Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

Multi-Domain Active Learning for Recommendation

Zihan Zhang[†], **Xiaoming Jin**[†], **Lianghao Li**[‡], **Guiguang Ding**[†] and **Qiang Yang**[‡]
[†]Tsinghua National Laboratory for Information Science and Technology (TNList)
School of Software, Tsinghua University, Beijing, China
[‡]Hong Kong University of Science and Technology, Hong Kong
{zhangzihan.zac, luckycatcs}@gmail.com, {xmjin, dinggg}@tsinghua.edu.cn, qyang@cse.ust.hk

Abstract

Recently, active learning has been applied to recommendation to deal with data sparsity on a single domain. In this paper, we propose an active learning strategy for recommendation to alleviate the data sparsity in a *multi-domain* scenario. Specifically, our proposed active learning strategy simultaneously consider both specific and independent knowledge over all domains. We use the expected entropy to measure the generalization error of the domain-specific knowledge and propose a variance-based strategy to measure the generalization error of the domain-independent knowledge. The proposed active learning strategy use a unified function to effectively combine these two measurements. We compare our strategy with five state-of-the-art baselines on five different multi-domain recommendation tasks, which are constituted by three real-world data sets. The experimental results show that our strategy performs significantly better than all the baselines and reduces human labeling efforts by at least 5.6%, 8.3%, 11.8%, 12.5% and 15.4% on the five tasks, respectively.

Introduction

Recommendation techniques are widely used in many real-world applications, such as product recommendation in online shopping (Linden, Smith, and York 2003), friendship suggestion in social networks (Ding et al. 2013) or news recommendation in web portals (Lu et al. 2015). The performance of recommendation methods suffers from the data sparsity problem due to the expensive cost of acquiring ratings from users (Su and Khoshgoftaar 2009). Active learning techniques were proposed to alleviate the problem of data sparsity in recommendation systems (Boutilier, Zemel, and Marlin 2002; Elahi, Ricci, and Rubens 2014). Existing active learning strategies for recommendation mainly focus on querying ratings on a single domain (Harpale and Yang 2008; Houlsby, Hernández-Lobato, and Ghahramani 2014). However, we have many real-world recommendation problems that involve more than one domain. For example, most online shopping websites (e.g. Amazon) have more than one product domain, such as Book, Beauty, Movie, and etc. As ratings in different domains have different data distributions, simply merging all ratings from different domains (e.g. Movie and Book domains) for model training could hurt the recommendation quality. On the other hand, as some user preferences are shared by different domains (e.g. youth prefer sci-fi stories in both Movie and Book domains), training models for different domains separately could lead to a sub-optimal model. How to select the most informative products to generate new ratings poses a new challenge to active learning in multi-domain recommendation.

In this paper, we propose an active learning strategy for recommendation in the multi-domain scenario. The proposed strategy aims to select ratings by simultaneously considering both the domain-specific and domain-independent knowledge derived from multiple domains. Whether a user-item pair is informative enough to generate new rating is measured by the generalization error (Rubens, Kaplan, and Sugiyama 2011). By decomposing multiple rating matrices into several specific and independent factors, the generalization error of the model can be divided into a domain-specific part and a domain-independent part correspondingly. The generalization error of the domain-specific part is measured by the expected entropy, which is the existing active learning strategy used in the single domain recommendation scenario (Harpale and Yang 2008). For the domain-independent part, we propose to utilize variance to measure its generalization error. By splitting the optimization goal into two parts, our proposed active learning strategy can consider both the domain-independent and the domain-specific knowledge simultaneously, and therefore reduce the redundant rating efforts.

Note that the problem of multi-domain recommendation (Li, Yang, and Xue 2009b; Zhang, Cao, and Yeung 2012) is different from that of cross-domain recommendation (Pan and Yang 2013; Zhao et al. 2013). The goal of cross-domain recommendation is to utilize the knowledge derived from the source domain with sufficient ratings to alleviate the data sparsity in the target domain. However, in multi-domain recommendation, all domains suffer from the problem of data sparsity. The goal of multi-domain recommendation is to utilize the shared knowledge across multiple domains to alleviate the data sparsity in all domains. The key issue in active learning for multi-domain recommendation is to acquire ratings from one domain, which can contribute not only to the current domain but also other ones.

The main contributions of our work include: We propose an active learning strategy for multi-domain recommendation by considering both the domain-specific knowledge and domain-independent knowledge. To the best of our knowledge, this is the first work which applies active learning to recommendation in multi-domain scenario. The experimental results demonstrate its effectiveness and superiority on five multi-domain recommendation tasks.

Related Work

The recommendation quality always suffers from the data sparsity problem. Since acquiring enough ratings from users is time-consuming and expensive, active learning techniques were applied to recommendation to alleviate this problem (Boutilier, Zemel, and Marlin 2002; Rubens, Kaplan, and Sugiyama 2011).

