Research Paper

JIS

Journal of Information Science
1–13

DOI: 10.1177/0165551518782831
journals.sagepub.com/home/jis
SAGE

Topic-based hierarchical Bayesian linear regression models for niche items recommendation

Yezheng Liu
School of Management, Hefei University of Technology, China; Key Laboratory of Process Optimization and Intelligent Decision-Making, Ministry of Education, China

Qiang Xiong
School of Management, Hefei University of Technology, China; Key Laboratory of Process Optimization and Intelligent Decision-Making, Ministry of Education, China

Jianshan Sun
School of Management, Hefei University of Technology, China; Key Laboratory of Process Optimization and Intelligent Decision-Making, Ministry of Education, China

Yuanchun Jiang
School of Management, Hefei University of Technology, China

Thushari Silva
Faculty of Information Technology, University of Moratuwa, Sri Lanka

Haifeng Ling
School of Management, Hefei University of Technology, China; Key Laboratory of Process Optimization and Intelligent Decision-Making, Ministry of Education, China

Abstract
A vital research concern for a personalised recommender system is to target items in the long tail. Studies have shown that sales of the e-commerce platform possess a long-tail character, and niche items in the long tail are challenging to be involved in the recommendation list. Since niche items are defined by the niche market, which is a small market segment, traditional recommendation algorithms focused more on popular items promotion and they do not apply to the niche market. In this article, we aim to find the best users for each niche item and proposed a topic-based hierarchical Bayesian linear regression model for niche item recommendation. We first identify niche items and build niche item subgroups based on descriptive information of items. Moreover, we learn a hierarchical Bayesian linear regression model for each niche item subgroup. Finally, we predict the relevance between users and niche items to provide recommendations. We perform a series of validation experiments on Yahoo Movies dataset and compare the performance of our approach with a set of representative baseline recommender algorithms. The result demonstrates the superior performance of our recommendation approach for niche items.

Keywords
Expectation–maximisation algorithm; hierarchical Bayesian linear regression models; niche item recommendation; personalised recommendation

Corresponding author:
Jianshan Sun, School of Management, Hefei University of Technology, Tunxi Road No.193, Hefei 230009, Anhui, China.
Email: sunjs9413@hfut.edu.cn

1. Introduction

Recommender systems [1] mediate information overload problem associated with finding relevant items to users. Traditional recommendation algorithms used in such systems focus more on gaining high accuracy in general [2,3] and have overlooked user-level satisfaction. Accuracy-based recommendation algorithms, that is, collaborative filtering recommendation algorithms [4,5] and matrix factorisation techniques [6], often suggest items which are popular among users and ignore items which are less popular. Recent studies have focused on other evaluation metrics including diversity [7,8] and serendipity [9,10] to evaluate how good the recommendations are [11]. It is important to dig out niche preferences of users in order to achieve high level of user satisfaction. The phenomena of 'Long-tail' in the item recommendation context were introduced by Anderson [12] to handle recommendation of less popular items based on users' niche preferences.

Long-tail items are less popular items that have very few ratings and belong to the tail of the sales distribution. Long-tail recommendation has gained wide research attention recently and consists of diverse recommendation algorithms for discovering long-tail items. Wang et al. [13] proposed a multi-objective evolutionary algorithm to recommend accurate and unpopular items through trade-off accuracy and novelty. However, the niche items which are a specific class of long-tail items have been overlooked in the item recommendation literature. The study by Wang et al. [14] did not distinguish niche items and long-tail items. Niche items are defined by its niche market and item features aimed at satisfying specific market needs, such as craft beer and left-hand mouse. Thus, niche items refer to less popular items which are similar to long-tail items. Previous research [15] suggests that niche items are the resultant of special tastes and habits of consumer and thus have better quality and higher rating than general long-tail items. Therefore, a niche item can be discovered based on the popularity and rating.

Since niche items are defined by its niche market which represents potential incoming, the niche market has been considered as an important competitive market [16]. Niche marketing strategy can be considered as an excellent approach for specialised companies to develop niche items to satisfy specific market needs [17–19]. As niche marketing strategy is to find a potential user group for each niche item, niche item recommendation should focus more on users who need niche items; but, traditional recommendation techniques provide the recommendation list for each user, which is not effective for niche item recommendation. In order to achieve niche item recommendation effectively, we should find a new recommendation approach which forms a niche item perspective.

