Informative Household Recommendation with Feature-based Matrix Factorization

Qiuxia Lu, Diyi Yang, Tianqi Chen, Weinan Zhang, Yong Yu
Shanghai Jiao Tong University
800 Dongchuan Road, Shanghai China
{luqiuxia, yangdiyi, tqchen, wnzhang, yyu}@apex.sjtu.edu.cn

ABSTRACT

In this paper, we describe our solutions to the first track of CAMRa2011 challenge. The goal of this track is to generate a movie ranking list for each household. To achieve this goal, we propose to use the ranking oriented matrix factorization and the matrix factorization with negative examples sampling. We also adopt feature-based matrix factorization framework to incorporate various contextual information to our model, including user-household relations, item neighborhood, user implicit feedback, etc.. Finally, we elaborate two kinds of methods to recommend movies for each household based on our models. Experimental results show that our proposed approaches achieve significant improvement over baseline methods.

Categories and Subject Descriptors

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

General Terms

Models, Experimentation

Keywords

Collaborative Filtering, Recommender Systems, Matrix Factorization

1. INTRODUCTION

While benefiting from the large amount of information on the World Wide Web, we are also challenged by its overload with an explosive growing rate. In dealing with such problem, recommender systems, which try to recommend products, news and other information automatically for users, have become a particularly useful tool for Web services. Collaborative filtering(CF), based on the observed users' preferences on items without explicit profiles of users/items, have drawn much attention recently. Some widely publicized recommendation contests such as Netflix prize[1] and Yahoo! Music Recommendation (KDD-Cup 2011)[2] have made great contribution to the research and industry fields of collaborative filtering recommendation.

The first track of the CAMRa2011 challenge is to recommend a movie ranking list for each household. However, it has some differences from the traditional recommendation task in the following aspects. Firstly, in contrast to the traditional datasets, there are many kinds of contextual information such as time stamps of user ratings and users' membership information in a household, which is not considered in traditional CF. Secondly, the goal is to generate a movie ranking list instead of rating prediction. Thirdly, the recommending targets are households consisting of user members, making it a group targeted recommendation problem, which has not been well learnt so far.

In this paper, we propose several approaches to solve the choice prediction problem [12]. Experimental results show that our proposed methods give significant improvement over baseline methods. To sum up, the key contribution of our work is as follows:

- **Informative Models** - We incorporate rich contextual information to form the informative models, such as the relation between users and households. It proves to be significantly helpful in our work.
- **Ranking Oriented Matrix Factorization** - We propose a new ranking matrix factorization approach based on the work of [12] to generate ranking lists directly without the extra rating prediction step.
- **Negative Examples Sampling** - We adopt the negative examples sampling scheme and conduct experiments which show that negative sampling is extremely important and effective in our problem.
- **Household Recommendation** - We have tried several approaches to recommend movies to household instead of individual users.

The remainder of this paper is structured as follows. Section 2 presents preliminaries. Our main approaches are described in Section 3, Section 4 and 5. Section 6 analyzes experimental results with related work summarized in Section 7. Section 8 concludes our work.

[1] http://www.netflixprize.com
[2] http://kddcup.yahoo.com

CAMRa2011, October 27, 2011, Chicago, IL, USA.
Copyright 2011 ACM 978-1-4503-0825-0 ...$10.00.

2. PRELIMINARIES

In this section, we will briefly present two classical models which are the foundation of our models for CAMRa2011.

2.1 Problem Statement

The first track of CAMRa2011 challenge is recommending a list of movies for household based on basic user-item rating matrix. The recommended lists should contain relative movies as many as possible for each household.

The dataset consists of around 4.5M ratings assigned by about 17K users to about 2.4K items. Besides, relations between users and households, the number of recommended movies and time stamps of ratings are also provided.

Here are some notational conventions. Suppose we have m users and n items. The rating matrix R $(m \times n)$ has its entries r_{ui}, meaning the predicted rating of user u on the corresponding item i.

2.2 Classical Neighborhood Model

The classical neighborhood model estimates the unknown ratings of a given item using the ratings of its neighbor items(users), corresponding to item-oriented and user-oriented models respectively. Neighborhood is built based on similarity of ratings. We adopt item-oriented approach due to its wide application and popularity [10].

$$\hat{r}_{u,i} = b_{u,i} + \frac{\sum_{j \in S^k_{(i;u)}} s_{i,j}(r_{u,j} - b_{u,j})}{\sum_{j \in S^k_{(i;u)}} s_{i,j}}, \quad (1)$$

Here $s_{i,j}$ denotes the similarity between items i and j. $b_{u,i}$ is the baseline estimation capturing user bias b_u, item effect b_i and average ratings μ:

$$b_{u,i} = \mu + b_u + b_i. \quad (2)$$

Prevalent similarity implementations include Pearson Correlation Coefficient and cosine similarity.