Existing active learning strategies (Rubens, Kaplan, and Sugiyama 2011) select the user-item pair that can minimizes the generalization error to generate the new rating, which can save many human labeling efforts. The estimation of the generalization error can be categorized into three types: 1) expected entropy (Jin and Si 2004; Harpale and Yang 2008), which selects the user-item pair with minimum expected entropy; 2) mutual information (Silva and Carin 2012), which aims to select the user-item pair that is most different from current observed ratings; 3) variance (Sutherland, Póczos, and Schneider 2013), which selects the user-item pair with highest prediction variance. According to the adopted recommendation techniques, the aforementioned active learning works can be divided into two types: aspect model (Boutilier, Zemel, and Marlin 2002) and matrix factorization (Wang, Srebro, and Evans 2014). However, all previous active learning strategies focused on querying ratings from a single domain. To the best of our knowledge, no previous active learning strategy for recommendation work can consider not only the domain-specific knowledge but also domain-independent knowledge that shared among multiple domains.

Most multi-domain recommendation works focused on deriving the connection among multiple domains (Li and Lin 2014), which can be utilized in our multi-domain active learning recommendation strategy. Li, Yang, and Xue(2009b) proposed a multi-domain recommendation method by constructing a cluster-level rating matrix shared across multiple domains. Zhang, Cao, and Yeung(2012) proposed to utilize a link function to connect all domains. Also, there are several other methods for multi-domain recommendation (Singh and Gordon 2008).

Our work is also related to cross-domain active learning recommendation. Zhao et al.(2013) proposed an active learning strategy for cross-domain recommendation. However, it is quite different from our problem. They utilize active learning to solve the problem of entity-correspondence in transfer learning and the densities of two domains vary substantially. In our problem, we propose an active learning for the problem of data sparsity in multi-domain recommendation and all domains are sparse.

Multi-domain active learning for text classification (Li et al. 2012) is relevant to our work. However, active learning for classification is quite different from active learning for recommendation as classification and recommendation have quite different optimization goals. Therefore active learning strategy for classification cannot be directly applied to solve the recommendation problem.

Problem Definition

In this section, we first present some definitions and then give a formal definition of the problem addressed in this paper.

Definition 1 *(Domain) A domain is a collection of ratings which are drawn under the same data distribution.*

For example, ratings collected from different types of products, such as Movie, Music, Electronics, can be regarded as different domains. More specifically, as defined in the previous work (Li, Yang, and Xue 2009b), we regard ratings collected from one web site as a domain in this paper. For example, rating collection from a DVD online shopping website is regarded as a domain.

Definition 2 *(Multi-Domain Recommendation) Given a set of ratings collected from D domains. Let $\mathcal{X} = \{\mathcal{X}^{(1)}, \mathcal{X}^{(2)}, ..., \mathcal{X}^{(D)}\}$ be the training set and $\mathcal{T} = \{\mathcal{T}^{(1)}, \mathcal{T}^{(2)}, ..., \mathcal{T}^{(D)}\}$ be the testing set. The task is to train an accurate model to predict the rating $r_{uv}^{(d)}$ in the test set $\mathcal{T}$*

Definition 3 *(Multi-Domain Active Learning for Recommendation) Given a test set $\mathcal{T}$ and a rating pool $\mathcal{P} = \{\mathcal{P}^{(1)}, \mathcal{P}^{(2)}, ..., \mathcal{P}^{(D)}\}$ where each user-item pair $(u^{(d)}, v^{(d)}) \in \mathcal{P}^{(d)}$ represents that user u has interacted with item v in the d-th domain but has not rated it, the task is to build an accurate prediction model by selecting as few ratings as possible from the rating pool $\mathcal{P}$.*

Our active strategy is based on pool-based sampling. In the pool-based sampling, the active learning strategy is to iteratively select the most informative items from the pool, which is assumed to be stationary, for users to rate and add the ratings to the training set (Houlsby, Hernández-Lobato, and Ghahramani 2014). Typically, the learning system scans the unlabeled item pool and chooses the most informative item to ask for users' ratings.

Our Solution

In this section, we will first define the general active learning strategy. Then, we will briefly review the multi-domain recommendation method adopted in this paper. Finally, we present our active learning strategy for multi-domain recommendation.

The General Active Learning Strategy

Existing active learning strategies for recommendation on a single domain select the most informative user-item pair that can minimize the generalization error (i.e. expected entropy) (Harpale and Yang 2008; Rubens, Kaplan, and Sugiyama 2011). However, in multi-domain recommendation scenario, the learning system should measure the global generalization error by considering both the domain-specific error and domain-independent error.