As niche items receive few ratings, it can cause sparsity problems which were overlooked by traditional recommendation techniques. Traditional recommendation algorithm usually added other associated information to alleviate sparsity problem [20,21]. Other methods such as the hierarchical Bayesian model is also used to alleviate sparsity problem through sharing information among items [22,23]; but, there are various types of items which have different rating pattern and popularity. Item information may not be consistent in each corresponding type. The inconsistent information will impair the performance of the hierarchical Bayesian model [24], thus directly sharing information among all items may harm the niche item recommendation. Sparsity problem restricts the effectiveness of the niche item recommendation.

In order to address the above-mentioned issues, this article proposes a topic-based hierarchical Bayesian linear regression (THBLR) model for niche item recommendation. The proposed method first distinguishes the popular items, long-tail items and niche items with the aim of identifying the best users for a given niche item based on their relevance. In order to alleviate sparsity problem in niche item recommendation, we build a niche item subgroup for each niche item and formulate a hierarchical Bayesian linear regression (HBLR) model for each niche item subgroup to share information among items. Thus, the sparsity in niche items subgroup is greatly reduced compared with the sparsity in whole U-I rating matrix, and sharing information among items in niche item subgroup adds value in increasing effectiveness for niche item recommendation.

The main contributions of this work can be summarised as follows:

1. We distinguish the popular items, long-tail items and niche items and proposed a method for niche item recommendation which is better applied to the niche market.
2. We utilise the latent Dirichlet allocation (LDA) model to find the items which have the same topic with the niche item and build a niche item subgroup for each niche item.
3. We build a HBLR model for each niche item subgroup to share information among items in the same subgroup to alleviate the sparsity problem in niche item recommendation.

The rest of this article is structured as follows. Section 2 introduces related work. Section 3 describes the proposed THBLR model in detail. Section 4 presented the experimental settings and results. Finally, in section 5, we provided the conclusion and the future work.

2. Related work

A niche market is a small market segment made up of a small group of users who need the niche item. The companies that develop niche items are specialised firms which design a separate marketing plan for each niche item [16,17]. Since the mass market company achieves high yields, whereas the specialised company achieves high margins [15], it is important to implement a niche marketing strategy for the specialised company. Traditional recommendation algorithms usually employ ratings to design recommendation system to provide users with items. Thus, the personalised recommendation technology for niche items can be used as niche marketing strategy. First, the concept of the niche market and niche marketing strategy are presented in detail. Then, we present a comprehensive review on recommendation system literature regarding long-tail items and niche items.

2.1. Niche market and niche marketing strategy

The niche items are defined by its niche market and aimed at satisfying the market needs from the specific user groups. Since niche items represent potential incoming, it is important to implement an efficient niche marketing strategy for the specialised company [25].

Michaelson [26] defines that niche marketing strategy is to find a small group of users with similar needs. The traditional marketing strategy for niche item is mainly for specific niche item. Wu et al. [27] proposed that niche tourism was a new way of personalised tourism which described the characteristics of a small group of tourists with similar desires. Murray and O'Neill [28] proposed that craft beer market was a niche market to meet the needs of craft beer lovers. As marketing strategy for niche item mainly depends on the nature of niche item [27–29], only few marketing strategies are available for niche marketing. Thus, there is a lack of intelligent methods for niche marketing strategy.

2.2. Recommender systems for niche items

The traditional recommendation algorithm is used to provide users with items that suit their needs. Most widely used recommender algorithms are content-based recommendation algorithms [30], collaborative filtering recommendation algorithms [31] and hybrid recommendation algorithms [32]. The content-based recommendation algorithms suggest items similar with those rated by the users based on the attributes of items. The collaborative filtering recommendation algorithms suggest items preferred by users who are similar to the target users based on the user–item interaction information. The hybrid recommendation algorithms combine content-based recommendation algorithms and collaborative filtering recommendation algorithms in order to overcome their limitations. Traditional recommendation algorithms mainly focused on accuracy which provide users with popular items, thus did not apply to the niche market. An efficient intelligent recommendation algorithm is particularly important for niche marketing strategy.

