Active Learning for Aspect Model in Recommender Systems

Rasoul Karimi, Christoph Freudenthaler, Alexandros Nanopoulos, Lars Schmidt-Thieme
Information Systems and Machine Learning Lab (ISMLL)
Samelsonplatz 1, University of Hildesheim, D-31141 Hildesheim, Germany
Email:[karimi,freudenthaler,nanopoulos,schmidt-thieme]@uni-hildesheim.ismll.de

***Abstract*—Recommender systems help Web users to address information overload. Their performance, however, depends on the amount of information that users provide about their preferences. Users are not willing to provide information for a large amount of items, thus the quality of recommendations is affected specially for new users. Active learning has been proposed in the past, to acquire preference information from users. Based on an underlying prediction model, these approaches determine the most informative item for querying the new user to provide a rating. In this paper, we propose a new active learning method which is developed specially based on aspect model features. There is a difference between classic active learning and active learning for recommender system. In the recommender system context, each item has already been rated by training users while in classic active learning there is not training user. We take into account this difference and develop a new method which competes with a complicated bayesian approach in accuracy while results in drastically reduced (one order of magnitude) user waiting times, i.e., the time that the users wait before being asked a new query.**

I. Introduction

The central goal of machine learning is to develop systems that can learn from experience or data and improve their performance at some task. In many natural learning tasks, this experience or data is gained interactively, by taking actions, making queries, or doing experiments. Most machine learning research, however, treats the learner as a passive recipient of the data to be processed. This passive approach ignores the learners ability to interact with the environment and gather data. Active learning is the study of how to use this ability effectively. Active learning algorithms have been developed for classification, regression and function optimization and is found to improve the predictive accuracy of several algorithms compared to passive learning.

Recommender systems guide users in a personalized way to interesting or useful objects in a large space of possible options [1]. For example, which movie should I see? or what book should I read? We have too many choices and too little time to explore them all and the exploding availability of information makes this problem even tougher.

There are several techniques for recommendation and collaborative filtering is one them [2], [3]. Given a domain of items, users give ratings to these items. The recommender system can then compare the users ratings to those of other users, find the most similar users based on some criterion of similarity, and recommend items that similar users have already liked [5].

Evidently, the performance of recommender systems depends on the number of ratings that the users provide. However, a well identified problem, is that users are not willing to provide ratings for a large amount of items [18], [19]. This problem is amplified even more in the case we have lack of ratings due to a new user or a new item (cold-start problem). Therefore, the queries presented to the new users, have to be selected carefully, because users are not willing to answer a lot of queries. To address this situation active learning methods have been proposed to acquire those ratings from new user, that will help most in determining her interests [19], [18].

When performing active learning in recommender systems, besides the accuracy of recommendations, another important factor is the minimization of the user waiting time, that is, the time that the new user has to wait before being asked a new query. Existing active learning methods that are based on aspect model [18] [19], present the advantage of accurate recommendation (compare to the more simplistic memory-based approaches [21] [20]), but they usually result in prohibitively large waiting time. For instance, as will be demonstrated by our experimental results, Bayesian active learning approaches despite their good accuracy, may lead to user waiting time in the order of several tens of seconds. Such large waiting times are hardly suitable for the interactivity required when performing active learning for recommender systems.

A. Motivation

The motivation in our work is twofold;

1) The selection of queries for the new user (active user) in existing methods is performed mostly based on criteria which are inspired from the literature of classic active learning [14]. However, aspect model has special features that needs its own active learning algorithm and the original active learning criteria are not suitable for this prediction model.
2) When performing active learning in recommender systems, besides the accuracy of recommendations, another important factor is the minimization of the user waiting time, that is, the time that the active user has to wait before being asked a new query. Existing active learning methods that are based entropy [18] [19] present the advantage of accurate recommendation (compare to the

more simplistic memory-based approaches [21] [20]), but they usually result in prohibitively large waiting time. Such large waiting times are hardly suitable for the interactivity required when performing active learning for recommender systems. Our aim is to exploit the accuracy of model-base recommender system and develop a fast active learning method. To achieve this aim, we avoid to apply directly a classic active learning criterion on recommender system. Instead, the characteristics of aspect model is taken into account and a new criterion is developed.