In this paper, we propose to select the most informative user-item pair $(u^{(d)}, v^{(d)})^*$ from the pool $\mathcal{P}$ which can minimize the global generalization error. And the global generalization error $\widehat{G}$ is defined as follow:

$$\widehat{G}(u^{(d)}, v^{(d)}) = \mu\widehat{G}_{in} + \omega\widehat{G}_{sp} \quad (1)$$

We divide the global generalization error $\widehat{G}$ introduced by the user-item pair $(u^{(d)}, v^{(d)})$ into a domain-independent part $\widehat{G}_{in}$ and a domain-specific part $\widehat{G}_{sp}$, where μ and ω are parameters, which control the influence between these two parts. Note that only one parameter (μ or ω) is needed to decide $\widehat{G}(u^{(d)}, v^{(d)})$, but it is more convenient to define the baselines by using two parameters.

Our multi-domain active learning optimization goal can be formulated as follows:

$$(u^{(d)}, v^{(d)})^* = \underset{(u^{(d)}, v^{(d)}) \in \mathcal{P}}{\arg\min}\ \widehat{G}(u^{(d)}, v^{(d)}) \quad (2)$$

Multi-Domain Recommendation

This section reviews the multi-domain recommendation method used in this paper. Recently, several multi-domain recommendation models have been proposed. Rating-Matrix Generative Model ($RMGM$) (Li, Yang, and Xue 2009b) is one of the state-of-the-art method among them. $RMGM$ is a two-side aspect model (Hofmann and Puzicha 1999) for mining shared knowledge across different domains and uses transfer learning technique to alleviate the sparsity problem in cross-domain recommendation, which can be easily extended to multi-domain recommendation. In $RMGM$, a cluster-level rating matrix is learned to capture the shared relationship between user and item groups across domains. More specifically, the rating of each user-item pair $(u^{(d)}, v^{(d)})$ is generated as follows:

$$r_{uv}^{(d)} = \sum_r r \sum_{kl} P(r|c_u^{(k)}, c_v^{(l)}) P(c_u^{(k)}|u^{(d)}) P(c_v^{(l)}|v^{(d)}) \quad (3)$$

$c_u^{(k)}$ and $c_v^{(l)}$ represent the user group of user $u^{(d)}$ and the item group of item $v^{(d)}$, respectively. $P(c_u^{(k)}|u)$ and $P(c_v^{(l)}|v)$ are the probability of the user $u^{(d)}$ belonging to the user group $c_u^{(k)}$ and the probability of the item $v^{(d)}$ belonging to the item group $c_v^{(l)}$, separately. $P(r|c_u^{(k)}, c_v^{(l)})$ is the probability of the user-item pair $(u^{(d)}, v^{(d)})$ co-cluster. Equation 3 can be simplified as below.

$$r_{uv}^{(d)} = \mathbf{p}_{u^{(d)}}^\top \mathbf{B} \mathbf{q}_{v^{(d)}} \quad (4)$$

where $P(c_u^{(k)}|u)$ and $P(c_v^{(l)}|v)$ are simplified as $[\mathbf{p}_u]_k$ and $[\mathbf{q}_v]_l$, respectively. And $P(r|c_u^{(k)}, c_v^{(l)})$ is simplified to $\mathbf{B}_{u,v}$. Note that $\mathbf{p}_{u^{(d)}}$ and $\mathbf{q}_{v^{(d)}}$ are specific in different domains and $\mathbf{B}$ is shared by all domains.

In addition to $RMGM$, several other multi-domain recommendation works about mining the shared knowledge across multiple domains can be integrated into the proposed strategy, such as Collective Matrix Factorization(CMF) (Singh and Gordon 2008), CodeBook Transfer(CBT) (Li, Yang, and Xue 2009a). CBT is very similar to $RMGM$. In CMF, user factors are shared across domains and item factors are specific across different domains, so user factor is the shared knowledge and item factor is the specific knowledge. Both models can be easily adopted in our strategy by using the domain-independent knowledge and domain-specific knowledge correspondingly.

Multi-Domain Active Learning Strategy

In this section, we will present our active learning strategy for multi-domain recommendation in detail.

Multi-domain recommendation model $RMGM$ adopted in this paper can be divided into a domain-independent part (i.e. $\mathbf{B}$) and a domain-specific part (i.e. $\mathbf{p}_{u^{(d)}}$ and $\mathbf{q}_{v^{(d)}}$). According to Equation 1, the global generalization error introduced by an user-item pair $(u^{(d)}, v^{(d)})$ can be specified as follows:

$$\widehat{G}(u^{(d)}, v^{(d)}) = \mu\widehat{G}_{\mathbf{B}} + \omega(\widehat{G}_{\mathbf{p}_{u^{(d)}}} + \widehat{G}_{\mathbf{q}_{v^{(d)}}}) \quad (5)$$

where $\widehat{G}_{\mathbf{p}_{u^{(d)}}}$ and $\widehat{G}_{\mathbf{q}_{v^{(d)}}}$ are domain-specific parts of the global generalization error and $\widehat{G}_{\mathbf{B}}$ is the domain-independent part of it.

