Influence-based Collaborative Active Learning

Neil Rubens *
Department of Computer Science
Tokyo Institute of Technology, Japan
neil@hrstc.org

Masashi Sugiyama*
Department of Computer Science
Tokyo Institute of Technology, Japan
sugi@cs.titech.ac.jp

ABSTRACT

In order to learn a user's preferences in collaborative recommender systems it is crucial to select the most informative items for a user to rate. For example, rating a popular item will provide little discriminative information about user's preferences since most users like popular items. Existing approaches select the most informative items based primarily on items' *uncertainty*, but tend to ignore an important metric of *coverage* – the number of items for which we are able to accurately estimate preferences. Selecting an item based only on uncertainty will reduce the uncertainty of the selected item, but will not necessarily reduce the uncertainty of other items – which is the ultimate goal. Therefore, in order to reduce the uncertainty over all items, we propose to select items that are not only uncertain but are also *influential*. Experimental results demonstrate the advantages of the proposed approach.

Categories and Subject Descriptors

G.3 [**Mathematics of Computing**]: Probability and Statistics—*Experimental design*; I.2.6 [**Computing Methodologies**]: Artificial Intelligence—*Learning*; H.1.1 [**Information Systems**]: Models and Principles—*Value of information*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces—*Collaborative Computing, Theory and Models*

General Terms

Algorithms, Design, Theory

*The authors acknowledge financial support from MEXT (Grant-in-Aid for Young Scientists 17700142, Grant-in-Aid for Scientific Research (B) 18300057, Research Fellowship 053155), the Global COE project (Computationism as Foundation of Sciences); and are thankful for the software provided by the open source community, in particular Sean Owen (http://taste.sourceforge.net).

RecSys'07, October 19–20, 2007, Minneapolis, Minnesota, USA.

1. INTRODUCTION

When the number of items is very large, it may not be possible for a user to examine all of the choices in order to find the most suitable one. Recommender systems allow us to cope with this problem by suggesting items (e.g. movies, books, or web sites) that a user is likely to be interested in. Items are presented by a recommender system mainly for two reasons: *exploitative* – presenting an item that a user will probably like and buy; *explorative* – presenting an item in order to better learn user's preferences. In this paper we concentrate on the latter (explorative) issue of learning user's preferences.

Collaborative filtering is a commonly used approach that allows us to approximate a user's preferences [11, 5, 1]. It is based on the assumption that if a group of users shares common interests, then it is probable that an item that one user likes will also be favored by the rest of the group's users. Therefore if we can find users that are similar to the target user we will be able to predict the user's preferences based on the preferences of similar users. A user must express preferences for at least a few items before it is possible to find other similar users. However, most users are willing to express their preferences only for a small number of items. Therefore it is crucial to select the most informative items for learning the preferences of a new [4, 9, 13, 2] or existing user [12, 14, 8].

Existing approaches obtain information about user's preferences often by asking a user to rate the most uncertain items, where uncertainty is usually estimated by entropy [4, 9, 13] or variance [2, 4]. However, existing methods tend to ignore an important metric of *coverage* (the number of items for which we are able to accurately estimate preferences) [3]. Ignoring coverage and selecting items based only on uncertainty could result in reducing uncertainty in a local manner. For example, picking an outlier item (e.g. a movie 'Eating habits of ducks') will reduce the uncertainty of only a few other items. In order to reduce uncertainty in a global manner, we propose to reduce uncertainty with respect to coverage. To achieve this goal, we select items to be rated that are not only uncertain, but are also influential (measured by the changes in the approximated ratings of other unrated items).

2. PROBLEM DEFINITION

In recommender systems user's preferences are often estimated based on the user's ratings and the ratings of other users. It is possible to ask a user to rate items. However, the user is often not willing to rate items unless it results in the significant improvement in the accuracy of predictions. Therefore it is crucial to select the most informative items in order to be able to accurately determine the user's preferences. In the following paragraphs we will formulate this problem in a more formal way.