However, little research exists on recommendation algorithms for niche items. Wang et al. [14] proposed a pattern-based method to recommend both popular and niche items flexibly, but fail to distinguish niche items from long-tail items. For niche items, since fewer ratings are available, researchers have focused on the long-tail item recommendation. Traditionally, recommendation techniques for long-tail items have different ways for increasing long-tail items in the recommendation list. There are two main types of recommendation algorithms for long-tail items. The first category is the multi-objective optimisation recommendation algorithms which optimise multiple goals such as accuracy and diversity and other goals simultaneously to promote the long-tail items in the recommendation process. Shi [33] proposed a graph-based recommendation algorithm to promote the long-tail items with trade-off among accuracy, diversity, similarity and long tail. Wang et al. [13] proposed a novel multi-objective evolutionary algorithm to recommend accurate and unpopular items considering accuracy and novelty simultaneously. The other type of recommendation algorithms increases the long-tail items in the recommendation list directly, which divides the items into popular items and long-tail items and treats these two items separately. For example, Johnson and Ng [34] recommended long-tail items to users based on the tripartite graph and Markov process. Yin et al. [35] proposed a graph-based algorithm to find long-tail items preferred by users which extended Hitting Time and Absorbing Time algorithm.

Traditional long-tail recommendation technique is to provide the recommendation list for each user with more long-tail items. However, niche item recommendation is different from long-tail recommendation and it is mainly to find the latent users for each niche item. Furthermore, these algorithms fail to consider the sparsity problem as well.

Niche item recommendation needs to alleviate the sparsity problem to a greater extent than traditional recommendation algorithms. In order to address this gap, we develop a THBLR models to alleviate the sparsity problem in niche item recommendation. For each niche item, we provide a recommended user list which has the most top relevance with target niche items. This approach caters to the niche marketing strategy because it can find the latent users for each niche item.

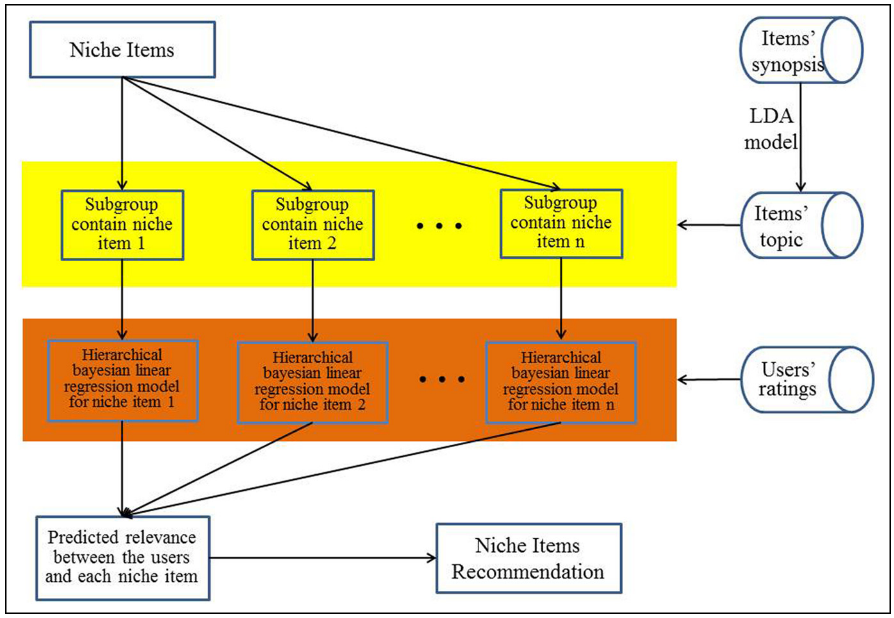

Figure 1. The framework of topic-based hierarchical Bayesian linear regression model.

3. Niche item recommendation algorithm

The proposed THBLR model for niche item recommendation is shown in Figure 1. From the perspective of niche items, we aim to find best users for each niche item. We first distinguish the popular items, long-tail items and niche items and build a niche item subgroup for each niche item based on descriptive information of items. Each niche item subgroup contains the target niche item and other items which have the same topic with the target niche item. Then, we formulate a HBLR model for each niche item subgroup to share information among items. Finally, we predict the relevance between the users and each niche item and provide each niche item with a user list that has the most top relevance with niche items. The THBLR models are developed using the likelihood optimising via the expectation–maximisation (EM) algorithm.

The task of the niche item recommendation system is to recommend a ranked list of users that are relevant to niche item. For each niche item, the recommendation system generates a model based on the items' history. To make a clear description of our recommendation model, the notations used in the text are presented in Table 1.