For the domain-specific factors $\mathbf{p}_{u^{(d)}}$ and $\mathbf{q}_{v^{(d)}}$, we apply the existing single domain active learning strategy for aspect recommendation model (Hofmann 2004) as follows:

$$p(r|u, v) = \sum_{k \in K} p(r|v, k) p(k|u) \quad (6)$$

The generalization error (i.e. expected entropy) (Jin and Si 2004) of the model is calculated as follows:

$$\widehat{G}(u^{(d)}, v^{(d)}) = -\langle \sum_k [\theta_{u|v,r}]_k \log[\theta_{u|v,r}]_k \rangle_{p(r|v,u)} \quad (7)$$

where $[\theta_u]_k$ denotes the user-group mixture probability $p(k|u)$ and $[\theta_{u|v,r}]_k$ denotes the model posterior after retraining the aspect model with the estimated rating r_{uv}.

However, in order to select the most informative instance, the above strategy needs to re-train the model after adding each user-item pair with each possible rating (i.e. 1, 2, 3, 4, 5), which is extremely time-consuming. To speed up this strategy, we propose to use the posterior $\mathbf{p}_{u^{(d)}}$ and $\mathbf{q}_{v^{(d)}}$ of current iteration to approximate the expected entropy of the model for each user-item pair $(u^{(d)}, v^{(d)})$. Therefore, we only need to update the posterior once after adding new acquired ratings from the pool. For each user-item pair $(u^{(d)}, v^{(d)})$, the expected entropy of model introduced by domain-specific factors $\mathbf{p}_{u^{(d)}}$ and $\mathbf{q}_{v^{(d)}}$ can be computed as follows:

$$\widehat{G}_{\mathbf{p}_{u^{(d)}}} = -\sum_k [p_{u^{(d)}}]_k \log[p_{u^{(d)}}]_k \quad (8)$$

$$\widehat{G}_{\mathbf{q}_{v^{(d)}}} = -\sum_l [q_{v^{(d)}}]_l \log[q_{v^{(d)}}]_l \quad (9)$$

Recall that, each element in matrix $\mathbf{B}$ represents the preference of one user-group on one item-group. For every candidate user-item pair $(u^{(d)}, v^{(d)})$, $\mathbf{p}_{u^{(d)}}$ is the user-group distribution of user $u^{(d)}$ and $\mathbf{q}_{v^{(d)}}$ is the item-group distribution of item $v^{(d)}$. Accordingly, the probability of choosing the

Algorithm 1: Multi-Domain Active Learning

Input : (1) A pool $\mathcal{P}$ of unrated user-item pairs which are collected from D domains; (2) Number of initial ratings in each domain Ini; (3) Number of iteration $Iter$; (4) Number of new labeled ratings per iteration $S(\times D)$

Output: The recommendation model $\boldsymbol{\mathcal{M}}$;

Randomly initialize Ini ratings of each domain to construct the training set $\mathcal{X}$;

Learn $\mathbf{p}_{u^{(d)}}$, $\mathbf{q}_{v^{(d)}}$ and $\mathbf{B}$ by training $RMGM$ on $\mathcal{X}$;

for $i \leftarrow 1$ **to** $Iter$ **do**

  foreach *user-item pair* $(u^{(d)}, v^{(d)}) \in \mathcal{P}$ **do**

    Estimate the global generalization error $\widehat{G}(u^{(d)}, v^{(d)})$;

  end

  Query the ratings Q^* of the $S(\times D)$ user-item pairs $(u^{(d)}, v^{(d)})^*$ with least global generalization error;

  Update the training set by $\mathcal{X} \leftarrow \mathcal{X} \cup ((u^{(d)}, v^{(d)})^*, Q^*)$, and remove $(u^{(d)}, v^{(d)})^*$ from $\mathcal{P}$;