2.3 Matrix Factorization for Collaborative Filtering

Matrix factorization (MF) has drawn much attention these days for its success in scalability. MF uses the observed user-item ratings to train model, predicting the unobserved users' ratings. Every user u and item i are associated with vectors $p_u, q_i \in \mathbb{R}^k$. The rating of user u on item i is predicted as follows:

$$\hat{r}_{u,i} = f(p_u^T q_i + b_u + b_i), \quad (3)$$

where f is a function mapping the real number to a certain range like [0,1]. In this work, we adopt the sigmoid function below.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (4)$$

By solving the following regularized least squares problems, all parameters of the MF model are learnt. Here λ_1 and λ_2 are parameters guarding the extent of regularization.

$$\min_{b^*,p^*,q^*} \sum_{(u,i)\in K} (r_{u,i} - p_u^T q_i - b_u - b_i)^2 +$$
$$\lambda_1(\sum_u \|p_u\|^2 + \sum_i \|q_i\|^2) + \lambda_2(\|b_u\|^2 + \|b_i\|^2) \quad (5)$$

3. CHOICE PREDICTION MODELS

In the first track of CAMRa2011, given the users' rating history data, we should predict which movies the household will watch later based on observed data. In the work of [12], this task is called *choice prediction.* Our approaches to solve the choice prediction problem will be elaborated in the following.

3.1 Ranking Matrix Factorization

A natural solution to choice prediction is learning to rank using the implicit information provided in training data. Some learning to rank CF models have been proposed recently. These models address the ranking problem directly without the rating prediction step in traditional CF problems. EigenRank [13] and probabilistic latent preference analysis [14] are two models that advocate ranking oriented approach. Their key idea is to preserve the user dependent order of items instead of approximating the scores. In these models, the item choice is turned into user dependent pairs, denoted by $\delta_{u,i,j}$, which is equal to $+1$ if user u prefers item i over j, or -1 otherwise.

Nathan N.Liu et al. [12] generates pairwise preferences based on both rating values and rating occurrences. In their rules, if two items are both rated by a user, they derive $\delta_{u,i,j}$ by comparing $r_{u,i}$ and $r_{u,j}$. In addition, items rated by a user are more preferable to those have not been rated by him or her. The pairwise preference generation rule is stated as follows:

$$\delta_{u,i,j} = \begin{cases} +1 & i,j \in I_u \text{ and } r_{u,i} > r_{u,j} \\ +1 & i \in I_u \text{ and } j \notin I_u \\ -1 & i,j \in I_u \text{ and } r_{u,i} < r_{u,j} \\ -1 & i \notin I_u \text{ and } j \in I_u \end{cases} \quad (6)$$

Our approach is different from Equation 6. We derive $\delta_{u,i,j} = +1$ in the following two situations: (a) Item i, j are both rated by user u, and u gives a much higher rate to i than j. (b) User u gives a high rate to item i while item j is not rated by u. This leads to the following rule:

$$\delta_{u,i,j} = \begin{cases} +1 & i,j \in I_u \text{ and } r_{u,i} - r_{u,j} > r_1 \\ +1 & i \in I_u \text{ and } j \notin I_u \text{ and } r_{u,i} > r_2 \\ -1 & i,j \in I_u \text{ and } r_{u,j} - r_{u,i} > r_1 \\ -1 & i \notin I_u \text{ and } j \in I_u \text{ and } r_{u,j} > r_2 \end{cases} \quad (7)$$

Equation 6 is a special case of Equation 7(by setting $r_1 = 0, r_2 = 0$). When $r_1 > 0$ and $r_2 > 0$, our rule is stricter than Equation 6, which yields more confidence for the generated pairwise preference. To implement the ranking MF model, we simply adopt the following parametrization.

$$\hat{r}_{u,i} = p_u^T q_i + b_u + b_i \quad (8)$$

Each user-item pair is still associated with value $\hat{r}_{u,i}$ as in the rating prediction problem. However, $\hat{r}_{u,i}$ does not represent the actual rating that user u assigns to item i. We do not try to make $\hat{r}_{u,i}$ to be close to actual rating $r_{u,i}$. Instead, we use the difference between $\hat{r}_{u,i}$ and $\hat{r}_{u,j}$ to generate $\delta_{u,i,j}$ to produce the following probabilistic model.

$$P(\delta_{u,i,j} = +1) = \frac{1}{1 + e^{-(\hat{r}_{u,i} - \hat{r}_{u,j})}} \quad (9)$$

The parameters are trained by maximizing the log-likelihood of the pairwise preference data. The loss function comes as follows.

$$Loss = \sum_{\delta_{u,i,j}=+1} \ln\left(1 + e^{-(\hat{r}_{u,i} - \hat{r}_{u,j})}\right) + \text{ regularization} \tag{10}$$

This loss function can be easily solved by using stochastic gradient descent. Though the number of pairwise preference is much larger than the number of ratings, we only need to sample $\delta_{u,i,j}$ based on the rating data. By tuning the parameter r_1 and r_2, we can make a balance between sampling confidence and sampling size. Through experiments, we observe that r_1 value of 100 and r_2 value of 70, works best in the CAMRa2011 dataset.