3.1. Niche items subgroup identification

Initially, we distinguish the popular items, long-tail items and niche items. We can argue that the items belong to the head of the sales distribution represent the set of the popular items and the items belong to the tail of the sales distribution conform the set of the long-tail items. We select niche items from a subset of long-tail items, while niche item also receives few ratings just like long-tail items, but they exhibit better quality and higher rating than the long-tail items. Therefore, items which are gaining ratings higher than the average rating of all long-tail items are considered as niche items.

For each niche item, we constructed a niche item subgroup which contains items which have the same topic with the target niche item. In this article, we utilised the LDA model to find items which have the same topic with the target niche item. The LDA model [36] is a generative probabilistic model. We consider the descriptive information (synopsis) of items as the document and utilise the synopsis information of items to obtain the topic distribution of items based on LDA model.

Since we get the topic distribution θ_m of each item m based on LDA model, we sample a topic from the distribution of item m as the topic which belongs to the item. Thus, we can randomly select an item list including items which have the same topic with each niche item. As the different number of topics could result the different topic distribution of items, and the topic to which items belong is also changed, items to be included in the niche item subgroups will also vary. Therefore, we conducted experiments with a different number of topics to obtain the best number of topics. In addition, we also obtain user feature vector based on the topic distribution of items the user rated.

Table 1. Notation.

Notation	Description
M	Number of items
V	Number of users
N	Number of niche items
J_n	Item list whose topic similar to niche item *n*
U_m	Number of users who rates item *m*
m	The index of items
n	The index of niche items
u	The index of users
ω_m	The item model parameter associated with item *m* is a *K*-dimensional vector
ϕ	The prior of parameter $\phi = (\mu, \Sigma)$
μ	The mean of the Gaussian distribution
Σ	The covariance matrix of the multivariate Gaussian distribution
ε	Random noise
σ	Standard deviation over all ratings
D_m	A set of data associated with item *m*, *Dm* = {(*xu*, *ym*, *u*)}
x_u	*X* is the user feature matrix, and $x_u \in X$ is the *K*-dimensional feature vector representing the user *u*
$y_{m,\,u}$	The relevance between user *u* and item *m*
θ_m	Distribution over topics for item *m*
a, *b*	Real numbers

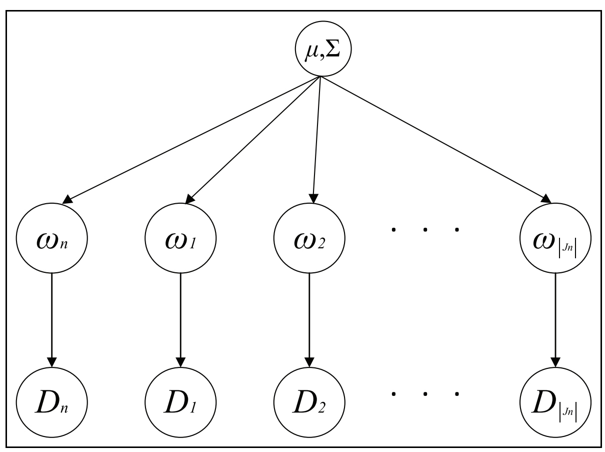

Figure 2. Illustration of dependencies of items in the topic-based hierarchical Bayesian model.

3.2. THBLR models and parameter learning

The hierarchical Bayesian models [22] have been widely used in information retrieval applications. HBLR models, which are one of the simplest hierarchical Bayesian models, have been used in recommendation system and achieved good performance [37]. Figure 2 shows the graphical representation of dependency of items in a topic-based hierarchical Bayesian model for niche item recommendation. In this graph, the items in niche item subgroup containing niche item n form a HBLR model, and a set of data associated with item m is represented by a set D_m. The item model parameter is represented by a vector ω_m. Then, the relevant y between user u and item m can be predicted through the user feature vector x_u and a given estimation of parameter ω_m using a function $y = f(x_u, \omega_m)$. One commonly used Bayesian linear regression model is $y = \omega^T x + \varepsilon$, where ε is a random noise, $\varepsilon \sim N(0, \sigma^2)$. Assume that the model parameters ω_m of item m which is also the regression coefficient of the Bayesian linear regression model is an independent draw from a prior distribution $P(\omega|\phi)$. Since one commonly used prior distribution for Bayesian linear regression models is the Gaussian distribution, we set the prior distribution of ω_m be a Gaussian distribution with parameter $\phi = (\mu, \Sigma)$.