  Update $\mathbf{p}_{u^{(d)}}$, $\mathbf{q}_{v^{(d)}}$ and $\mathbf{B}$ by re-training $RMGM$ on the new training set $\mathcal{X}$;

end

preference in $\mathbf{B}$ for each user-item pair $(u^{(d)}, v^{(d)})$ can be calculated as follows:

$$\mathbf{\Phi} = \mathbf{p}_{u^{(d)}}^{\top} \mathbf{q}_{v^{(d)}} \tag{10}$$

For each user-item pair $(u^{(d)}, v^{(d)})$ in the rating pool $\mathcal{P}$, the variance of the predicted rating caused by $\mathbf{B}$ can be computed as follows:

$$\mathbb{V}_{(u^{(d)}, v^{(d)})}[\mathbf{B}] = \mathbb{E}_{\mathbf{\Phi}}[\mathbf{B} - \mathbb{E}_{\mathbf{\Phi}}(\mathbf{B})]^2 \tag{11}$$

where $\mathbb{E}_{\mathbf{\Phi}}(\mathbf{B})$ is the expectation of the rating produced by $\mathbf{B}$ for the candidate user-item pair $(u^{(d)}, v^{(d)})$. A larger variance implies more uncertainty about the prediction. Therefore, for the domain-independent part, our active learning strategy is to select user-item pair with the largest variance. And our total goal is to minimize $\widehat{G}_{\mathbf{B}}$. The generalization error for $\mathbf{B}$ is defined as follows:

$$\widehat{G}_{\mathbf{B}} = -\mathbb{V}_{(u^{(d)}, v^{(d)})}[\mathbf{B}] \tag{12}$$

The overall algorithm of our active learning strategy for multi-domain recommendation is shown in Algorithm 1.

Experiments

To evaluate the effectiveness and superiority of our strategy, we conduct the experiments on five different tasks which are constructed by four real-world domains (i.e., DoubanBook[1], MovieLens[2], eachMovie[3] and Netflix[4]).

[1]http://www.douban.com

[2]http://www.grouplens.org/datasets/movielens/

[3]http://www.cs.cmu.edu/ lebanon/IR-lab.htm

[4]http://www.netflix.com

Data Sets

We first preprocess the data sets in a way similar to (Li, Yang, and Xue 2009b). This step can provide active learning strategies with the largest pool of ratings to select from (Houlsby, Hernández-Lobato, and Ghahramani 2014).

Doubanbook (DB): A collection of book ratings consists of 3.6 million ratings (ranging from 1 to 5). We randomly select 774 users and 1360 items both with more than 20 ratings to comprise **DB** domain.

MovieLens (ML): It consists of 1 million ratings (ranging from 1 to 5). We randomly select 891 users and 1029 items to comprise **ML** domain.

EachMovie (EM): This domain consists of 2.8 million ratings (ranging from 1 to 6). We randomly select 809 users and 756 items both with more than 20 ratings to comprise **EM** domain. For unity, we replace rating 6 with 5 in the data set which is similar to (Li, Yang, and Xue 2009b).

Netflix (NF): A collection of movie ratings contains 100 million ratings (ranging from 1 to 5). We randomly select 971 users and 784 items both with more than 20 ratings to comprise **NF** domain.

After the preprocessing phase, we obtain four domains: **DB**, **ML**, **EM**, **NF**. The density, which is calculated by Equation 13, and the rating number of each domain are shown in Table 1.

$$density(Z) = O/(M \times N) \tag{13}$$

where Z is the dataset of one domain. O, M and N are the numbers of the observed ratings, users and items, respectively.

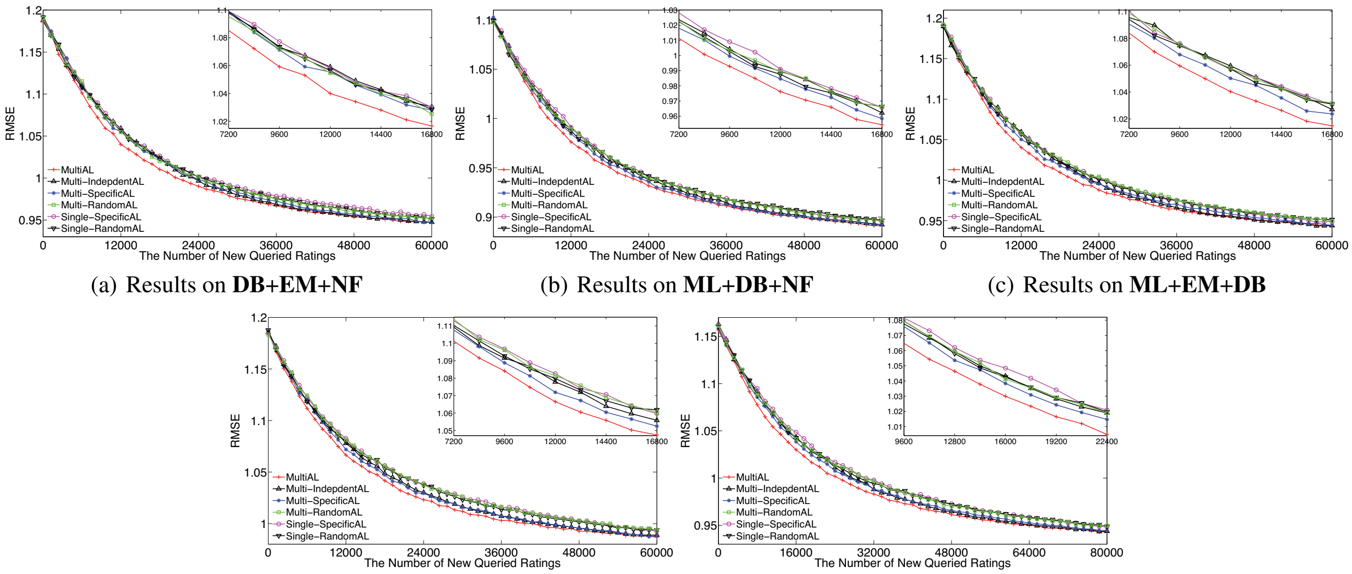
Table 1: The density and rating amount of four domains