B. Contributions and Outline

- We propose a new selection criteria for active learning which is based on aspect model features. In contrast to the previous works, this method is developed specially for aspect model in recommender system. The proposed method selects items for querying which are most effective to improve the user latent parameters in aspect model.
- The proposed approach results in drastically reduced user waiting times (one order of magnitude). The reason is that this method takes into account the difference between classic active learning and active learning for recommender system. In the recommender system context, each item has already been rated by training users while in classic active learning there is not training user. Considering this difference, we can find new algorithms which rely on this additional information instead of complicated computations.

The rest of this paper is organized as follows: in section 2, the related works are reviewed. Active learning for aspect is explained in section 3. Then the proposed method will be described in section 4, followed by experimental result in section 5. Finally the conclusion is stated in section 6.

II. Related Works

In this section, we will first briefly discuss the previous work on active learning, followed by the previous work on active learning for recommender system.

A. Related Works on Active Learning

Cohn et al. [22] describe how optimal data selection techniques can be applied to statistically-based learning algorithms like a mixture of Gaussians and locally weighted regression. The algorithm selects instances that if labeled and added to the training set, minimizes the expected error on future test data. The authors show that the statistical models perform more efficiently and accurately than the feedforward neural networks.

Similar querying functions have been proposed by Tong and Koller [12], Campbell et al. [24] and Schohn [11], called Simple which uses SVMs as the induction component. Here, the querying function is based on the classifier. The algorithm tries to pick instances which are the most informative to the SVM - the support vectors of the dividing hyperplane. This can be thought of as uncertainty sampling where the algorithm selects those instances about which it is most uncertain. In the case of SVMs, the classifier is most uncertain about the examples that are lying close to the margin of the dividing hyperplane. Variations of the Simple algorithm - MaxMin and Ratio methods have been proposed by Tong and Koller [12] , which also use SVM as the learner

Roy and McCallum [8] describe a method to directly maximize the expected error rate reduction, by estimating the future error rate by a loss function. The loss functions help the learner to select those instances that maximize the confidence of the learner about the unlabeled data. Rather than estimating the expected error on the full distribution, this algorithm estimates it over a sample in the pool. The authors base their class probability estimates and classification on naive Bayes, however SVMs or other models with complex parameter space are also recommended.

Osugi et al. [23] propose an active learning algorithm that balances the exploration and exploitation while selecting a new instance for labeling by the expert at each step. The algorithm randomly chooses between exploration and exploitation at each round and receives feedback on the effectiveness of the exploration step, based on the performance of the classifier trained on the explored instance.

Jain and Kapoor [25] studied active learning for large multi-class problems. Most of the active learning algorithms are inherently for binary classification and do not scale up to the large number of classes. In this paper, they introduce a probabilistic variant of the K-Nearest Neighbor method for classification that can be seamlessly used for active learning in multi-class scenarios.

B. Related Works on Active Learning for Recommender System

Active Learning in the context of the new user problem was introduced by Kohrs and Merialdo [20]. They suggested a method based on nearest-neighbor collaborative filtering which uses entropy and variance as the loss function to identify best item for query. Mamunur et al. [5] expanded this work by considering the popularity of items and also personalizing the item selection for each individual user. Boutilier et al. [21] applies the metric of expected value of utility to find the most informative item for query, which is to find the item that leads to the most significant change in the highest expected ratings. Jin and Si [18] developed a new active learning algorithm based on aspect model which is similar to the active learning approach towards parameter estimation in Bayesian networks [14]. In this method, entropy is used to compute the uncertainty. They propose a complex bayesian approach which its application for a real recommender system is intractable. This is because in each query, for each candidate item and for each rating, the user parameters should be retrained.