We assume the prior distribution of μ be a Gaussian distribution and the prior distribution of Σ which is a covariance matrix be an inverse Wishart distribution. Then, we can build a hierarchical Bayesian model to share information among items through the prior distribution of Σ. We can reliably estimate the regression coefficients ω_m of each item m through sharing information among items in the same subgroup based on the prior $\phi = (\mu, \Sigma)$. Thus, we can alleviate the sparsity problem in niche item recommendation through the HBLR model. The conjugate prior distribution μ and Σ can be expressed as

$$p(\mu, \Sigma^2) = N(\mu|\mu_0, a\Sigma^2)IW(\Sigma^2|b, \Sigma_0^2) \tag{1}$$

where a, b, μ_0 and Σ_0^2 are provided to the system.

With these settings, we have the following sampling process for the HBLR model.

1. Σ is sampled randomly first from an inverse Wishart distribution: $\sum \sim IW(b, \Sigma_0^2)$; μ is sampled randomly after Σ is observed from a Gaussian distribution: $\mu \sim N(\mu_0, a\Sigma^2)$.
2. For each item m, ω_m is sampled randomly from a Gaussian distribution: $\omega_m \sim N(\mu, \Sigma^2)$.
3. For each user u, $y_{m,u}$ is sampled randomly from a Gaussian distribution: $y_{mu} \sim N(\omega_m^T x_u, \sigma^2)$.

The joint likelihood distribution for all the variables in the HBLR model is

$$P(D, \omega, \mu, \Sigma, \sigma) = P(\mu, \Sigma)P(\sigma) \prod_{m=1}^{|J_n|+1} \left[P(\omega_m|\mu_\omega, \Sigma_\omega) \prod_{u=1}^{U_m} P(y_{m,u}|x_u, \omega_m, \sigma) \right] \tag{2}$$

The regression coefficients ω of each item model in the topic-based Bayesian linear regression models are not directly visible, and only associated data D are observed. Therefore, the regression coefficients ω of each item model need to be estimated. It is easy to find the optimal ω of each item model when the conjugate prior$\phi = (\mu, \Sigma)$ is known. The maximum a priori solution of ϕ is given by

$$\varphi_{MAP} = \arg\max_{\varphi} P(\varphi|D) = \arg\max_{\varphi} \int_{\omega} P(D|\omega, \varphi)P(\omega|\varphi)P(\varphi)d\omega \tag{3}$$

However, finding the optimal solution of ϕ is not easy. Since the EM algorithm [38] is an iterative method for parameter learning, which is a commonly used method due to its convergence guarantee, we learn the HBLR model via the EM algorithm. We build a HBLR model for each niche item. For each niche item n, there are $|J_n| + 1$ items containing target niche item in a HBLR model. The regression coefficients ω_m of each item m in Bayesian linear regression model is the unobservable hidden variables and we have the likelihood of complete data as

$$P(y, \omega|x, \mu, \Sigma, \sigma) = \prod_{m=1}^{|J_n|+1} (P(\omega_m|\mu_\omega, \Sigma_\omega) \prod_{u=1}^{U_m} P(y_{m,u}|x_u, \omega_m, \sigma)) \tag{4}$$

and the log-likelihood of complete data is

$$\ln P(y, \omega|X, \mu, \Sigma, \sigma) = \sum_{m=1}^{|J_n|+1} (\ln P(\omega_m|\mu_\omega, \Sigma_\omega) + \sum_{u=1}^{U_m} \ln P(y_{m,u}|x_u, \omega_m, \sigma)) \tag{5}$$

The corresponding expectation is

$$\begin{aligned} Q = E(\ln P(y, \omega|X, \mu, \Sigma, \sigma)) = \sum_{m=1}^{|J_n|+1} \left[-\frac{K + U_m}{2} \ln(2\pi) - \frac{U_m}{2} \ln(\sigma^2) - \frac{1}{2} \ln(\Sigma_\omega^2) \right] \\ -\tfrac{1}{2\sigma^2} E\left(\sum_{m=1}^{|J_n|+1} \sum_{u=1}^{U_m} (y_{m,u} - {\omega_m}^T x_u)^2 \right) - \tfrac{1}{2} E\left(\sum_{m=1}^{|J_n|+1} (\omega_m - \mu_\omega)^T (\Sigma_\omega^2)^{-1} (\omega_m - \mu_\omega) \right) \end{aligned} \tag{6}$$

Based on the EM formulas presented by Yu et al. [39], we find the optimal hyperparameters through the following EM steps.