Domain / Metric	**DB**	**ML**	**EM**	**NF**
Density	2.94%	6.52%	6.24%	6.62%
Rating amount	30910	59746	38149	50396

By selectively combining some of the domains above, we obtain five different multi-domain recommendation tasks denoted by **DB+EM+NF**, **ML+DB+NF**, **ML+EM+DB**, **ML+EM+NF**, **ML+EM+NF+DB**. For example, **ML+EM+NF** denotes the task including MovieLens, EachMovie and Netflix domain.

Baseline Methods

In the experiment, we compare our active learning strategy, which is denoted by MultiAL, with the following five state-of-the-art strategies: (1) Single-RandomAL is to randomly select the same amount of user-item pairs from each domain separately; (2) Multi-RandomAL is to randomly select user-item pairs from all domains together. The performance difference between first two strategies is mainly influenced by whether the amount of ratings in each domain is balanced; (3) Single-SpecificAL is to separately select the same amount of user-item pairs from each domain according to domain-specific part of the generalization error (set μ to



(a) Results on **DB+EM+NF** (b) Results on **ML+DB+NF** (c) Results on **ML+EM+DB**

(d) Results on **ML+EM+NF** (e) Results on **ML+EM+NF+DB**

Figure 1: Performances of different active learning strategies on five multi-domain recommendation tasks.

0 in Equation 1); (4) Multi-SpecificAL is to select the user-item pair according to domain-specific part from all domains without considering which domain it belongs to. Different from the previous one, this strategy can select global optimal instance with regard for domain-specific knowledge; (5) Multi-IndependentAL is to select the most informative user-item pair according to the domain-independent part (set ω to 0 in Equation 1). Note that our strategy and the strategies with the prefix Multi- belong to multi-domain active learning strategies and the remaining strategies belong to single-domain active learning strategies.

Setting Ups and Evaluation

In the experiments, we randomly divided the whole rating data set into three parts: training set (10%), test set (20%) and rating pool (70%). The ratings in rating pool are assumed to be those users who interacted with items but did not rate. In each active iteration, we query $400 \times D$ unknown ratings of the user-item pairs from $\mathcal{P}$ and then add the rating of each user-item pair into the training set, where D is the number of domains in the task. After that, the multi-domain recommendation model is re-trained on the new training set. In total, we do 50 active iterations, which totally queries $20000 \times D$ ratings from $\mathcal{P}$. All the experimental results reported in this paper are averaged over 10 random runs. For the MultiAL, both μ and ω in Equation 5 are set to 1 for the assumption that the specific knowledge and independent knowledge are equally important.

The root mean squared error ($RMSE$) is adopted as the accuracy metric. It is defined as follows:

$$RMSE = \sqrt{\sum_{(u,v)\in\mathcal{T}} (r_{uv} - \hat{r}_{uv})^2/|\mathcal{T}|} \qquad (14)$$

where $\mathcal{T}$ represents the test set, r_{uv} is the correct rating and $\hat{r}_{uv}$ is the predicted rating.

In active learning problem, how much rating efforts can be saved is a unique and important metric which can directly reflect the performance of the active learning strategy. In this paper, we measure this metric by comparing the required ratings to achieve the same RMSE reduction between the proposed strategy and baselines.

Results and Discussion

Performances on Different Tasks Figure 1 shows the whole 50 active iterations of MultiAL with five baselines on the five tasks. From Figure 1, the following conclusions can be easily observed. Firstly, MultiAL performs better than all the baselines on all tasks. Secondly, multi-domain strategies except Multi-RandomAL are better than single-domain strategies, which demonstrates the necessity to consider the independent knowledge in multi-domain active learning strategy. Thirdly, all the strategies tend to converge to the same point in the last few iterations, which means that the model is learned well enough on that density of training set. Finally, all curves in figures seem to be close. The reason is that we show the whole active procedure and the RMSE reduction is quite a lot, which makes all curves look close. Therefore, in the sub-window of Figure 1, we zoom in on the area around the 10-th iteration (acquiring $4000 \times D$ ratings). It can be seen that MultiAL is significantly better than all the baselines. Moreover, we use how much rating efforts can be saved and significant tests to measure the superiority of MultiAL over baselines.

