A Design of Recommendation Based on Flexible Mixture Model Considering Purchasing Interest and Post-Purchase Satisfaction

Takeshi SUZUKI \(^{1}\), Gendo KUMOI \(^{2}\), Kenta MIKAWA \(^{1}\) and Masayuki GOTO \(^{3}\)

Abstract: The recommender system is an effective Web marketing tool that have been used especially on electric commerce sites in recent years. The recommender system provides each user with a list of new recommended items that are predicted to be preferred by the user. Collaborative filtering is one of the most representative and powerful methods to predict user preference in the recommender system. Collaborative filtering measures the similarity of preference between users and uses it to decide items to be recommended. Based on previous research on this method, user preference is considered to have two aspects: Purchasing interest for items and post-purchase satisfaction with items. However, the conventional methods do not consider the two different preferences at the same time. This paper suggests taking these two preferences into account and proposes a new method that allows users to choose the balance between them. The proposed method is evaluated through simulation experiments with MovieLens data. It demonstrates the effectiveness of our proposal in precision and average rating compared with a previous method.

Key words: recommender systems, collaborative filtering, flexible mixture model, probabilistic models, latent class model

1 INTRODUCTION

In recent years, the number of electric commerce (EC) sites such as Amazon.com \([1]\) has been increasing due to the development of information technology. On EC sites, web marketing tools have been developed based on IT systems with customer databases. Especially, the recommender systems have become a strong marketing tool \([2]-[5]\). The recommender system provides an active user (for whom the system makes recommendations) with a list of items which are predicted to be preferred by the user. Usually, the recommender system uses databases which are provided with many kinds of information such as product details, purchase records, click records, item ratings and so on, in order to predict user preference through various methods.

Predicting methods for recommender systems are usually classified into two categories: content-based filtering \([4]\) and collaborative filtering \([6]-[8]\). Content-based filtering utilizes product details (such as the cast of a movie) and/or user profile (such as age, sex or occupation) and relies only on information about an active user. Collaborative filtering utilizes purchase records, click records or item ratings of an active user and other users. The ratings can be explicit indications on a 1-5 scale. Both content-based filtering and collaborative filtering have limitations: problems of item domain for content-based filtering and cold start problems for collaborative filtering \([9]\). However, Collaborative Filtering is generally more accurate than Content-based Filtering in predicting user preference and Collaborative Filtering has been widely used on many EC sites. We thus focus on collaborative filtering in this paper.

Collaborative filtering attempts to identify a group of users with similar preference to an active user and assumes that the user would like the items which the group liked. The algorithm aggregates items from these similar customers, outputs
the degree of preference which the user is considered to have for each item, and eliminates items the user has already purchased or rated. The recommender system with collaborative filtering thus provides the user with a list of items ranked according to the predicted degree of preference.

The earliest collaborative filtering techniques utilize purchase records of whether items are purchased or not and predicts the degree of purchasing interest for unpurchased items for an active user. In other words, collaborative filtering with purchase records predicts purchase probability of the active user. In previous studies of collaborative filtering with purchase records, various models have been proposed: latent class model [10], [11], clustering model [12], simple Bayesian classifier [13], improved naive Bayes method [14] and so on [15].

On the other hand, there are now many EC sites on which users can give ratings of post-purchase satisfaction with their purchased and used items. Most of the recent research of collaborative filtering utilizes item ratings to predict ratings of unrated items. Therefore, collaborative filtering with item ratings predicts the degree of post-purchase satisfaction. Several models are also proposed in this field: Pearson correlation coefficient [16], [17], vector similarity [18], Gaussian latent semantic model [19] and so on. Among the models in this research, the flexible mixture model (FMM) [20], [21] is one of the most accurate models in predicting ratings. The FMM is the simplest model to express the latent classes for both users and items. In the field of marketing science, it is usual to assume the latent classes for both product items and users [22]. The FMM is the most basic and cardinal model to express these two latent classes and several extensions toward different directions can be considered.