E steps: Estimate the regression coefficients ω_m of each item m in Bayesian linear regression model $P(\omega_m|D, \varphi) = N(\bar{\omega}_m, \Sigma_{\omega_m}^2)$ based on the prior ϕ

$$\bar{\omega}_m = \left((\Sigma_\omega^2)^{-1} + \frac{1}{\sigma^2}\sum_{u=1}^{U_m} x_u x_u^T\right)^{-1}\left(\frac{1}{\sigma^2}\sum_{u=1}^{U_m} x_u y_{m,u} + (\Sigma_\omega^2)^{-1}\mu_\omega\right) \tag{7}$$

$$\Sigma_{\omega_m}^2 = \left((\Sigma_\omega^2)^{-1} + \frac{1}{\sigma^2}\sum_{u=1}^{U_m} x_u x_u^T\right)^{-1} \tag{8}$$

M step: Optimise the prior ϕ and parameter σ based on the last E step

$$\mu_\omega = \frac{1}{|J_n|+1}\sum_{m=1}^{|J_n|+1} \bar{\omega}_m \tag{9}$$

$$\Sigma_\omega^2 = \frac{1}{|J_n|+1}\sum_{m=1}^{|J_n|+1}\left(\Sigma_{\omega_m}^2 + (\bar{\omega}_m - \mu_\omega)(\bar{\omega}_m - \mu_\omega)^T\right) \tag{10}$$

$$\sigma^2 = \frac{1}{\sum_{m=1}^{|J_n|+1} U_m}\sum_{m=1}^{|J_n|+1}\sum_{u=1}^{U_m}(y_{m,u} - \omega_m{}^T x_u)^2 \tag{11}$$

The EM algorithm keeps alternating between the E step and M step until the value obtained in two successive iteration steps is less than the threshold γ. In our experiments, γ is 10^{-5}.

3.3. Top-N recommendation with the THBLR model

We have learned a THBLR model for each niche item. In this section, we describe how to find best users for each niche item.

Since we obtained the parameters ω_m and σ of the HBLR models for each niche item m via EM algorithm, we can sample $y_{m,u}$ randomly from a Gaussian distribution $y_{m,u} \sim N(\omega_m{}^T x_u, \sigma^2)$ based on each user feature vector x_u. Therefore, we can get the relevance of all users for niche items and commonly the users who having top relevance are the best users for niche items; thus, we provide each niche item with recommendation for users that have the most top relevance with niche items.

4. Experimental results

In order to evaluate the effectiveness of our proposed method, we conduct a series of experiments. We describe the experimental settings in detail and analyse the results.

4.1. Data description

Our experiments are performed on Yahoo Movies dataset. Yahoo Movies dataset is a movie rating dataset provided by the Yahoo Research Alliance Webscope programme. It contains 221,367 ratings from 7642 users on 11,916 movies, each movie has corresponding descriptive information (synopsis), and each rating had five possible values, ranging from 1 (not relevant) to 5 (very relevant). Due to the data sparsity, we filtered out a subset of the dataset via the following two criteria, one for each movie that has been rated by at least five users and another one for each user that has rated at least 20 movies. The statistics of our experimental dataset is shown in Table 2.

The experimental dataset was divided into training subset and test subset. We included the 80% of each item's ratings in the training subset and the rest 20% of each item's ratings in the test subset while following Valcarce et al. [40].

We chose the long-tail items and niche items as follows.

Since long-tail items are less popular items that have very few ratings and belong to the tail of the sales distribution, we selected a proportion (e.g. 80%) of least rated items as the long-tail items. We use a fixed threshold c, and then, a set of long-tail items can be selected with less than c ratings

$$I_1 = \{m \in I | p_m < c\} \tag{12}$$

where I is the set of all items, I_1 is a set of long-tail items, and p_m is the number of users who rated item m.

Table 2. Statistics of the Yahoo Movies dataset.

	Raw dataset	Experimental dataset
Number of users	7642	2878
Number of movies	11,916	3032
Total number of ratings	221,367	139,121
Average number per user	28	48
The sparsity (%)	99.8	98.4

We selected niche items from a subset of long-tail items, while niche item also receives few ratings just like long-tail items, but it has better quality and higher rating than long-tail items. Therefore, we selected niche items from the long-tail items which have a higher average rating than the average rating of all long-tail items. We also used a fixed threshold d based on the average rating of all long-tail items, and then, we can select a set of niche items I_2 that have higher than average rating d

$$I_2 = \{m \in I_1 | q_m > d\} \tag{13}$$

where I_2 is a set of niche items and q_m is the average rating of item m.