Table 2 summarizes the $RMSE$ on each domain of the task **ML+EM+NF** after obtaining $4000\times D$ ratings (10-th iteration). It can be observed that the best overall performance is achieved by MultiAL. More specifically, some baselines

Table 2: The $RMSE$ of each domain in task **ML+EM+NF** after 12000 new ratings being queried

Method / Domain	**MultiAL**	**Multi -IndependentAL**	**Multi -SpecificAL**	**Multi -RandomAL**	**Single -SpecificAL**	**Single -RandomAL**
ML	**1.0125**	1.0343	1.0126	1.0174	1.0199	1.0279
EM	1.2003	**1.1857**	1.2193	1.2426	1.2314	1.2234
NF	1.0207	1.0425	**1.0202**	1.0230	1.0337	1.0229
Overall	**1.0665**	1.0779	1.0719	1.0815	1.0826	1.0798

Table 3: The Required Number of New Labeled User-Item Pairs to achieve 0.1 reduction on $RMSE$

Comparison / Task	**MultiAL**	**Multi -IndependentAL**	**Multi -SpecificAL**	**Multi -RandomAL**	**Single -SpecificAL**	**Single -RandomAL**
DB+EM+NF	**7200** (16.7%)	9600	8400	8400	9600	9600
ML+DB+NF	**8400** (14.3%)	10800	9600	10800	12000	10800
ML+EM+DB	**7200** (16.7%)	8400	8400	8400	8400	8400
ML+EM+NF	**9600** (12.5%)	12000	10800	12000	12000	10800
ML+EM+NF+DB	**11200** (14.3%)	14400	12800	14400	14400	14400

Table 4: The Required Number of New Labeled User-Item Pairs to achieve 0.15 reduction on $RMSE$

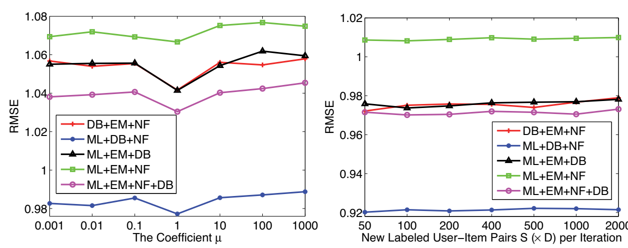
Comparison / Task	**MultiAL**	**Multi -IndependentAL**	**Multi -SpecificAL**	**Multi -RandomAL**	**Single -SpecificAL**	**Single -RandomAL**
DB+EM+NF	**13200** (15.4%)	15600	15600	15600	16800	16800
ML+DB+NF	**18000** (11.8%)	20400	20400	21600	21600	21600
ML+EM+DB	**13200** (8.3%)	15600	14400	15600	15600	15600
ML+EM+NF	**20400** (5.6%)	22800	21600	25200	25200	25200
ML+EM+NF+DB	**22400** (12.5%)	25600	25600	28800	28800	27200

Table 5: The p-values after 20 active iterations (Querying $8000 \times D$ ratings)

Metric / Task	**P-value**
DB+EM+NF	1.51E-03
ML+DB+NF	5.22E-03
ML+EM+DB	3.70E-04
ML+EM+NF	1.01E-03
ML+EM+NF+DB	4.42E-04

are better than MultiAL on some specific domains, but they cannot perform equally well on other domains. Table 3 and 4 show the required number of ratings to achieve 0.1 and 0.15 reduction on RMSE over the five tasks, respectively. It can be seen that MultiAL uses much less ratings than each baseline and numbers in parentheses are the percentage of saved rating efforts over the best baseline on each task.

Significance Tests The significance metric p-value is adopted to evaluate the improvements gained by MultiAL against baselines after 20 active iterations (querying $8000 \times D$ ratings). In statistics, a p-value less than 0.01 means extremely strong evidence against the null hypothesis. Table 5 shows p-values of our strategy against the best baseline on the five tasks. All the p-values are far smaller than 0.01, which means that MultiAL is significantly better than the best baseline on the five tasks.



(a) (b)

Figure 2: Influence of (a) Parameter μ and (b)$S(\times D)$ per iteration

Parameter Sensitivity In our strategy, μ is an important parameter which controls the balance between the specific knowledge and independent knowledge. Besides, the number of queried ratings per iteration ($S \times D$ for our problem, controlled by parameter S) is a unique parameter in active learning problem. The sensitivities result of two parameters are shown in Figure 2. Figure 2(a) visualizes the performances by varying μ from 0.001 to 1000 and setting ω to 1 after querying $4000 \times D$ ratings (10-th iteration) on the five tasks. The results under setting μ to 1 are better than other parameter settings, which proves the rationality of our parameter settings. Figure 2(b) shows the results after querying totally $10000 \times D$ ratings by varying the number of new labeled user-item pairs $S \times D$ per iteration from $50 \times D$ to $2000 \times D$. The results shows that our strategy works stably when S varies from 50 to 2000.