3.2 Matrix Factorization with Negative Sampling

Besides using learning to rank to settle this problem, we can also adopt the traditional MF approach. However, as we stated before, choice prediction is different from rate prediction. Since a user rarely rates the movies he or she doesn't like, only a small fraction of positive examples(which he/she likes) are available. Negative examples and missing values are mixed together and cannot be distinguished in the remaining part. We call this negative sample missing problem.

Several related strategies have been proposed to tackle this kind of problem. The two extreme strategies are all missing as negative and all missing as unknown. Rong Pan et al. [15] proposed a collaborative filtering task called One-Class Collaborative Filtering (OCCF) which only has positive examples. In their work, they proposed two approaches, the weighting based approach and the sampling based method, to balance the extend of treating missing values and negative examples. According to [15], the sampling based method can approximate the exact solution more efficiently for large scale sparse datasets.

The first track is expressed in explicit scores, which is not OCCF in strict sense. We still face the same problem of lacking negative examples. So we leverage the sampling based method to our problem. In the sampling procedure, we transfer all the ratings in the given training set into a new one. We then add the negative examples sampled from missing data by using user-oriented sampling [15] into the new training set. That is, the number of negative samples of a user will be in proportion to number of user's positive samples. Finally, we use the new training set to train our models in the following. As we will show in our experiments, negative sampling scheme is useful and makes it possible to use matrix factorization method for choice prediction problems.

4. INFORMATIVE MODELS

The contextual information plays an important role in recommendation performance. In our previous work of KDD-Cup 2011 track1 [4], we use over 10 different kinds of information to build a best single model. The work of [12] also shows that the temporal and social network information can lead to improvement in recommendation result. Since the the CAMRa2011 challenge dataset provides us with rich contextual information, we have to utilize as much as possible. However, it is inefficient to design a specific model for each kind of contextual information. In this section, we will describe our framework which can incorporate different kinds of contextual information easily to build informative models. Section 4.1 describes the feature-based matrix factorization framework, and the later subsections demonstrate how we use this framework to utilize different kinds of contextual information.

4.1 Feature-based Matrix Factorization

In order to utilize various kinds of information available, we adopt an abstract framework called feature-based matrix factorization [5]. The formalization of the model is shown in Equation 11.

$$\begin{aligned} y = f\Big(\mu + (\sum_j b_j^{(g)}\gamma_j + \sum_j b_j^{(u)}\alpha_j + \sum_j b_j^{(i)}\beta_j) \\ +(\sum_j p_j\alpha_j)^T(\sum_j q_j\beta_j)\Big) \end{aligned} \tag{11}$$

The input consists of three kinds of features $< \alpha, \beta, \gamma >$, where α, β, γ denote user, item, and global features respectively. Most existing MF models can fit into this abstract model. New models can also be developed simply by defining features. Similar idea has been proposed before by libFM [18]. Compared with their approach, our model divides the features into three types, while there is no distinction among features in libFM. This difference allows us to include global feature that does not need to be taken into factorization part, which is important for bias features such as user day bias, neighborhood based features. The division of features also gives us hints for model design. For global features, we shall consider what aspect may influence the overall rating. For user and item features, we shall consider how to describe user preference and item property better.

Basic MF model is a special case of Equation 11. For predicting user item pair $< u, i >$, we can define:

$$\gamma = \emptyset, \alpha_h = \begin{cases} 1 & h = u \\ 0 & h \neq u \end{cases}, \beta_h = \begin{cases} 1 & h = i \\ 0 & h \neq i \end{cases} \tag{12}$$

Pairwise-ranking model is also a special case of Equation 11, for $r_{ui} > r_{uj}$, we set $f(x) = \frac{1}{1+e^{-x}}, y = 1$, and define features as follows

$$\gamma = \emptyset, \alpha_h = \begin{cases} 1 & h = u \\ 0 & h \neq u \end{cases}, \beta_h = \begin{cases} 1 & h = i \\ -1 & h = j \\ 0 & h \neq i, j \end{cases} \tag{13}$$

To include more information, we only need to design new features. In the following subsections, we will display how to incorporate contextual information into our feature-based matrix factorization framework to form the informative models.

4.2 Integrate Neighborhood Information

Comparing the MF model (Equation 3) with the neighborhood model(Equation 1), it is not hard to discover that MF has much better generalization capability due to its success in representing abstract user-item relationship, while the neighborhood model is effective on detecting very localized relationship. As suggested in [2, 10], we can integrate the two models together to get a new model as follows

$$\begin{aligned} \hat{r}_{u,i} &= p_u^T q_i + b_u + b_i \\ &\quad + |N(u,i;k)|^{-\frac{1}{2}} \sum_{j \in R(u,i;k)} w_{i,j}(r_{u,j} - \overline{r_u}) \\ &\quad + |N(u,i;k)|^{-\frac{1}{2}} \sum_{j \in N(u,i;k)} c_{i,j} \end{aligned} \tag{14}$$

Here $N(u, i; k)$ stands for the k-nearest neighbors of item i which are rated by user u. The k-nearest neighborhood set can be pre-calculated using classical neighborhood methods. Then all the parameters will be learnt jointly.