In this paper we consider memory-based collaborative filtering algorithms [11, 7, 1], i.e. the user's preferences are approximated based on the preferences expressed by other users. Let X be the set of all items and $X_c^{(r)}(\subset X)$ be the set of items that have been rated by the target user c, where r stands for rated. Let us denote the rating of the item $x(\in X_c^{(r)})$ by the user c as $y_{c,x}$. The set of items that have not been rated by the user c is expressed as $X_c^{(u)}$, where u stands for for unrated. Note that $X_c^{(r)} \bigcap X_c^{(u)} = \emptyset$ and $X_c^{(r)} \bigcup X_c^{(u)} = X$. The approximated value of the unknown rating $\hat{y}_{c,x}$ is commonly [1] expressed as:

$$\hat{y}_{c,x} = k \sum_{c' \in \hat{C}} sim(c, c') \times y_{c',x}, \tag{1}$$

where $\hat{C}$ denotes the set of users that are the most similar to the given user c and who have rated item x; k is a normalizing constant $k = (\sum_{c' \in \hat{C}} |sim(c, c')|)^{-1}$; and sim is a similarity function usually defined by Pearson correlation as:

$$sim(c, c') = \frac{\sum_{x \in X_{cc'}^{(r)}} (y_{c,x} - \bar{y}_c)(y_{c',x} - \bar{y}_{c'})}{\sqrt{\sum_{x \in X_{cc'}^{(r)}} (y_{c,x} - \bar{y}_c)^2 \sum_{x \in X_{cc'}^{(r)}} (y_{c',x} - \bar{y}_{c'})^2}}, \tag{2}$$

where $X_{cc'}^{(r)}$ is the set of items that have been rated by both users c and c', i.e. $X_{cc'}^{(r)} = X_c^{(r)} \bigcap X_{c'}^{(r)}$; and the mean rating of the user c is $\bar{y}_c = \frac{1}{|X_c^{(r)}|} \sum_{x \in X_c^{(r)}} y_{c,x}$.

Eqs.(1) and (2) imply that the choice of $X_c^{(r)}$ (used for finding similar users) affects the accuracy of approximated ratings. In practice, a user is willing to rank only a small number of items. Therefore it is important to optimize the items to be ranked so that for all of the items $x \in X_c^{(u)}$, $\hat{y}_{c,x}$ well approximates the true rating $y_{c,x}$. This is the problem we address in this paper.

3. PROPOSED APPROACH

In existing collaborative active learning methods [4, 9] an informative item is selected according to the uncertainty reduction in a *local* manner. That is an item with the highest uncertainty is selected and its rating is obtained. Obtaining the rating of the most uncertain item reduces the uncertainty of the rated item. However, it will not necessarily reduce the uncertainty of other items – which is the ultimate goal. Thus, items to be rated should be selected with respect to the *global* reduction in uncertainty. In order to achieve this we introduce the *influence* criterion which measures the effect that rating an item has on the approximated values of other unrated items.

To illustrate this, let us consider a simplified example shown in Figure 1. This figure shows items that are projected onto two dimensional space. A rated item is represented as a black circle, while an unrated item is represented as a white circle. The distance between items represents their relation (the smaller is the distance the stronger is the relation). The items that have strong relation between each other could be considered to belong to the same cluster. When an item becomes rated it reduces the uncertainty of other related items. Suppose the uncertainty value is equal for the items in clusters 2 and 3. Since the number of related items is larger in cluster 2, each item in cluster 2 *influences* more items than each item in cluster 3. Therefore, selecting an item to rate based not only on the uncertainty criterion but also on the influence criterion reduces uncertainty in a global manner.

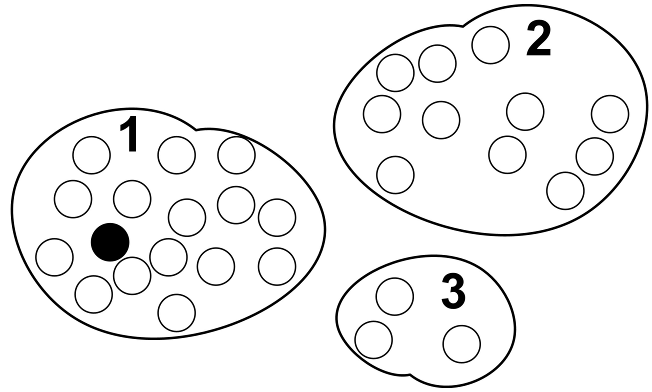

Figure 1: Illustration of the importance of the influence criterion described in Sections 3, 4.