Harpale and Yang [19] extended [18] by relaxing this assumption that a user can provide rating for any queried item. It personalize active learning for the user, and query for only those items which the user can provide rating for. The details of these two works will be given in the next section.

In this paper we assume that the new user will always be able to rate the items presented by the active learning. Since the focus of this paper is on the behavior of aspect model, we will leave this issue as the future work.

III. Active learning for aspect model

The primary works to apply active learning in recommender system were based on nearest-neighbor [20], [5]. Then, in order to improve the performance of active learning, the aspect model which is a stronger prediction model, was engaged [18], [19]. In this section, we provide a short introduction to aspect model and review the existing active learning algorithms based on that.

A. Aspect Model

For the convenience of discussion, we will first introduce the annotation. Let items denoted by $M = \{m_1, m_2, ..., m_M\}$, users denoted by $U = \{u_1, u_2, ..., u_N\}$, and ratings denoted by r. A tuple (u, m, r) means that rating r is assigned to item m by user u.

Aspect model is a probabilistic latent space model, which models individual preferences as a convex combination of preference factors [16], [17]. The latent class variable $z \in Z = \{z_1, z_2, ..., z_k\}$ is associated with each pair of a user and an item. The aspect model assumes that users and items are independent from each other given the latent class variable. Thus, the probability for each observation tuple (m, u, r) is calculated as follows:

$$p(r|m, u) = \sum_{z \in Z} p(r|z, m)p(z|u) \tag{1}$$

where $p(z|u)$ stands for the likelihood for user u to be in class z and $p(r|z, m)$ stands for the likelihood of assigning item m with rating r by users in class z. In order to achieve better performance, the ratings of each user are normalized to be a normal distribution with zero mean and variance as 1 [17]. The parameter $p(r|z, m)$ is approximated as a Gaussian distribution $N(m_z, s_z)$ and $p(z|u)$ as a multinomial distribution.

B. Bayesian Active Learning

When a new user starts to use a recommender system, although the system already has an accurate global-model, but the new user model is weak, because the system has very few ratings available from her. Therefore, active learning technique is incorporated to acquire additional ratings.

There are several criteria in active learning for item selection and entropy minimization is one of them. In the context of aspect model, this leads to the following equation for a user u, where θ_{u_z} denotes the current user parameters $(P(z|u))$, and $\theta_{u_z|m,r}$ denotes user parameters after retraining the user-model based on a newly obtained rating r for movie m from the user i.e. $P(z|u, m, r)$:

$$m_u^* = argmin_m \in M - \left\langle \sum_{z \in Z} \theta_{u_z|m,r} \log \theta_{u_z|m,r} \right\rangle_{P(r|u,m)} \tag{2}$$

Equation 2 denotes the expected entropy of the user parameters after being trained over additional information of rating movie m with rating r. As the exact rating r is not known for the unrated movies, the expected value of rating based on the current model $P(r|u, m)$ is used, as shown in the equation. The drawback of this approach is that it tries to strictly categorize users into one class while in the reality users may belong to several classes. For example, users may like both comic and action movies. In order to eliminate this problem, Jin and Si [18] proposed a Bayesian selection approach. This method identifies item m, such that the updated model $\theta_{u_z|m,r}$ will be accelerated towards the true user model true θ_u^{true}.

$$m_u^* = argmax_m \in M \left\langle \sum_{z \in Z} \theta_{u_z}^{true} \log \frac{\theta_{u_z|m,r}}{\theta_{u_z}^{true}} \right\rangle_{P(r|u,m)} \tag{3}$$

As the true user model is unknown beforehand, it is estimated as the expectation over the posterior distribution of the user model. Jin and Si [18] provides a computationally efficient way of approximating the posterior distribution, but still it is intractable for real recommender systems.

The items selected by the Bayesian selection approach will accelerate the user-model to the true user-model, only if the user knows the rating of the queried item. However, in reality, users cannot provide ratings for all items. For example, users do not watch all movies, so they cannot provide ratings.

