings. This seems to be quite important for user-based collaborative filtering recommendation systems. We observe that the median and mode both result in dramatically higher recommendation accuracy than the arithmetic mean, and we conjecture that this is due to the greater robustness of these aggregators against outlier ratings. Another indication of this is that the median seems to largely solve the problem of shilling attacks. However, we note that the mode is somewhat brittle and its performance is very dependent on the tie-breaking rule and the similarity metric. We also note that the mean-average error seems to be a very poor indicator of actual recommendation performance as characterised by notions of precision and recall.

Interestingly, in item-based collaborative filtering the differences between different aggregators are much smaller. While the mean is still the worst performing aggregator, it is only about 2% worse than the median. Furthermore, there is no noticable difference in the hit ratio in response to shilling attacks. We explain this observation by the fact that item-based CF aggregates ratings of the same user, which are less likely to contain outliers for most users.

Another interesting aspect is that the median is a *strategyproof* aggregation rule [2], meaning that for a rater who wants to obtain an aggregated result as close as possible to her own rating, it is best to report this rating truthfully. If users are aware of this fact, they may report less outlying ratings, and thus further increase the quality of information and recommendations that can be provided.

Of all three aggregators, the mean seems to be the worst way of aggregating rankings: it changes the most frequently, it is the least robust, and it is not strategyproof. We thus suggest to study alternative ways of aggregating rankings to both improve the accuracy and the robustness of collaborative filtering recommender systems, since it seems that both can be improved dramatically.

5. REFERENCES

[1] L. Candillier, F. Meyer, and F. Fessant. Designing specific weighted similarity measures to improve collaborative filtering systems. In *ICDM '08*, 2008.

[2] F. Garcin, B. Faltings, and R. Jurca. Aggregating reputation feedback. In *ICORE 09*, 2009.

[3] J. Herlocker, J. Konstan, L. Terveen, and J. Riedl. Evaluating Collaborative Filtering Recommender Systems. *ACM Trans. on Inf. Sys.*, 22:5 – 53, 2004.

[4] R. Jurca, F. Garcin, A. Talwar, and B. Faltings. Reporting incentives and biases in online review forums. *ACM Transaction on the Web.* to appear.

[5] B. Mehta, T. Hofmann, and W. Nejdl. Robust Collaborative Filtering. In *RecSys 07*, 2007.

[6] B. Mobasher, R. Burke, R. Bhaumik, and C. Williams. Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Trans. Internet Technol.*, 7(4):23, 2007.

[7] M. O'Mahony, N. Hurley, N. Kushmerick, and G. Silvestre. Collaborative recommendation: A robustness analysis. *ACM Trans. Internet Technol.*, 4(4):344–377, 2004.

[8] P. Resnick and R. Sami. The influence limiter: provably manipulation-resistant recommender systems. In *RecSys 07*, 2007.

[9] P. Resnick and R. Sami. The information cost of manipulation-resistance in recommender systems. In *RecSys 08*, 2008.

[10] K. Walsh and E. G. Sirer. Fighting peer-to-peer spam and decoys with object reputation. In *P2PECON 05*, 2005.