Following the above idea, we propose to select an item to rate based on both the uncertainty and the influence criterions. We combine the uncertainty criterion $U(x)$ and the influence criterion $I(x)$ as:

$$argmax_x U(x) I(x). \tag{3}$$

Item's rating uncertainty $U(x)$ could be estimated by utilizing the existing methods through the use of variance [4, 2]. Variance of the rating value y of item x could be calculated as:

$$var(y|x) = \frac{1}{|Y_x|} \sum_{y \in Y_x} (y - \overline{y}_x)^2, \tag{4}$$

where Y_x is the set of ratings of item x given by the users that have rated item x, $|Y_x|$ is the number of elements in Y_x, and $\overline{y}_x$ is the mean value of elements in Y_x.

The influence information $I(x)$ is not available in advance. In the following section we propose an influence estimation approach.

4. INFLUENCE ESTIMATION

When the rating of an item is obtained it affects approximated ratings of other unrated items (Eqs.(1) and (2)). We propose to estimate the influence of an item by changing its rating and observing the changes in the approximated ratings of other unrated items as illustrated in Figure 2. In the following paragraphs we will formulate this approach in a more formal way.

We want to estimate the influence that rating a candidate item $\widetilde{x}(\in X_c^{(u)})$ will have on the approximated ratings of other unrated items in $X_c^{(u)} \backslash \widetilde{x}$. Let us denote the training set of user c as a set of rated items and their corresponding

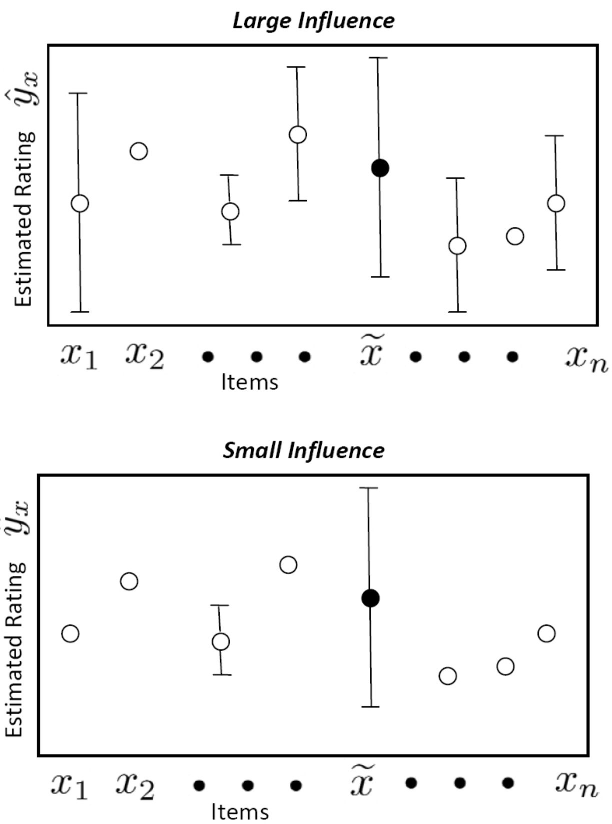

Figure 2: Influence estimation as described in Section 4.

ratings i.e. $T_c = \bigcup_{x \in X_c^{(r)}} (x,\ y_{c,x})$. We estimate the effect that adding a varying hypothetical rating $\hat{y}_{c,\tilde{x}} + \delta$ of item $\tilde{x}$ to the training set T_c has on the approximated output values of other unrated items in $X_c^{(u)} \backslash \tilde{x}$. Let $\widetilde{T}_{c,\delta}$ be the augmented training set obtained by adding $(\tilde{x},\ \hat{y}_{c,\tilde{x}} + \delta)$ to T_c:

$$\widetilde{T}_{c,\delta} = (\tilde{x}, \hat{y}_{c,\tilde{x}} + \delta) \bigcup T_c\ . \tag{5}$$

Then the influence criterion is defined as:

$$I(\tilde{x}) = \sum_{x \in X_c^{(u)} \backslash \tilde{x}} \sum_{\delta} \left| \hat{y}_{c,x}^{(T_c)} - \hat{y}_{c,x}^{(\widetilde{T}_{c,\delta})} \right|, \tag{6}$$

where $\hat{y}_{c,x}^{(T_c)}$ is the approximated value of $y_{c,x}$ given the training set T_c; and in the case of $\hat{y}_{c,x}^{(\widetilde{T}_{c,\delta})}$ given training set $\widetilde{T}_{c,\delta}$.