Our task is to predict whether a user will rate a movie. This scenario is different from rate prediction problem in classical collaborative filtering. In our scenario, we find implicit information(associated with $c_{i,j}$) is more important than explicit information(associated with $w_{i,j}$). And we cannot guarantee the correctness of the estimation of $\overline{r}_u$ in explicit part due to the fact that a user may not rate the movies he does not like. So we propose a new model only using the implicit neighborhood information:

$$\hat{r}_{u,i} = b_u + b_i + p_u^T q_i + |N(u, i; k)|^{-\frac{1}{2}} \sum_{j \in N(u,i;k)} c_{i,j} \quad (15)$$

To describe this model in feature-based matrix factorization framework, we can simply redefine γ as Equation 16. Here *index* is a function that maps the possible pairs in k-nearest neighborhood set to consecutive integers.

$$\gamma_h = \begin{cases} \frac{1}{\sqrt{|N(u,i;k)|}} & h = index(i, j), j \in N(u, i; k) \\ 0 & otherwise \end{cases} \quad (16)$$

We also note that it is straightforward to build a pairwise ranking model by defining the global feature to be $\gamma(u, i) - \gamma(u, j)$, where $\gamma(u, i)$ is the global feature for $\hat{r}_{u,i}$.

4.3 Implicit Feedback

In general, implicit feedback information [10] indicates users' possible preferences on the items. For example, whether the user has seen some news about the movie, whether he/she has related chatting with his/her friends, whether he/she is a fan of the hero in the movie, or even whether he/she has watched the movie or has bought the DVD of the movie. In our work, the implicit information available is whether a user rated a movie or not. To incorporate such implicit feedback information, we can adjust our model as follows:

$$\hat{r}_{u,i} = b_u + b_i + q_i^T \left(p_u + |R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j \right) \quad (17)$$

Where $R(u)$ is the set of items rated by user u. The sum $|R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} y_j$ denotes the implicit preferences of user u. To describe implicit feedback in feature-based matrix factorization framework, we can add the information to user feature(m means number of user)

$$\alpha_h = \begin{cases} 1 & h = u \\ \frac{1}{\sqrt{|R(u)|}} & h = j + m, j \in R(u) \\ 0 & otherwise \end{cases} \quad (18)$$

4.4 User-Household Hierarchy

In CAMRa2011 dataset, we can get household information of users. Each house has several related user members. This hierarchical information can be useful in modeling, because house members are related with each other in real life. Inspired by this idea, we build a house-aware model taking the household information into account:

$$\hat{r}_{u,i} = q_i^T (p_u + p_{house(u)}) + b_u + b_i + b_{house(u)} \quad (19)$$

$house(u)$ maps users into corresponding house. $b_{house(u)}$ and $p_{house(u)}$ are house-aware parameters that model the bias and preference of the house. To formalize the idea in feature-based matrix factorization framework, we only need to add house information to user feature:

$$\alpha_h = \begin{cases} 1 & h = u \\ 1 & h = house(u) + m \\ 0 & otherwise \end{cases} \quad (20)$$

5. HOUSEHOLD RECOMMENDATION

The goal of the first track of CAMRa2011 challenge is to generate recommendations for households instead of individual users. We will describe two approaches for household recommendation in the following. In both approaches, we use methods described above to generate ranking values for users or households.

5.1 User Oriented Approach

The models described in previous sections can produce ranking values of items for each user. To recommend items for each household, we need to generate ranking values of items for households. To achieve this goal, we first get the predicted users' preferences for items and then combine the preferences of household members to get the household preference. We denote this approach user oriented approach. Aggregation methods become a crucial part of this approach. Strategies like Average and Least Misery as well as Random have existed already [15]. And in our experiments, we adopt the following rule to combine users preference for a household.

$$\hat{r}_{h,i} = \sum_{u \in H(h)} w_u \cdot \hat{r}_{u,i} \quad (21)$$

$H(h)$ is the set of house members in h. $\hat{r}_{h,i}$ denotes the predicted ranking value that household h will assign to item i, which is generated by the weighted average of its members' ratings. The user weight parameter w_u can be set according to different standards, for example, the authority of the users in the household. A natural way is to simply set all w_u to 1, to let every house member donate the same contribution to the final result, and we adopt it in our experiments.

5.2 Household Oriented Approach

The user oriented approach first learns the preference of household members, and then combines them together. We can also learn preference of households directly. We call this approach house oriented approach. In this subsection, we will describe the household oriented approach based on basic matrix factorization or ranking matrix factorization.