Thus, in addition to selecting the items which will accelerate the model towards the true user model θ_u^{true}, the active learning algorithm should also try to select items which have a very high probability of getting a rating from a user. To address this issue, Harpale and Yang [19] extended Jin and Si [18] and introduced a new term $P(m|u)$ into Equation 3, that is the probability of getting a rating, on the item m from the user u:

$$m_u^* = \tag{4}$$

$$argmax_m \in M \left(\left\langle \sum_{z \in Z} \theta_{u_z}^{true} \log \frac{\theta_{u_z|m,r}}{\theta_{u_z}^{true}} \right\rangle_{P(r|u,m)} \right) P(m|u)$$

The multiplication of the Bayesian selection criterion from Equation 3 and the personalization term $P(m|u)$ acts as a soft-AND which is maximized when both the multiplicands are maximized. $P(m|u)$ is approximated as follows:

$$P(m|u) = \sum_{z \in Z} P(m|z)P(z|u) \tag{5}$$

$$P(m|z) = \frac{\sum_{u\in U} P(z|u)I(u,m)}{\sum_{u\in U}\sum_{m'\in M} P(z|u)I(u,m')} \quad (6)$$

I(u,m) = 1, if user has rated item m and 0 otherwise

IV. Proposed active learning

There are several methods in the classic active learning literature which could be incorporated for active learning in recommender systems. However, recommender system has its own characteristics that makes us think carefully about how to apply active learning in this context. There is a big difference between classic active learning and active learning for recommender system: In the classic active learning there is pool of unlabeled examples and the goal is to find best examples for query. In the recommender system context, items are examples but these items have already been rated by training users. This is something which is not available in classic active learning. Ignoring this fact and applying directly an active learning algorithm for recommender system provides a weak solution which is not applicable for real recommender systems. This is the big drawback of bayesian approach explained in section III-B. This method customizes Bayesian active learning [18] for aspect model in recommender systems which is too complicated and makes it intractable for a real recommender system.

In order to develop an active learning algorithm for aspect model, one should consider the rating prediction method in aspect model and then optimize it. In aspect model, the unknown ratings are estimated according to equ. 1. This equation consists of user latent parameters and item latent parameters. After getting a new rating, these parameters are retrained to improve them toward true latent parameters. Better estimation of user and item parameters provides a more accurate prediction for unknown ratings. According to [17] item parameters have a normal distribution with mean and variance that are computed as following :

$$\mu_{m,z} = \frac{\sum_{\langle u,m',r\rangle:m'=m} rP(z|u,r,m)}{\sum_{\langle u,m',r\rangle:m'=m} P(z|u,r,m)} \quad (7)$$

$$\sigma^2_{m,z} = \frac{\sum_{\langle u,m',r\rangle:m'=m} (r-\mu_{m,z})^2 P(z|u,r,m)}{\sum_{\langle u,m',r\rangle:m'=m} P(z|u,r,m)} \quad (8)$$

For each item, the parameters are trained with all ratings relevant to it, i.e all training users who have rated that item. The average In the datasets that will be used in the experiments, each item has enough ratings among training users that Therefore, items have trained enough in the training phase and updating corresponding item parameters after getting a new rating from the test user does not effect them too much. But the new rating is very relevant for the new user parameters. It is because these parameters have already been trained with a few ratings and updating them after each new rating causes a lot of improvement. These parameters are trained as follow:

$$P(z|u,r,m;\hat{\theta}) = \frac{\hat{p}(r|m,z)\hat{P}(z|u)}{\sum_{z'} \hat{p}(r|m,z)\hat{P}(z'|u)} \quad (9)$$

and then are normalized :

$$P(z|u) = \frac{\sum_{\langle u',m,r\rangle:u'=u} P(z|u,r,m;\hat{\theta})}{\sum_{z'}\sum_{\langle u',m,r\rangle:u'=u} P(\prime z|u,r,m;\hat{\theta})} \quad (10)$$

Therefore, the best way to improve the aspect model is to retrain it with the item that has highest effect to improve the user parameters. This is a new active learning criteria for item selection which is proposed in this paper.