In practice, it is computationally expensive to use all possible values of δ. In preliminary experiments we used $\delta \in \{-1,\ 1\}$, $\delta \in \{-1\}$ and obtained similar results.[1] For this reason, we decided to use $\delta \in \{-1\}$ in numerical experiments (Section 5).

5. NUMERICAL EXPERIMENTS

We evaluated proposed and existing methods (random, popularity [4], variance [4], entropy [4] and a hybrid popularity-

[1]When $\hat{y}_{c,\tilde{x}} + \delta$ is out of range, we project the value into the range.

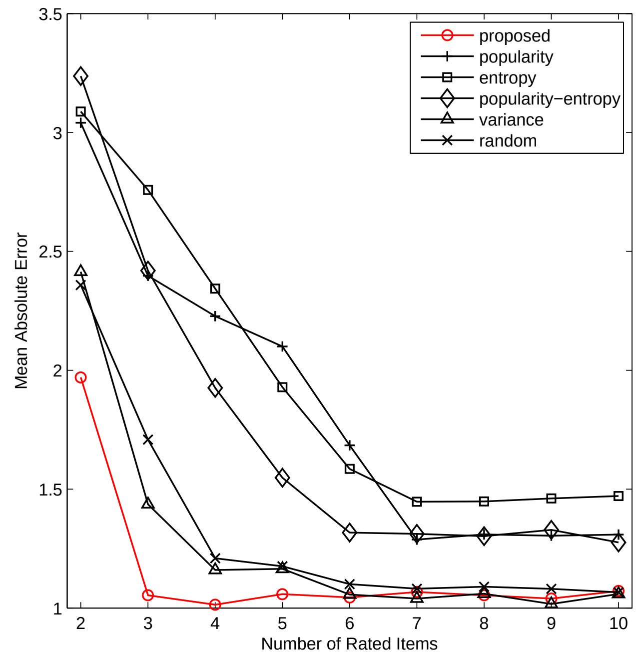

Figure 3: Mean absolute error (MAE) achieved after obtaining a user's preferences for the given number of items (in the current implementation MAE is undefined when only one item has been rated). Items to be rated are selected by corresponding methods. Details of experimental setup are given in Section 5.

entropy method [9]) in an offline experiment utilizing the MovieLens dataset[2]. We randomly selected 100 users that have each rated at least 100 items. For each user we withheld all of the user's items and by applying corresponding methods sequentially selected 10 items for which the preferences would be expressed. At each step we obtained the user's preference function by applying memory-based Pearson correlation to selected rated items (Eqs.(1) and (2)), and calculated mean absolute error by using the rest of the user's items (Figure 3).

At each step an item to be rated x is selected by the corresponding methods as follows. In the random approach, an item is selected randomly following the uniform distribution. In the popularity approach, the most popular item is selected i.e. $argmax_x Pop(x)$, where popularity is measured by the number of expressed ratings for a given movie by all of the users i.e. $Pop(x) = |Y_x|$ [4]. In the variance approach, the item with the largest rating variance (calculated by Eq.(4)) is selected [4]. In the entropy approach, the item with the highest entropy is selected i.e. $argmax_x H(y_x)$, where the entropy of an item is estimated using the relative frequency of all possible ratings r i.e. $H(y_x) = -\sum_r P(y_x = r|Y_x) \log P(y_x = r|Y_x)$, where $P(y_x = r|Y_x)$ is the probability that item x is rated r, given the set Y_x of all expressed ratings of item x. Finally, in the hybrid popularity-entropy approach, an item that is popular and has a high entropy is selected as $argmax_x H(y_x) \log Pop(x)$ [9].

