Recommendation: A Less Explored Killer-App of Uncertainty?

Seung-won Hwang and Jong-won Roh

POSTECH, Pohang, Korea
{swhwang,nbanoh}@postech.ac.kr

Due to the unprecedented amount of information available, it is becoming more and more important to provide personalized recommendations on data, based on past user feedbacks. However, available user feedbacks or ratings are extremely sparse, which motivates the needs for rating prediction. The most widely adopted solution has been collaborative filtering, which (1) identifies "neighboring" users with similar tastes and (2) aggregates their ratings to predict the ratings of the given user. However, while each of such aggregation involves varying levels of uncertainty, *e.g.*, depending on the distribution of ratings aggregated, which has not been systematically considered in recommendation, though recent study suggests such consideration can boost prediction accuracy. To consider uncertainty in rating prediction, this paper reformulates the collaborative filtering problem as aggregating community ratings into multiple predicted ratings with varying levels of certainty, based on which we identify top-k results with both high confidence and rating. We empirically study the accuracy of our proposed framework, over a classical collaborative filtering system.

1 Introduction

Recommender system has been widely adopted for filtering an overwhelming amount of data into a small set of recommended items of the specific user's interests. Such items are typically decided by predicting the *rating* of this specific user on the items, by aggregating past ratings on similar items (*i.e.*, content-based approach) or ratings on this specific item by other users with similar tastes (*i.e.*, collaborative approach), as Example 1 illustrates.

Example 1 (Rating prediction). Consider a user debating which movie to watch. To help her, we need to predict her rating on the movies she is yet to watch, *e.g.*, '007 Quantum of Solace'. For such prediction, we can either aggregate her ratings on previous 007 movies (*i.e.*, content-based recommendation) or the ratings of other users with similar tastes (*i.e.*, collaborative recommendation).

Toward this goal, many systems have been proposed for accurate aggregation of ratings and its efficient computation. Typically, in collaborative filtering [1], for such aggregation, k "neighbors" with the closest tastes are identified, then their ratings are averaged with weights proportional to the similarity, to favor ratings from the neighbors with higher similarity. Once the rating is predicted,

Dagstuhl Seminar Proceedings 08421
Uncertainty Management in Information Systems
http://drops.dagstuhl.de/opus/volltexte/2009/1933

recommender systems rely solely on these predicted ratings, to recommend the items with the highest predicted rating.

However, these predicted values are associated with varying levels of uncertainty, which are not typically considered in existing systems. For instance, the same predicted rating '3' could be obtained from $k = 2$ neighbors unanimously rating '3' on the item (*i.e.*, high certainty), or from strongly disagreeing neighbors, with one rating '1' and another rating '5' (*i.e.*, low certainty).

Recent work [2] studied how such *variance* of neighbor ratings affects the quality of recommendation. According to the study, the rating accuracy can be improved by recommending only the items with both high rating and certainty.

Motivated by this finding, this paper studies how to systematically factor in the uncertainty for rating prediction. More specifically, instead of generating a single prediction value with no notion of uncertainty attached, we propose to generate multiple rating predictions with varying levels of uncertainty associated. We then abstract rating prediction problem as a uncertain data ranking problem, which has been extensively studied lately [3–6]. We stress that, our approach is more systematic than the proposed heuristics in [2], simply filtering uncertain predictions with variance over some user-specific threshold or reranking the results based on the variance.

In summary, we believe this paper has the following contributions:

- We study a systematic way to factor in uncertainty for accurate rate prediction. This expands the state-of-the-art of both database and recommender system research, by identifying a highly demanded application scenario for uncertain data management and introducing extensive developments on uncertainty for building recommender systems.
- We implement our proposed framework and evaluate with real-life movie recommendation datasets.

2 Backgrounds and Related Work

To establish the context of our discussion, we highlight the research efforts for recommender system and uncertain data management.

2.1 Recommender System

Recommendation has been widely adopted in many real-life data-intensive Web sites, such as Amazon.com or digg.com. In such systems, each user can either be represented as a vector of ratings she provided (*i.e.*, heuristics-based [7–10]) or a classifier model trained by other ratings (*i.e.*, model-based [11–13]). In this paper, we focus on predicting unknown ratings from the heuristics-based recommender system.

While details may vary, heuristics-based recommender systems adhere to the following template to predict rating $r_{u,o}$ of user u on object o:

1: Find top-k user set N with highest user similarity ω with u

2: Apply aggregation function $\mathcal{F}$ on the ratings of top-k users.

More specifically, we describe how a typical CF system implementation, *e.g.*, Open source toolkits CoFE[1], implements ω and $\mathcal{F}$ discussed above.