From this background of collaborative filtering, we thus consider user preference to have two aspects: the purchasing interest (buying intention) for items and post-purchase satisfaction about the items after purchase. To our best knowledge, however, most of the existing approaches lack the view of the balance between the two preferences and only consider either of them. The purchasing interest is very important in marketing science because it will be a cause of buying behavior of consumers. If a user doesn’t have purchasing interest in a product item, he or she doesn’t buy it. In order to increase the sales amount, it must be important to enhance the purchasing interest of users. The post-purchase satisfaction equates to users’ rating. Since it is the objective of recommender systems to recommend items with high expected rating, there is no room for doubting the importance of the post-purchase satisfaction in recommender systems. Moreover, both purchasing frequency and post-purchase satisfaction can be observed in the purchasing histories on a database. Therefore, it is possible to acquire the probabilities of parameters with respect to both purchasing interest and post-purchase satisfaction.

Therefore, we propose a new method of collaborative filtering considering both predicted purchase probability and predicted rating based on the FMM. To consider the balance between them, we introduce a weight parameter with which users or site managers can control this trade-off. This proposed method is evaluated through simulation experiments with MovieLens data [23] and demonstrates effectiveness of precision and average ratings compared with a method considering only predicted rating.

2 PREVIOUS RESEARCH

2.1 Flexible Mixture Model

The FMM [20], [21] is a probabilistic latent class model which was proposed to predict ratings of unpurchased items. We briefly discuss the FMM here.

We consider the following formal setting: Given are a set of items $X = \{x_i : 1 \leq i \leq I\}$, a set of users $Y = \{y_j : 1 \leq j \leq J\}$ and a set of ratings $(1 \leq r \leq R)$. A set of $(x_i, y_j, r)$ represents that a user $y_j$ gives a rating $r$ on an item $x_i$. In this model, we only use this three dimensional information. This model introduces two latent classes: a set of latent classes for items $Z = \{z_k : 1 \leq k \leq K\}$ and a set of latent classes for users $W = \{w_l : 1 \leq l \leq L\}$. This model is assumes that a group of similar users is likely to give a same rating on a group of similar items and has the feature that users and items belong not to a single class but to multiple classes.

The probability for a set of $(x_i, y_j, r)$ is calculated as Eq.(1) and the corresponding graphical model for Eq.(1) is shown in Fig.1. This model assumes
that a rating \( r \) is given on condition that the latent classes of users and items are given.

\[
P(x_i, y_j, r) = \sum_{k,l} P(z_k) P(w_l) P(x_i | z_k) P(y_j | w_l) P(r | z_k, w_l).
\]

(1)

Fig. 1  A graphical model for FMM

The latent variable \( z_k \) indicates the class membership for an item \( x_i \) and \( P(z_k) \) is a multinomial distribution on the item classes \( Z \). The latent variable \( w_l \) indicates the class membership for a user \( y_j \) and \( P(w_l) \) is also a multinomial distribution on the user classes \( W \). \( P(x_i | z_k) \) is a conditional probability of item \( x_i \) given a specific user class \( z_k \) and is also a multinomial distribution on the item set \( X \). \( P(y_j | w_l) \) is a multinomial distribution on the user set \( Y \) representing the conditional probability of a user \( y_j \) given a specific user class \( w_l \). \( P(r | z_k, w_l) \) is a multinomial distribution for a rating \( r \) given a specific item class \( z_k \) and a specific user class \( w_l \).

The latent classes cannot be observed in the database but the parameters of these distributions can be estimated using the EM algorithm [26] to maximize the log-likelihood.

The FMM is the simplest and important model to express the latent classes for both users and items. Jin et al. extended the FMM by introducing two latent variables that account for rating patterns and intrinsic preference of users and called this model the decoupled model (DM) [20],[21]. From the viewpoint of prediction accuracy, the DM approach is interesting because Jin et al. showed the result of a numerical experiment where the prediction accuracy of the DM is superior to that of the FMM [21]. However, Jin et al. also proposed other models expressing the user preferences [24], [25]. For example, Jin and Si also proposed a method expressing the user preferences by normalizing user ratings and showed that the FMM approach with normalized ratings shows a similar prediction accuracy to that of the DM approach. Regarding how to model the user preferences, there is room for improvement and the DM approach may be improved in future. The FMM will be the most basic and cardinal model expressing the two latent classes of users and items and several directions of extension of the FMM can be expected in future studies.