Equations 9 indicates that latent parameters of user u are trained with all tuples (u,m,r) relevant to u which are already computed in equ. 11. In the other hand, equ. 9 relies on the mean and variance of the item parameters. Therefore, we can conclude that user parameters depend on the item parameters:

$$P(z|u) \propto \left(\mu_{m,z}, \sigma^2_{m,z}\right)_{\forall m\in M} \quad (11)$$

It means that the improvement which is gained after retraining the user parameters, depends on the item which is used for retraining. More certainty about mean and variance of the item leads to greater improvement in user parameters.

From statistic we know that the certainty about mean and variance depends on the number of samples, i.e the validity of mean and variance increases as the number of samples grows up. Therefore, the most popular item has the most certainty mean and variance and retraining the user parameters with it will improve the user parameters as much as possible.

A. Online Updating

The prediction model,aspect model, should be updated to learn user latent factors for new user. As there are already a lot of users in recommender system, it does not make sense to retrain the model from scratch because it needs a long time. Therefore, we have to do online updating which means to retrain only latent features for new user and not the rest. Please note that online updating takes place after each question sequentially.

For online updating, we use the method introduced in [26]. In this method after getting a new rating from user, the user latent parameters are restarted to random and then learned again using all ratings.

V. Experimental Result

A. Data Sets

We use MovieLense[1] and MovieRating [2] datasets in our experiments. MovieLense contains 943 users and 1682 items. In MovieRating, there are 500 users and 1000 items. The datasets were randomly split into training and test sets .In MovieLens there are 343 training users and MovieRating includes 200 training users (the same number used in [19]). Test data contains only users who have rated more than 33 items which at least 20 of them, that will be used for test data, already appeared in training data. It is because to test aspect model, we need an item that is used in training phase. Each test user is considered as a new user and her preliminary model is built using three random initial ratings. Remaining items are split into pool set and the test set. Active learning algorithm selects items for query from the pool set. Number of latent factors is 5 and 10 in MovieRating and MovieLense respectively according to [18] and [19].

B. Accuracy

The mean absolute error (MAE) is used to evaluate the performance of active learning algorithms, which is defined as follows

$$MAE = \frac{1}{N_{Test}} \sum_{m \in M_{Test}} \left| r^{true} - r^{predicted} \right| \qquad (12)$$

in which N_{Test} is the number of test items and M_{Test} is the test item set.

For each test user, the latent parameters are trained by 3 initial random ratings using online updating technique. Online updating is repeated after getting new ratings sequentially.

C. Result

The accuracy of the proposed active learning and bayesian method for Movielense and MovieRating datasets are presented in Fig. 1 and 2 respectively.

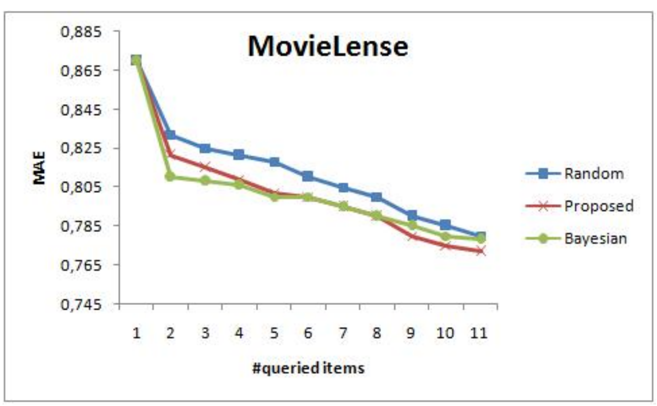

Fig. 1. MAE results of the proposed method,bayesian active learning and random algorithms over MovieLense dataset.