A direct idea to learn preference of households is to treat households as users, and use matrix factorization models described in previous sections to train a ranking value predictor for house item pairs. However, the given training set only contains user ratings, thus, we need to generate household ratings from existing data to get a household training set. In basic MF, there are several algorithms [1] which can be used to generate household preference according to user ratings. The simplest way is average aggregation, in which the rating household h assigns to item i is equal to the average of the ratings of its members,

$$r_{h,i} = \frac{\sum_{u \in H(h)} r_{u,i}}{|H(h)|} \quad (22)$$

In pairwise ranking matrix factorization models, we generate household preference pairs from household members' preference. If one member of a household prefers item i to item j, we infer the household prefers item i to item j. Training cases with conflict are ignored.

After adding the generated household ratings or preference pairs into the training set, we can train household oriented models. There are two methods of training, to train it separately or together with user predictor. The latter tries to optimize the convex combination of loss functions of user predictor and household predictor. Because the item parameters q_i and b_i are shared in two predictors, joint-training can potentially make better use of the information available. We choose the joint-training in our approach.

6. EXPERIMENTS AND EVALUATION

In this section, we present the experiments to evaluate the proposed models. Through the experiments, we want to answer the following questions:

- Does the ranking matrix factorization model outperform the basic matrix factorization model?
- Can the negative examples sampling scheme improve recommendation performance?
- Are the proposed informative models helpful in improving recommendation performance for the CAMRa-2011 challenge?
- What kind of household recommendation approaches is better?

6.1 Evaluation Metrics

The evaluation schemes of CAMRa2011 consist of four well-known metrics, MAP (mean average precision), P@5, P@10 (precision at 5 or 10) and AUC (area under curve). Since CAMRa2011 is household-targeted recommendation, there are two different means of evaluation, denoted as user-based metric and household-based metric respectively, based on whether the recommendation is user-based or household-based[3]. The difference between the two recommendation approaches lies whether the list of recommended items is recommended to individual users or to a household. As we have discussed both approaches in our paper(user-based method in Section 3 and Section 4, household-based method in Section 5). We adopt both metrics in the following sections. For the purpose of clarity, we will explicitly state which metric we use when we present results.

In both evaluation means, the four metrics are all firstly calculated for each user, then average per user in a household, and thirdly average per household to get the final result. Besides, we also calculate the average ranking for the ground truth movies, which is averaged per movie per household. We denote it as rank when we present our results.

6.2 Performance Comparison

In order to answer the proposed questions, we compare the results of different approaches in this section. We build all of our models under feature-based matrix factorization framework using *SVDFeature*[4]. The approaches in our experiments are denoted as follows:

Table 1: Comparison of recommendation performance (user-based metric)

Models	MAP	AUC	P@5	P@10
BMF	0.1390	0.8374	0.1344	0.1051
BMF-3N	0.2268	0.9926	0.2039	0.1680
BMF-3N-HIR	0.2315	0.9910	0.2124	0.1718
BMF-3N-IMFB	0.2383	0.9940	0.2150	0.1727
BMF-3N-I100NN	0.2614	0.9922	0.2402	0.1968
BMF-3N-ALL	0.2639	0.9924	0.2435	0.1970
RMF-S1	0.2053	0.9931	0.1931	0.1608
RMF-S2	0.2275	0.9939	0.2065	0.1741
RMF-S2-HIR	0.2387	0.9943	0.2167	0.1814
RMF-S2-IMFB	0.2477	0.9943	0.2322	0.1893
RMF-S2-I100NN	0.2847	0.9936	0.2550	0.2021
RMF-S2-ALL	0.3096	0.9956	0.2872	0.2190

- BMF: This model represents the basic matrix factorization approach which is equivalent to Equation 3. We set the parameter λ_1 to 0.004 and λ_2 to 0, which is optimal on the evaluation set. The dimensionality of user and item factors is 64 here for efficiency. The same parameters are used in the ranking matrix factorization model.
- RMF: This model represents the ranking matrix factorization approach, stated in Section 3.1. We try the two sampling schemes in our experiments, denoting RMF-S1(as in Equation 6) and RMF-S2(as in Equation 7, $r_1 = 100, r_2 = 70$) respectively.
- BMF-3N: This approach represents the BMF model which add three fold negative examples to the training set as stated in Section 3.2.
- BMF-3N-I100NN(HIR, IMFB, ALL): These approaches represent the informative models which integrate the item neighborhood information, user household hierarchy and user implicit feedback into the BMF-3N model respectively, as stated in Section 4. BMF-3N-ALL denotes the model that integrates all the useful information into the BMF-3N model.
- RMF-S2-I100NN(HIR, IMFB, ALL): These approaches represent the informative models which integrate the item neighborhood information, user household hierarchy and user implicit feedback into the RMF-S2 model respectively, as stated in Section 4. RMF-S2-ALL denotes the model that integrates all the useful information into the basic RMF-S2 model.

The approaches above are all user-based recommendation so we evaluate them in user-based metric. The results are shown in Table 1. We can see that the proposed models and data sampling schemes all outperform the basic matrix factorization model. RMF-S2 achieves about 60% improvement over BMF with respect to P@10 and MAP, which indicates that the ranking matrix factorization model indeed outperforms the basic matrix factorization model. RMF-S2 also achieves about 8% over RMF-S1 with respect to MAP, P@5 and P@10, which demonstrates the effectiveness of our method over the old one.