### 2.2 Training Procedure

The EM algorithm [26] is a well-known algorithm to calculate a maximum likelihood estimator using an iterative procedure. In order to estimate the parameters of the FMM model, the EM algorithm can be applied. The EM algorithm consists of two steps: The expectation step (E-step) and the maximization (M-step). In the E-step, the joint posterior probabilities of the latent variables are calculated. In the M-step, the model parameters are updated by maximizing the expected likelihood function using the given posterior probabilities calculated in the E-step.

Let \( (x_{(1)}, y_{(1)}, r_{(1)}), \ldots, (x_{(n)}, y_{(n)}, r_{(n)}) \) be the ratings in the training dataset. Here, \( x_{(t)} \in X \) and \( y_{(t)} \in Y \) for \( t = 1, 2, \ldots, n \). Then, the joint probability of \( (x_{(t)}, y_{(t)}, r_{(t)}) \) for FMM can be written as

\[
P(x_{(t)}, y_{(t)}, r_{(t)}) = \sum_{k,l} P(z_k) P(w_l) P(x_{(t)} | z_k) P(y_{(t)} | w_l) P(r_{(t)} | z_k, w_l).
\]

(2)

In the E-step, the joint posterior probabilities are given by

\[
P(z_k, w_l | x_{(t)}, y_{(t)}, r_{(t)}) = \frac{P(z_k) P(w_l) P(x_{(t)} | z_k) P(y_{(t)} | w_l) P(r_{(t)} | z_k, w_l)}{\sum_{k,l} P(z_k) P(w_l) P(x_{(t)} | z_k) P(y_{(t)} | w_l) P(r_{(t)} | z_k, w_l)}.
\]

(3)

In the M-step, the parameters of FMM can be updated by the following equations:

\[
P(z_k) = \frac{\sum_{t} P(z_k, w_l | x_{(t)}, y_{(t)}, r_{(t)})}{n},
\]

(4)
\begin{align}
P(w_l) &= \frac{\sum_t \sum_k P(z_k, w_l|x_t, y_t, r_t)\sum}{nP(z_k)}, \quad (5) \\
P(x|z_k) &= \frac{\sum_t: x_t = x \sum_k P(z_k, w_t|x_t, y_t, r_t)\sum}{nP(z_k)}, \quad (6) \\
P(x|w_l) &= \frac{\sum_t: y_t = y \sum_t P(z_k, w_t|x_t, y_t, r_t)\sum}{nP(z_k)} , \quad (7) \\
P(r|z_k, w_l) &= \frac{\sum_t: r_t = r \sum_t P(z_k, w_t|x_t, y_t, r_t)\sum}{\sum_t P(z_k, w_t|x_t, y_t, r_t)} , \quad (8)
\end{align}

Regarding Eqs.(4) and (5), the values $P(z_k)$ and $P(w_l)$ for all $z_k$ and $w_l$ in the next step are updated using the values in the previous step.

These parameters of the FMM converge after the enough iterations of the E-step and M-step. The EM algorithm converges to the global maximum or a local maxima. The estimates of the model parameters are denoted by $\hat{P}(z_k)$, $\hat{P}(w_l)$, $\hat{P}(x|z_k)$, $\hat{P}(y|w_l)$ and $\hat{P}(r|z_k, w_l)$.