Experimental results (Figure 3) demonstrate that the speed of convergence of the proposed method is faster than that of

[2]available at http://movielens.umn.edu/

existing approaches, especially in the early stages. However, as the number of items increases, the accuracy of all of the methods flattens out. It is crucial to be able to approximate user's preferences given only a few rated items. Therefore the proposed approach appears to be a promising candidate.

6. CONCLUSION

Collaborative filtering is based on the assumption that users share common interests, i.e. users' preferences are related. However, existing methods [4, 9] do not take advantage of the relations between items and as a result reduce uncertainty in a *local* manner (Section 3). On the other hand, by utilizing the strength of the relations between items (estimated by the proposed *influence* criterion) the proposed method reduces uncertainty in a *global* manner. This allows to improve not only the *uncertainty* but also the *coverage*.

Increased coverage could also allow a recommender system to suggest an unfamiliar item to a user that may prompt a user to be interested in the novel category [3, 6]. It has been shown in [10], that also from the user's perspective, influence is an important factor that affects recommendations from a recommender system (i.e. ratings of an influential user significantly affect the predicted recommendations of other users).

The proposed approach could be applied to *any* collaborative recommendation methods that utilizes user's rated items for estimating the rating of unrated item. In this paper, we concentrate on the explorative value of the information that rating an item would provide. Therefore we assume that it is possible to suggest an unfamiliar item for the user to evaluate (e.g. asking a user to watch and rate an unfamiliar movie). However, when it is necessary for a user to provide immediate feedback about an item, the probability that the user has already formed a preference could be incorporated by applying existing methods e.g. [13].

The computational complexity of influence estimation is $\mathcal{O}(n^2)$ (where n is the number of unrated items). In future works we are planning to further reduce the computational complexity of influence estimation by e.g. incrementally updating itemwise influence instead of recalculating it at each step. In addition, we are planning to investigate the theoretical properties of the proposed method.

7. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749, 2005.

[2] D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996.

[3] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, 2004.

[4] A. Kohrs and B. Merialdo. Improving collaborative filtering for new users by smart object selection. In *Proceedings of International Conference on Media Features (ICMF)*, 2001.

[5] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl. Grouplens: applying collaborative filtering to usenet news. *Commun. ACM*, 40(3):77–87, 1997.

[6] S. M. McNee, J. Riedl, and J. A. Konstan. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pages 1097–1101, New York, NY, USA, 2006. ACM Press.

[7] A. Nakamura and N. Abe. Collaborative filtering using weighted majority prediction algorithms. In *Proc. 15th Intl Conf. Machine Learning*, 1998.

[8] B. F. Paolo Viappiani and P. Pu. Preference-based search using example-critiquing with suggestions. *Journal of Artificial Intelligence Research (JAIR)*, 27:pages 465–503, 2006.

[9] A. M. Rashid, I. Albert, D. Cosley, S. K. Lam, S. M. McNee, J. A. Konstan, and J. Riedl. Getting to know you: learning new user preferences in recommender systems. In *IUI '02: Proceedings of the 7th international conference on Intelligent user interfaces*, pages 127–134, New York, NY, USA, 2002. ACM Press.

[10] A. M. Rashid, G. Karypis, and J. Riedl. Influence in ratings-based recommender systems: An algorithm-independent approach. In *SIAM International Conference on Data Mining*, 2005.

[11] P. Resnick, N. Iacovou, M. Suchak, P. Bergstorm, and J. Riedl. GroupLens: An Open Architecture for Collaborative Filtering of Netnews. In *Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work*, pages 175–186, Chapel Hill, North Carolina, 1994. ACM.

[12] B. Smyth and E. McKenna. Modelling the competence of case-bases. *Lecture Notes in Computer Science*, 1488:208–220, 1998.

[13] K. Yu, A. Schwaighofer, V. Tresp, X. Xu, and H.-P. Kriegel. Probabilistic memory-based collaborative filtering. *IEEE Transactions on Knowledge and Data Engineering*, 16(1):56–69, 2004.

[14] J. Zhu and Q. Yang. Remembering to add: Competence-preserving case-addition policies for case base maintenance. In *IJCAI*, pages 234–241, 1999.