To see the effectiveness of negative sampling, we observe that BMF-3N achieves an improvement around 60% over

[3] http://2011.camrachallenge.com/evaluation/

[4] project page: http://apex.sjtu.edu.cn/apex_wiki/svdfeature

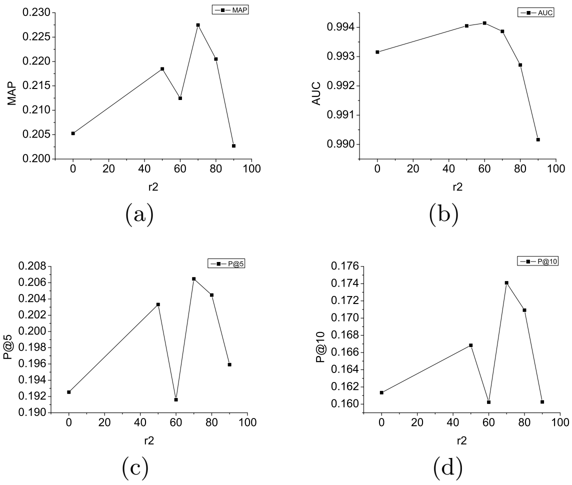

Figure 1: The impact of parameter r_2 in the performance of RMF

BMF, which indicates that adding negative examples into the training set can have significant improvement over basic matrix factorization. Besides, the size of negative examples also affects the performance. In our experiments, a larger sampling size leads to a better performance with respect to the metrics of MAP, P@5 and P@10. The effect of sampling size will be discussed in Section 6.3.

The results in Table 1 also show that the contextual information of user household hierarchy, user implicit feedback and item neighborhood contribute greatly to the performance. By incorporating neighborhood information into RMF-S2 model, an improvement of around 25% in MAP and P@5 is achieved. User implicit feedback contributes about 10% in the performance. User and household hierarchy information can make an improvement of about 3%, which is less significant than those of neighborhood information and implicit feedback information. Integrating all the useful information into RMF-S2 model is very convenient under the feature-based matrix factorization framework, and we can get a final informative RMF model, denoted RMF-S2-ALL, with an overall improvement of 30% over RMF-S2. Integrating contextual information into BMF-3N model also achieves improvement as well. The experiment results show that our proposed informative models based on feature-based matrix factorization are indeed effective.

6.3 Impact of Sampling Parameters

The parameters of r_1 and r_2 in ranking matrix factorization model will influence the performance. In our experiments, parameter r_1 is fixed to 100, which is proved to be optimal by our experiments, and parameter r_2 is tuned to get the best performance. Figure 1 illustrates the impact of r_2. It demonstrates that for P@5, P@10 and MAP the best r_2 is nearly 70, while AUC is best when r_2 is set to 60. Since metrics related to precision are more appropriate for our recommendation task, we prefer P@5, MAP and P@10 to AUC. Thus, we choose optimal r_2 to be 70, where all the three metrics give good results. From Figure 1 we can also see that the performance is much lower when the value of r_2 is set to 0, which is equivalent to Equation 6. This again proves the effectiveness of our new pairwise preference sampling approach. Figure 1 demonstrates that the r_2 value of 70 makes a best balance between sampling confidence and sampling size. A higher value of r_2 decreases the number of pairwise preference while a lower value of r_2 decreases the confidence of pairwise preference, thus both lead to a poorer performance.

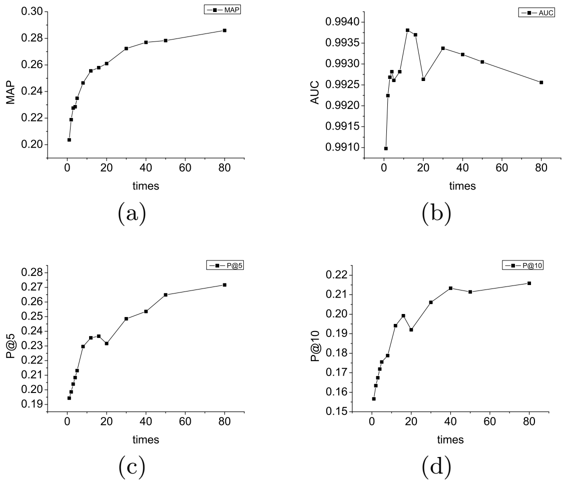

Figure 2: The impact of negative sampling size in the performance of BMF

Table 2: Performance of Household Recommendation Approaches (household-based metric)

Models	MAP	AUC	P@5	P@10
RMF-S2	0.2242	0.9941	0.2015	0.1696
RMF-S2-H	0.2242	0.9940	0.2080	0.1742

The size of negative examples will also affect the recommendation performance in matrix factorization with negative sampling. We compare the performance of different sampling size in Figure 2. It shows that a larger sampling size leads to better performance. However, larger negative sampling size will result in longer training time. So there is a tradeoff between training complexity and performance.