The joint posterior probability in training procedure may have no global maximum, local maxima or stationary points, which in general happens with high probability in maximum likelihood and maximum a posteriori estimations. Only the convergence to local maxima is certified by applying the EM algorithm. As shown in [21], in order to estimate better parameters, some smoothing techniques such as the, Annealed EM algorithm and the maximum a posteriori approach, can be applied to estimate the parameters of the FMM. Though the prediction performance can be improved by these smoothing approaches, it is necessary to regulate hyper parameters using an empirical method and we cannot decide the best one for all situations of problem settings in the estimation of the FMM. Through our numerical experiments of the FMM, we concluded that it is not necessary to apply these smoothing approaches to parameter estimation because it is not essential for the purpose of our research. The purpose of our research is to consider both users’ purchasing interest and post-purchase satisfaction. Since the proposed method can be applied to the case that a smoothing method is used to the estimation of FMM parameters, we apply the basic EM-algorithm given by Eqs.(4)-(8).

### 2.3 Prediction of Ratings Procedure

In order to predict ratings of unrated items, we calculate the estimator of the joint probability $\hat{P}(x_i, y_j, r)$ from Eq.(1) utilizing the parameters estimated by the EM algorithm.

\[
\hat{P}(x_i, y_j, r) = \sum_{k,l} \hat{P}(z_k)\hat{P}(w_l)\hat{P}(x_i|z_k)\hat{P}(y_j|w_l)\hat{P}(r|z_k, w_l).
\]

(9)

Supposing a predicted rating on an item $x_i$ by a user $y_j$ as $\hat{r}(x_i, y_j)$, we can calculate the predicted rating as in Eq.(10).

\[
\hat{r}(x_i, y_j) = \sum_r \hat{P}(x_i, y_j, r) \sum_r \hat{P}(x_i, y_j, r).
\]

(10)

This is the predicted value of the rating of the user $y_j$ for the item $x_i$. Usually, the items $y_j$ with high predicted rating $\hat{r}(x_i, y_j)$ are recommended to user $x_i$. Therefore, the ranking of items on $X$ by $\hat{r}(x_i, y_j)$ is necessary for recommendation to user $y_j$.

On the other hand, the high rating does not necessarily mean high purchase probability. The rating represents the degree of satisfaction. From the viewpoint of EC site managers, the customer satisfaction is of course important. However, increasing sales amount is also a purpose of introducing the recommender system. It is necessary to consider the balance of high rating and high purchase probability.

### 3 PROPOSED METHOD

#### 3.1 Problems of Existing Approaches

When items are recommended only from the viewpoint of predicted purchase probability (the degree of purchasing interest), there is a possibility that the predicted ratings of the items may be low because item ratings are not considered at all. Problems could arise, in this case, if some users would
like to be recommended items with high predicted ratings (the degree of post-purchase satisfaction).

Figure 2 shows the image when the items with high predicted purchase probability are recommended. The points in this scatter diagram mean items $x_1, x_2, \ldots, x_I$ with $\hat{r}(x_i, y_j)$ and $\hat{P}(x_i|y_j)$ for a specific user $y_j$. The horizontal axis is the predicted rating $\hat{r}(x_i, y_j)$ and the vertical axis is the predicted purchase probability $\hat{P}(x_i|y_j)$. The ranking of descending predicted purchase probability can be listed up by considering only the predicted purchase probability $\hat{P}(x_i|y_j)$. The items with high predicted purchase probability $\hat{P}(x_i|y_j)$ can be found by the descending of the parallel line to the horizontal axis.

Figure 3 shows the image when the items with high predicted ratings are recommended. The ranking of descending predicted rating can be listed up by considering only the predicted rating $\hat{r}(x_i, y_j)$. The items with high predicted purchase probability $\hat{P}(x_i|y_j)$ can be found by moving a parallel line to the vertical axis from the right to the left.

Therefore, when recommending items, it is better to consider both purchase probability and predicted rating. We calculate purchase probability based on the FMM and propose a new method where users can choose the balance between predicted purchase probability and predicted rating.

Figure 4 shows the image of the recommendation when the items with both high purchase probability and high predicted rating are recommended.

Such items can be acquired by lowering a line that is not parallel to the horizontal and vertical axis. By this idea, the items with high purchase probability and high predicted rating can be recommended. This strategy is sometimes useful for EC site managers.