[1] www.grouplens.org/system/files/ml-data.zip

[2] www.cs.usyd.edu.au/irena/movie-data.zip

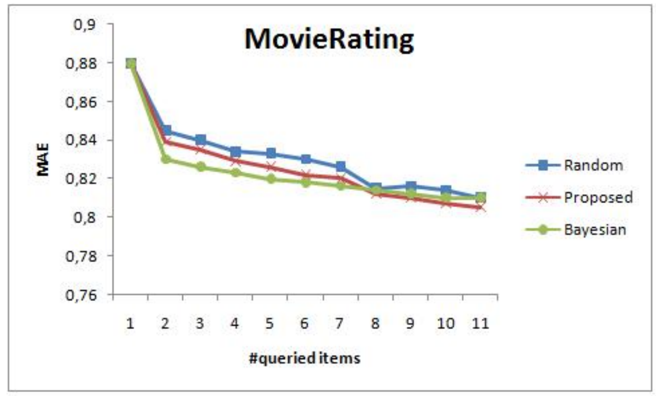

Fig. 2. MAE results of the proposed method, bayesian active learning, and random algorithms over MovieRating dataset.

These charts could be divided into two parts: in the first part bayeian method outperforms and in the second part the proposed method overtakes the bayesian method. These two parts have different meaning in the literature of e-commers. The first part deals with customer attraction which aims to find new customers. As the results suggest, bayesian method is suitable for this goal. In the other hand, the second part addresses customer satisfaction which tries to keep the current customers. The proposed method is appropriate for this part. Both bayesian and the proposed method methods rely on the training users. As the number of training data in Movielense is larger, the difference between these methods and random is more clear in this data set. Fig. 1.

The user waiting time of these methods, according to our experiments, are shown in Fig. 3.

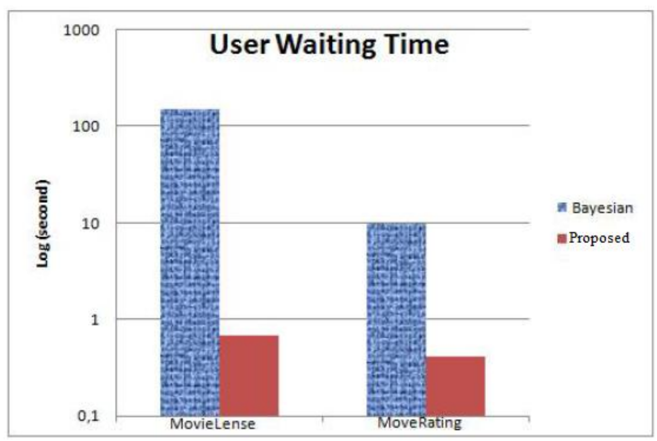

Fig. 3. Average user waiting time in MovieLense and MovieRating datasets. The axis y is log(seconds)

The computation time of the proposed method is much less than Bayesian method. It is because bayesian method needs a lot of computation for posterior estimation while the proposed method incorporates the characteristics of aspect model. This consideration provides a fast method which its accuracy is nevertheless same as the bayesian method.

By considering both time and accuracy, the proposed method is more suitable for recommender system specially as the size of recommender system grows up.

VI. Conclusion

Active learning is a suitable method to deal with some challenges in recommender system such as new user or new item. However, it should be revised according to the special features of recommender system and requirements of the problem. For example, in new user problem, in addition to the accuracy *user waiting time* is also important. If user is asked to wait for several minutes for next query, recommender system will not be interactive and user will leave the conversation. In this paper, we developed a new active learning which relies on aspect model as the prediction model of recommender system. As this method takes into account the special characteristics of aspect model, it is able to compete with bayesian active learning in accuracy while its running time is in the *order of magnitude* less than bayesian approach.

For the future work, we plan to personalize our method for each new user. This is important when the assumption that new user always knows the rating of the requested item is relaxed. Thus, in addition to the time and accuracy, we should take care of the queries that will not get any answer from user and so do not improve the prediction model.

Acknowledgment

The authors gratefully acknowledge the co-funding of their work through the European Regional Development Fund project REMIX under the grant agreement no. 80115106.