6.4 Comparison of Household Recommendation Approaches

We have proposed two approaches for household recommendation in Section 5. We compare the performance of the proposed household oriented approach with the user oriented approach in Table 2. RMF-S2 represents the user oriented RMF model and RMF-S2-H represents the household oriented RMF model. The results are evaluated in household-based metric. We can find that the two methods give similar results. This result shows that user oriented approach is enough to do current household recommendation task. However, we believe that the idea of learning household preference directly is interesting and may have its usage in other scenarios.

6.5 Best Models

This section summarizes the performance of our best models. We adopt both user-based metric and household-based metric to evaluate our results. The result of user-based met-

Table 3: Best Models for User-Based Recommendation (user-based metric)

Models	MAP	AUC	P@5	P@10	rank
BMF-80N-ALL	0.2951	0.9916	0.2696	0.2129	205
RMF-S2-ALL	0.3096	0.9956	0.2872	0.2190	111

Table 4: Best Models for Household-Based Recommendation (household-based metric)

Models	MAP	AUC	P@5	P@10	rank
BMF-80N-ALL	0.3056	0.9916	0.2843	0.2167	204
RMF-S2-ALL	0.3186	0.9958	0.2970	0.2210	104

ric is shown in Table 3. Since the negative sampling size has a significant impact on the performance of the BMF model, we adopt a sampling size as large as 80 times of the original training data in our final BMF model. There are many kinds of contextual information available in the first track dataset, and Table 1 shows that our proposed informative models are indeed effective, so we adopt the informative models which integrate all the useful information as our best models. The final BMF model with all the contextual information is denoted as BMF-80N-ALL. RMF-S2-ALL represents the best RMF model which integrates all the useful information into RMF-S2, and it has been described in Section 6.2. Comparing with the results of BMF model in Table 1, we can find that both the best matrix factorization model with negative sampling and ranking matrix factorization model perform much better than basic matrix factorization, with an improvement of around 100% in the metrics of MAP, P@5 and P@10.

For household-based recommendation, we adopt user oriented approach to generate household recommendation. The results are evaluated using household-based metric in Table 4. Compared with the simple RMF methods for household recommendation in Table 2, our best models also work better due to the effectiveness of informative models. This result shows that our proposed method works well for both user-based recommendation and household-based recommendation tasks.

7. RELATED WORK

Neighborhood-based approaches [20, 22] and model-based approaches [15, 14, 18] are two major types of collaborative filtering. The former recommend items from the perspective of similarity of its neighbors, while the latter leverage more on latent factors using the matrix factorization method to solve problems. Based on these well learnt and widely utilized approaches, ranking oriented models and contextual information should be carefully considered for the first track of CAMRa2011.

Ranking oriented collaborative filtering takes the item ranking list for each user as a goal. Just like the pairwise models of learning to rank [3, 9], Pessiot et al.[16] proposed a pairwise preference error minimization framework to optimize the item ranking. Rendle et al. [19] proposed a Bayesian probabilistic ranking model. In the work of [13], user-user similarity is firstly measured by the correlation between their rankings. Then the authors proposed their EigenRank algorithm based on user-user similarity. Moreover, Liu et al. [14] proposed a latent class model pLPA to capture user preferences and then an item ranking prediction approach was designed based on pLPA.

Contextual information, such as temporal dynamics [11], implicit/explicit feedback [7, 10], and users' social relationship, etc.., has been verified to be very important for predicting users' current preference and making better recommendation. The challenge on context-aware movie recommendation (CAMRa) aims at utilizing more contextual information to improve the performance of recommendation. For example, in the CAMRa2010, more attention has been paid on temporal dynamics [12], movie mood [23, 24], and user social networks [6]. It motivates us to excavate more contextual information, such as the hierarchical information contained, to further improve the performance of our experiments.

It is obvious that CAMRa2011 is a group recommendation(GR) problem. In general, GR tries to aggregate users' recommendation lists into a single group [1]. Alternatively, some authors create group profiles to replace the original user profiles to do recommendation [8]. Recently, much attention has been paid to GR, since it has popular applications in many domains such as web/news pages [17], TV programs [25]. However, it has not been well learnt so far due to difficulties in its evaluation [21] and also unappropriate dataset.

8. CONCLUSIONS

In this paper, we mainly study the informative feature-based model as well as household recommendation particularity to handle the first track of CAMRa2011 challenge. We adjust the learning to rank and the data sampling approaches to settle the choice prediction problem. Those models are further improved by adding different kinds of contextual information using feature-based matrix factorization framework. Moreover, we adopt both user oriented approach and household oriented approach to generate recommendation for households based on our models. The experiment results show that our proposed approaches give nice performance on the first track of CAMRa2011 over other baselines.

However, accuracy is only one indicator of recommendation characteristics, which may lead to the monotony of recommendation. There are many other characteristics to evaluate the quality of recommendation. In the future, we will work on other aspects of recommendation, such as the explanation, the novelty and the diversity, etc.. Besides those evaluations, more work concerning group recommendation also need to be advocated.