3.2 Making Recommendations

First, we calculate a predicted purchase probability $\hat{P}(x_i|y_j)$ that is a probability a user $y_j$ purchases an item $x_i$. Considering $\hat{P}(x_i|y_j)$ in addition to $\hat{r}(x_i, y_j)$ from Eq.(10), we propose a new criterion $N(x_i, y_j)$ and make a list of recommending items based on the value of $N(x_i, y_j)$. Here, $\hat{P}(x_i|y_j)$ and $N(x_i, y_j)$ are calculated as the following equations.
\[ \hat{P}(x_i|y_j) = \frac{\sum_r \hat{P}(x_i, y_j, r) P(x_i, y_j, r)}{\sum_{i,r} \hat{P}(x_i, y_j, r)} \] (11)

\[ N(x_i, y_j) = \alpha \frac{\hat{r}(x_i, y_j)}{R} + (1 - \alpha) \frac{\hat{P}(x_i|y_j)}{\max P(x_i|y_j)} \] (12)

The value \(\alpha(0 \leq \alpha \leq 1)\) is a weight parameter for users to decide the importance of two terms: purchase probability and predicted rating. In Eq.(12), the two terms in the right hand side are normalized to make the maximum of either term 1. When \(\alpha\) is 1, we can view Eq.(12) as the previous method considering only predicted rating. When \(\alpha\) is 0, the items with high purchase probabilities are recommended to the user. Thus, \((1 - \alpha)\) can be regarded as the weight parameter of the expected purchase probabilities.

Changing the weight parameter \(\alpha\), users can be recommended items with a desirable balance between the purchase probability and the predicted rating. In the proposed method, we arrange a number of items in descending order by \(N(x_i, y_j)\).

4 EXPERIMENTS

4.1 Experimental Conditions

To demonstrate the utility of the proposed method, we performed a series of experiments with the MovieLens dataset [23] where 943 users, each with more than 20 ratings, rated a subset of 1,682 movies on a scale of 1 to 5. The dataset was collected on the internet during the period from July 1997 to April 1998. The whole dataset is made up of 100,000 ratings and is randomly divided into the training dataset of 80,000 ratings and the test dataset of 20,000 ratings. In the simulation experiment, five sets of training and test datasets are made and average performance in these datasets is examined. The number of two latent classes \(K\) and \(L\) were set to 10 and 20, following the previous research [20]. The same numbers of latent classes, 10 and 20, were used in [20] and it is stated that varying the number of classes from \(5 \times 10\) to \(20 \times 40\) gives us similar results to those reported in the tables shown in [20]. We also conducted many pre-experiments. Although there is the best combination of numbers of latent classes for the prediction of rating, the results were not greatly influenced by varying the numbers of latent classes if those are not too large or too small.

As initial values for probabilities of latent variables, we generate random values following a uniform distribution and set these as initial values in iterative procedures of the EM algorithm in the experiments. Since the result varies according to initial values, we conducted experiments 10 times changing default values. We varied \(\alpha\) ranging from 0 to 1 by units of 0.1. We also show the results with three different numbers of recommended items: 10, 20 and 30.

4.2 Evaluation Metrics

We utilize two evaluation metrics: precision [27] and average rating. Precision is commonly used in much research on recommender systems and is calculated as in Eq.(13).

\[ \text{Precision} = \frac{G}{NJ}, \] (13)

where \(N\) is the number of recommended items for each user, and \(G\) is the number of recommended items which are in the test dataset. Employing this metric, we are able to evaluate the performance of our proposal for the degree of purchasing interest for recommended items.

Average rating is our newly invented metric and is calculated as in Eq.(14). This can evaluate the degree of post-purchase satisfaction of recommended items after purchase of those items.

\[ \text{Average Ratings} = \frac{S_R}{G}, \] (14)

where \(S_R\) is the sum of the ratings of recommended items which are also included in the test dataset.