User Similarity $\boldsymbol{\omega}$ To identify the most similar k user set N, CoFE uses Pearson correlation coefficient between two users v and w, when $\overline{r_v}$ and σ_v represent the average and standard deviation of the ratings by user v respectively.

$$\omega_{v,w} = \frac{\sum_{i=1}^{m} (r_{v,i} - \overline{r_v}) \times (r_{w,i} - \overline{r_w})}{\sigma_v \times \sigma_w} \tag{1}$$

Rating Aggregation $\boldsymbol{\mathcal{F}}$ The k rating vectors identified are then aggregated to predict unknown ratings. For such aggregation, CoFE uses a weighted average, with weights proportional to user similarity to u.

$$r_{u,o} = \mathcal{F}(N, o) = \overline{r_u} + \frac{\sum_{i \in N} (r_{i,o} - \overline{r_i}) \times \omega_{u,j}}{\sum_{j=1}^{n} \omega_{u,j}} \tag{2}$$

Once the ratings are predicted, despite different levels of uncertainty involved with each prediction, existing systems treat all the ratings equally– Meanwhile, recent study [2] empirically suggests that distinguishing different levels of uncertainty in recommendation can boost accuracy.

2.2 Uncertain Data Management

In contrast, our proposed framework generate multiple *uncertain instances* for the prediction of $r_{u,o}$. More specifically, we adopt *possible worlds semantics*, as widely used in uncertain data management systems [3–6], representing dataset as a set of possible instances, each annotated with numerical membership *confidence*. In addition, to represent multiple instances representing the same real-world entity, *e.g.*, multiple predictions on $r_{u,o}$, generation rules are used, as we illustrate in Example 2 below.

TupleID	Rating	Confidence
t_1	5	0.8
t_2	4	0.2

Example 2 (Rating prediction). Continuing from Example 1, the rating on 'Quantum of Solace' can be predicted, by aggregating the ratings from the k users with similar tastes. For instance, when 80% of such users gives the perfect rating '5' and the rest gives '4', instead of aggregating this into a single predicted rating,

[1] http://eecs.oregonstate.edu/iis/CoFE/

e.g., '5', we can manage both predictions '5' and '4', annotated with the confidence scores, *e.g.*, 0.8 and 0.2 respectively. (Note, this illustrates one naive way to compute confidence for illustration. For our implementation, we normalize user ratings and weigh confidences proportionally based on user similarity, as similarly done in Eq. 2 for quantitative systems.)

Table 2.2 illustrates two tuples t_1 and t_2 representing these two uncertain instances. To indicate these two tuples are the multiple rating predictions for the same user-movie pair, we add a generation rules $t_1\ XOR\ t_2$, which represents the exclusiveness of the two instances.

Query processing for uncertain data [14–16] has been extensively studied lately for Boolean queries. Meanwhile, a more suitable query semantics for recommender system is ranking [3–6] based on both rating and confidence– Depending on how these two ranking criteria are combined, varying query semantics were studied, *e.g.*, U-Topk and U-kRanks in [3] and PT-k in [4], for which many efficient algorithms have been proposed [3–6].

In particular, in this paper, we consider U-kRanks semantics, which returns the tuple of the highest probability at each ranking position. Other semantics are less suitable for recommendation scenarios– PT-k returning all tuples with probability higher than some threshold in no order is not appropriate for recommendation where ordering is important. U-Topk requires all top-k results to belong together to the same mostly likely "possible world", which is a restricting requirement for recommendation.

3 Experiments

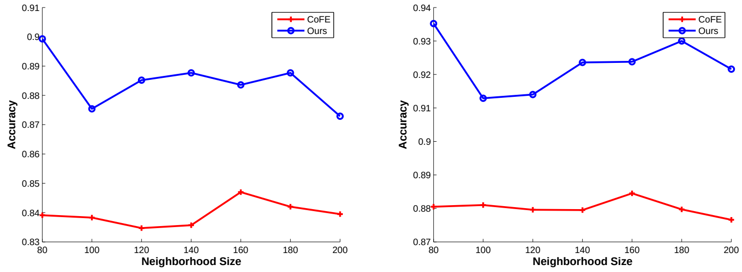
This section reports our evaluation results comparing historical and probabilistic CF systems.

For dataset, we used the same MovieLens rating dataset. Among the $100,000$ ratings by 943 users on 1682 items, We selected 568 users who have more than 50 ratings as testing users.