9. ACKNOWLEDGEMENT

The team is supported by grants from NSFC-RGC joint research project 60931160445.

10. REFERENCES

[1] L. Baltrunas, T. Makcinskas, and F. Ricci. Group recommendations with rank aggregation and collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, 2010.

[2] R. M. Bell and Y. Koren. Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In *Proceedings of the 2007 Seventh IEEE International Conference on Data*

Mining, pages 43–52, Washington, DC, USA, 2007. IEEE Computer Society.

[3] Y. Cao, J. Xu, T. Liu, H. Li, Y. Huang, and H. Hon. Adapting ranking svm to document retrieval. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 186–193. ACM, 2006.

[4] T. Chen, Z. Zheng, Q. Lu, X. Jiang, Y. Chen, W. Zhang, K. Chen, Y. Yu, N. Liu, B. Cao, L. He, and Q. Yang. Informative ensemble of multi-resolution dynamic factorization models. In *KDD-Cup Workshop*, 2011.

[5] T. Chen, Z. Zheng, Q. Lu, W. Zhang, and Y. Yu. Feature-based matrix factorization. Technical Report APEX-TR-2011-07-11, Apex Data & Knowledge Management Lab, Shanghai Jiao Tong University, July 2011.

[6] F. Díez, J. Chavarriaga, P. Campos, and A. Bellogín. Movie recommendations based in explicit and implicit features extracted from the filmtipset dataset. In *Proceedings of the Workshop on Context-Aware Movie Recommendation*, pages 45–52. ACM, 2010.

[7] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*, pages 263–272. IEEE, 2008.

[8] A. Jameson and B. Smyth. The adaptive web. chapter Recommendation to groups, pages 596–627. Springer-Verlag, Berlin, Heidelberg, 2007.

[9] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. ACM, 2002.

[10] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 426–434, New York, NY, USA, 2008. ACM.

[11] Y. Koren. Collaborative filtering with temporal dynamics. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 447–456, 2009.

[12] N. N. Liu, B. Cao, M. Zhao, and Q. Yang. Adapting neighborhood and matrix factorization models for context aware recommendation. In *Proceedings of the Workshop on Context-Aware Movie Recommendation*, CAMRa '10, pages 7–13, New York, NY, USA, 2010. ACM.

[13] N. N. Liu and Q. Yang. Eigenrank: a ranking-oriented approach to collaborative filtering. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 83–90, New York, NY, USA, 2008. ACM.

[14] N. N. Liu, M. Zhao, and Q. Yang. Probabilistic latent preference analysis for collaborative filtering. In *Proceeding of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 759–766, New York, NY, USA, 2009. ACM.

[15] R. Pan, Y. Zhou, B. Cao, N. N. Liu, R. Lukose, M. Scholz, and Q. Yang. One-class collaborative filtering. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pages 502–511, Washington, DC, USA, 2008. IEEE Computer Society.

[16] J. Pessiot, T. Truong, N. Usunier, M. Amini, and P. Gallinari. Learning to rank for collaborative filtering. In *9th International Conference on Enterprise Information Systems*. Citeseer, 2007.

[17] S. Pizzutilo, B. De Carolis, G. Cozzolongo, and F. Ambruoso. Group modeling in a public space: methods, techniques, experiences. In *Proceedings of the 5th WSEAS International Conference on Applied Informatics and Communications*, 2005.

[18] S. Rendle. Factorization machines. In *Proceedings of the 2010 IEEE International Conference on Data Mining*, ICDM '10, 2010.

[19] S. Rendle, C. Freudenthaler, Z. Gantner, and S.-T. Lars. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, 2009.

[20] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, WWW '01, pages 285–295, New York, NY, USA, 2001. ACM.

[21] C. Senot, D. Kostadinov, M. Bouzid, J. Picault, and A. Aghasaryan. Evaluation of group profiling strategies. In *IJCAI*, pages 2728–2733, 2011.

[22] Y. Shi, M. Larson, and A. Hanjalic. Exploiting user similarity based on rated-item pools for improved user-based collaborative filtering. In *Proceedings of the third ACM conference on Recommender systems*, RecSys '09, pages 125–132, New York, NY, USA, 2009. ACM.

[23] Y. Shi, M. Larson, and A. Hanjalic. Mining mood-specific movie similarity with matrix factorization for context-aware recommendation. In *Proceedings of the Workshop on Context-Aware Movie Recommendation*, CAMRa '10, pages 34–40, New York, NY, USA, 2010. ACM.

[24] L. Wang, X. Meng, Y. Zhang, and Y. Shi. New approaches to mood-based hybrid collaborative filtering. In *Proceedings of the Workshop on Context-Aware Movie Recommendation*, pages 28–33. ACM, 2010.

[25] Z. Yu, X. Zhou, Y. Hao, and J. Gu. Tv program recommendation for multiple viewers based on user profile merging. *User Modeling and User-adapted Interaction*, 2006.