The evaluation by Eqs. (13) and (14) cannot clarify the real performance of the recommender system when items are actually recommended to users because it only equates to the prediction accuracy for the test data of benchmark data which were previously divided from the training data. However, these equations are usually used in the research field of recommender systems [20], [21], [27]. This is because it is difficult to evaluate the accuracy of recommendation by reactions of real users.
4.3 Experimental Results

4.3.1 Precision

Figure 5 shows precision for different values of $\alpha$ and $N$. It shows that the precision curve looks unimodal with respect to $\alpha$. Though this result shows the average of five different sets of training and test datasets, we verified the unimodality of precision for all cases of different training data and test datasets. The shapes plotted for different datasets were similar and took maximum values at almost the same $\alpha$.

When $\alpha$ takes 0.7 for $N = 10$, the proposed method demonstrates the best results. When $\alpha$ takes 0.5 for $N = 20$ and 30, the proposed method has the best results. The proposed method is much better than the existing approach ($\alpha=1$). The predicted purchase probability $P(x_i|y_j)$ equals to the degree of purchasing interest. When $\alpha=1$, it equals the conventional method, where the predicted purchase probability $P(x_i|y_j)$ is not taken into account for recommendation. $(1 - \alpha)$ can be regarded as the weight of the predicted purchase probability $P(x_i|y_j)$. Compared with the setting $(1 - \alpha) = 0$, the precision is improved by calibrating the weight of the predicted purchase probability $P(x_i|y_j)$. We thus confirm that predicted purchase probability reflects the degree of purchasing interest for items.

4.3.2 Average Rating

Figure 6 shows the average rating for different values of $\alpha$ and $N$.

In the proposed method, the higher the value of $\alpha$, the better the average rating until $\alpha = 0.9$. In other words, considering predicted rating strongly contributes to the rise of post-purchase satisfaction of recommended items.

When $\alpha = 1.0$, the average rating is decreased compared with the case of $\alpha = 0.9$. This is because the items with many items with high predicted ratings do not appear in the test data. If some items with high predicted ratings have low ratings, then the average rating is strongly influenced when $N$ is small. This result shows that the proposed method considering not only the predicted rating but also the purchase probability is effective to improve the average rating.

4.3.3 Analysis of Precision and Average Rating

From Figs. 5 and 6, we are able to see the trade-off between predicted rating and predicted purchase probability. Each user can choose a desirable balance between predicted purchase probability and predicted rating by considering the trade-off.

In the proposed method, when a user designates $\alpha$ in the range $0.5 \leq \alpha \leq 0.7$, the precision takes almost the same value, while the average rating can be improved approximately 0.1 or 0.2 points by setting $\alpha = 0.7$ compared with the case of $\alpha = 0.5$. Thus users can be recommended the items which are expected to have relatively higher purchasing interest and post-purchase satisfaction.

4.3.4 Analysis of Purchasing Behaviors of Each User

In this section, the purchasing behaviors of each user are discussed. Figure 7 shows a scatter diagram with average predicted purchase probability and average predicted rating of each user.
Fig. 7  User Distribution

The points in the scatter diagram are users. The predicted purchase probability is normalized by the maximum value and the predicted rating normalized by the maximum rate 5. From this scatter diagram, there is no obvious relationship between average predicted purchase probabilities and average predicted ratings. This shows that the recommendation of the items with high predicted ratings does not always lead to recommendation the items with high purchase probability. The proposed method is an effective model to consider the balance between predicted purchase probabilities and predicted ratings.

5 CONCLUSION

In this study, we proposed a new method that takes both predicted purchase probability and predicted rating into account. In the simulation experiments, we clarified its effectiveness on precision and average rating. We can treat the trade-off between precision and average rating by adjusting the parameter $\alpha$. The EC site managers can set the parameter $\alpha$ based on their strategy and it is reasonable for them.

A future work is to extend this model to be able to automatically set appropriate $\alpha$ for each user. From a long-term viewpoint, the customer satisfaction should contribute to an increase in continuous purchase probability. If we can estimate the best parameter $\alpha$ for the purchase probability from the database, the sales amount can be maximized with high customer satisfaction.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Takashi Ishida and Dr. Go Hosoya for their helpful comments and suggestions. The authors would also like to express their thanks to all members of Goto laboratory for their helpful discussion. This research was partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (C), 23510192.

REFERENCES