References

[1] R. Burke, *Hybrid recommender systems*, in User Modeling and User Adapted Interaction, 12 (2002), pp. 331–370.

[2] G. Adomavicius, and 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, 17(2005), pp. 734–749.

[3] J. A. Konstan, et. al., *Applying collaborative filtering to usenet news*,Commun. ACM. 40 (1997),pp. 77–87.

[4] B. Sarwar, G. Karypis, and et. al, *Item-Based Collaborative Filtering Recommendation Algorithms*, World Wide Web (WWW), 2001.

[5] R. Mamunur, et al. ,*Getting to Know You: Learning New User Preferences in Recommender Systems*, International Conference on Intelligent User Interfaces(IUI), pp. 127–134. ACM Press,2002.

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

[7] K. Yu, et al.,*Probabilistic memory-based collaborative filtering*, IEEE Transactions on Knowledge and Data Engineering, 16(2004), pp. 56–69.

[8] R. Nicholas, and A. McCallum, *Toward Optimal Active Learning through Monte Carlo Estimation of Error Reduction*, International Conference on Machine Learning (ICML), 2001.

[9] T. H Nguyen, and A. Smeulders, *Active Learning Using Pre-Clustering*, 21th International Conference on Machine Learning,2004.

[10] Y. Baram, R. EI-Yavin, et al., *Online Choice of Active Learning Algorithms*, Journal of Machine Learning research,5(2004), pp. 255–291.

[11] G. Schohn, D. Cohn, *Less is more: Active learning with support vector machines*, 17th International Conference on Machine Learning (ICML), 2000.

[12] S. Tong, and D. Koller, *Support vector machine active learning with applications to text classification*, 17th International Conference on Machine Learning(ICML), pp. 999-1006, 2000.

[13] H. S. Seung, M. Opper, H. Sompolinsky, *Query by committee*, Computational Learning Theory,pp. 287-294,1992.

[14] S. Tong, D. Koller, *Active Learning for Parameter Estimation in Bayesian Networks*, Advances in Neural Information Processing Systems (NIPS), 2000.

[15] D. Lewis, et. al., *A sequential algorithm for training text classifiers*, 17th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR), pp. 3-12, 1994.

[16] T. Hofmann, J. Puzicha, *Latent class models for collaborative filtering*, International Joint Conference on Artificial Intelligence, 1999.

[17] T. Hofmann, *Gaussian latent semantic models for collaborative filtering*, 26th ACM SIGIR Conference on Research and Development in Information Retrieval(SIGIR), 2003.

[18] R. Jin, and L. Si, *A Bayesian Approach toward Active Learning for Collaborative Filtering*,Uncertainty in Artificial Intelligence(UAI),2004.

[19] A. S. Harpale, Y. Yang, *Personalized Active Learning for Collaborative Filtering*, ACM SIGIR Conference on Research and Development in Information Retrieval(SIGIR),2008.

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

[21] C. Boutilier, et. al., *Active collaborative filtering*, 19th Conference on Uncertainty in Artificial Intelligence(UAI),2003.

[22] D. A. Cohn, Z. Ghahramani, and M. Jordan,*Active learning with statistical models*, Advances in Neural Information Processing Systems(NIPS),1995.

[23] T. Osugi, D. Kun, and S. Scott, *Balancing Exploration and Exploitation: A New Algorithm for Active Machine Learning*, Fifth IEEE International Conference on Data Mining (ICDM), 2005.

[24] C. Campbell, N. Cristiani, and A. SMOLA. *Query learning with large margin classifiers*, International Conference on Machine Learning (ICML),pp. 111118,2000.

[25] P. Jain, and A. Kapoor, *Active learning for large multi-class problems*, IEEE Conference on Computer Vision and Pattern Recognition (CVPR),2009.

[26] S. Rendle, L. S. Thieme, *Online-Updating Regularized Kernel Matrix Factorization Models for Large-Scale Recommender Systems*, ACM Conference on Recommender Systems (RecSys), ACM, 2008.