Conclusion and Future Work

In this paper, we address a novel active learning method for multi-domain recommendation. Different from previous active learning works that focused on querying ratings in a single domain, the proposed multi-domain active learning strategy tries to query ratings by simultaneously considering both the domain-specific knowledge and domain-independent knowledge across multiple domains. In this way, more rating efforts can be saved. The experimental results on five tasks show that our proposed strategy significantly and stably outperforms the five baseline strategies. In future, we intend to promote our work in the following directions: (1) design a method to automatically adjust the balance between the domain-specific knowledge and the domain-independent knowledge, and (2) design a more efficient method that updates the model only when necessary instead of updating once per iteration.

Acknowledgements

The work is supported by National Basic Research Program of China (2015CB352300) and National Natural Science Foundation of China (61571269).

References

Boutilier, C.; Zemel, R. S.; and Marlin, B. 2002. Active collaborative filtering. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, 98–106. Morgan Kaufmann Publishers Inc.

Ding, X.; Jin, X.; Li, Y.; and Li, L. 2013. Celebrity recommendation with collaborative social topic regression. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, 2612–2618. AAAI Press.

Elahi, M.; Ricci, F.; and Rubens, N. 2014. Active learning in collaborative filtering recommender systems. In *E-Commerce and Web Technologies*. Springer. 113–124.

Harpale, A. S., and Yang, Y. 2008. Personalized active learning for collaborative filtering. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 91–98. ACM.

Hofmann, T., and Puzicha, J. 1999. Latent class models for collaborative filtering. In *IJCAI*, volume 99, 688–693.

Hofmann, T. 2004. Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems (TOIS)* 22(1):89–115.

Houlsby, N.; Hernández-Lobato, J. M.; and Ghahramani, Z. 2014. Cold-start active learning with robust ordinal matrix factorization. In *Proceedings of The 31th International Conference on Machine Learning*, 766–774. ACM.

Jin, R., and Si, L. 2004. A bayesian approach toward active learning for collaborative filtering. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, 278–285. AUAI Press.

Li, C.-Y., and Lin, S.-D. 2014. Matching users and items across domains to improve the recommendation quality. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 801–810. ACM.

Li, L.; Jin, X.; Pan, S. J.; and Sun, J.-T. 2012. Multi-domain active learning for text classification. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1086–1094. ACM.

Li, B.; Yang, Q.; and Xue, X. 2009a. Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction. In *IJCAI*, volume 9, 2052–2057.

Li, B.; Yang, Q.; and Xue, X. 2009b. Transfer learning for collaborative filtering via a rating-matrix generative model. In *Proceedings of the 26th Annual International Conference on Machine Learning*, 617–624. ACM.

Linden, G.; Smith, B.; and York, J. 2003. Amazon. com recommendations: Item-to-item collaborative filtering. *Internet Computing, IEEE* 7(1):76–80.

Lu, Z.; Dou, Z.; Lian, J.; Xie, X.; and Yang, Q. 2015. Content-based collaborative filtering for news topic recommendation. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Pan, W., and Yang, Q. 2013. Transfer learning in heterogeneous collaborative filtering domains. *Artificial intelligence* 197:39–55.

Rubens, N.; Kaplan, D.; and Sugiyama, M. 2011. Active learning in recommender systems. In *Recommender Systems Handbook*. Springer. 735–767.

Silva, J., and Carin, L. 2012. Active learning for online bayesian matrix factorization. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 325–333. ACM.

Singh, A. P., and Gordon, G. J. 2008. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 650–658. ACM.

Su, X., and Khoshgoftaar, T. M. 2009. A survey of collaborative filtering techniques. *Advances in artificial intelligence* 2009:4.

Sutherland, D. J.; Póczos, B.; and Schneider, J. 2013. Active learning and search on low-rank matrices. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 212–220. ACM.

Wang, J.; Srebro, N.; and Evans, J. 2014. Active collaborative permutation learning. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 502–511. ACM.

Zhang, Y.; Cao, B.; and Yeung, D.-Y. 2012. Multi-domain collaborative filtering. In *Twenty-Sixth Conference on Uncertainty in Artificial Intelligence (UAI 2010)*, 725–732.

Zhao, L.; Pan, S. J.; Xiang, E. W.; Zhong, E.; Lu, Z.; and Yang, Q. 2013. Active transfer learning for cross-system recommendation. In *AAAI*.