For evaluation metrics, we used the Normalized Discounted Cumulative Gain (NDCG) [17] metric, which is an popular metric for evaluating ranked results in information retrieval. The NDCG metric has advantages over the widely-adopted MAE metric for recommendation, by emphasizing the correctness of highly-ranked items more in metrics, as [18] recently adopted for evaluating his recommendation engine. More specifically, when Q is the set of users used for testing and $R(u,p)$ is the rating assigned by u to the item at the p-th position on the ranked list produced for user u, $NDCG(Q,k)$ is formally defined as follows:

$$NDCG(Q,k) = \frac{1}{|Q|} \sum_{u \in Q} Z_u \sum_{p=1}^{k} \frac{2^{R(u,p)} - 1}{log(1+p)} \tag{3}$$

where Z_u is a normalized factor calculated so that the NDCG value becomes one for the optimal ranking.



(a) Top-3 (b) Top-5

Fig. 1. NDCG vs. Neighborhood Size Results on MovieLens

Figure 1(a) and (b) reports the NDCG values for predicting Top-3 and Top-5 results, respectively when the size of neighborhood increases from 80 to 200. Regardless of the neighborhood size, the accuracy of our proposed framework is higher than that of a widely-adopted CF engine, by more than 5.23% and 5.07% for Top-3 and Top-5 predictions respectively.

4 Conclusion

This paper presents a new perspective on the classic collaborative filtering problem, aggregating community ratings to predict a rating for each item. In contrast, we aggregating community ratings into multiple predictions of varying levels of uncertainty and identify the top-k recommended items with both high rating and certainty. This system, to the best of our knowledge, is the first to systematically factor in the uncertainty for rating prediction in recommendation. Our evaluation results validate the accuracy our qualitative implementation, compared against a classic quantitative collaborative filtering implementation.

5 Acknowledgement

This work was supported by the Engineering Research Center of Excellence Program of Korea Ministry of Education, Science and Technology (MEST) / Korea Science and Engineering Foundation (KOSEF), grant number R11-2008-007-03003-0 and Korean Research Foundation Grant (MOEHRD, Basic Promotion Fund; KRF-2007- 331-D00377).

References

1. David Goldberg, David Nichols, Brian M. Oki, and Douglas Terry. Using collaborative filtering to weave an information tapestry. *Commun. ACM*, 35(12):61–70, 1992.

2. Y. Kwon. Improving top-n recommendation techniques using rating variance. In *RecSys*, 2008.
3. M. Soliman et. al. Top-k query processing in uncertain databases. In *ICDE*, 2007.
4. M. Hua et. al. Ranking queries on uncertain data: A probabilistic threshold approach. In *SIGMOD*, 2008.
5. K. Yi et. al. Efficient processing of top-k queries in uncertain databases with x-relations. In *TKDE*, 2008.
6. X. Lian et. al. Probabilistic ranked queries in uncertain databases. In *EDBT*, 2008.
7. GroupLens Research. [online] http://www.grouplens.org/.
8. John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI-98)*, 1998.
9. Jonathan L. Herlocker, Joseph A. Konstan, Al Borchers, and John Riedl. An algorithmic framework for performing collaborative filtering. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 230–237, New York, NY, USA, 1999. ACM.
10. Bharath Kumar Mohan, Benjamin J. Keller, and Naren Ramakrishnan. Scouts, promoters, and connectors: the roles of ratings in nearest neighbor collaborative filtering. In *EC '06: Proceedings of the 7th ACM conference on Electronic commerce*, pages 250–259, New York, NY, USA, 2006. ACM.
11. John Browning and David J. Miller. A maximum entropy approach for collaborative filtering. *J. VLSI Signal Process. Syst.*, 37(2-3):199–209, 2004.
12. Zhonghang Xia, Yulin Dong, and Guangming Xing. Support vector machines for collaborative filtering. In *ACM-SE 44: Proceedings of the 44th annual Southeast regional conference*, pages 169–174, New York, NY, USA, 2006. ACM.
13. Slobodan Vucetic and Zoran Obradovic. Collaborative filtering using a regression-based approach. *Knowl. Inf. Syst.*, 7(1):1–22, 2005.
14. R. Cheng et. al. Evaluating probabilistic queries over imprecise data. In *SIGMOD*, 2003.
15. A. Fuxman et. al. Conquer: Efficient management of inconsistent database. In *SIGMOD*, 2005.
16. J. Widom. Trio: A system for integrated management of data, accuracy, and lineage. In *CIDR*, 2005.
17. Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, 2002.
18. Nathan N. Liu and Qiang Yang. Eigenrank: a ranking-oriented approach to collaborative filtering. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 83–90, New York, NY, USA, 2008. ACM.