4.2. Benchmark methods

We compared our proposed algorithm with benchmark methods to assess the performance of our proposed THBLR models. Since no specific algorithm exists for niche item recommendation from the perspective of niche items we proposed, we had to use standard collaborative filtering algorithms from the state of the art, including item-based collaborative filtering recommendation algorithm [5] (CF_Item), probabilistic matrix factorisation (PMF) [41] and traditional HBLR model [22]. We describe the baselines in the following:

Item-based collaborative filtering (CF_Item): This method calculates the similarity between items and then recommends items to the target user based on the prediction ratings. In this article, we use Pearson correlation to compute the similarity between items.
PMF: This method computes the factorisation of the ratings matrix which adopts a probabilistic model with Gaussian distribution.
HBLR Model: This method borrows information from other users through the use of a Bayesian hierarchical model, and the relevance between users and items is sampled randomly from a linear regression model.

We adapted those algorithms to niche item recommendation from the perspective of niche items although these methods are designed from the perspective of users. For recommending users to niche items, we generated a recommendation list for each niche item. This list contains users with the largest predicted relevance.

4.3. Evaluation metric

In order to evaluate the performance of niche item recommendation system, we adopt precision and recall to evaluate recommender methods from the perspective of niche items which means we provide a recommended list containing users which have the most top relevance with niche items for each target niche items. The two measures are defined as

.

$$Precision = \frac{1}{|I_2|} \sum_{i \in I_2} \frac{|l_i|}{|L_i^2|} \tag{14}$$

$$Recall = \frac{1}{|I_2|} \sum_{i \in I_2} \frac{|l_i|}{|L_i^1|} \tag{15}$$

where L_i^1 is the users in the test subsets who rated the niche item i, L_i^2 is the users in recommendation list for niche item i, and l_i is the users in recommendation list for niche item i who also in test subsets.

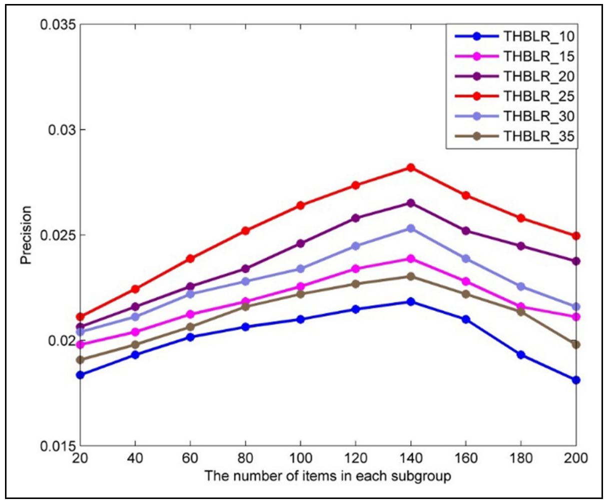

Figure 3. Evaluations on our THBLR method under different number of topics on different number of items in each subgroup.

4.4. Experimental result and analysis

In this section, we first show the recommendation results of our THBLR. We identify the performance differences of THBLR at different settings under the 80% proportion of long-tail items in all items. Then, we compare our results with the results of the three benchmark methods with the 80% proportion of long-tail items in all items. We also compare the recommendation results by considering different proportions including 50%, 60%, 70%, 80% and 90% of long-tail items.

4.4.1. Evaluation of THBLR at different settings. Our recommendation method considers the performance differences of THBLR at two variants: number of topics and the number of items in each subgroup.

The number of topics is the variable in LDA model, while we utilise the synopsis information of items to obtain the topic distribution of items based on LDA model. Since a different number of topics result in the different topic distribution of items, the items in the niche item subgroups are changed, and the user feature vector is changed too. Therefore, we conducted experiments with a different number of topics to determine the best values.

The number of items in each subgroup has a great impact on the THBLR model. Since the sparsity in niche items subgroup greatly reduced compared with the sparsity in whole U-I rating matrix, the different number of items in each subgroup results in different degree of sparseness reduction and different size of information shared among items in each niche group. Thus, we conducted experiments with different number of items in each subgroup to determine the best values.

Figure 3 illustrates the recommendation performance from testing the two variants, in comparison with different number of topics and the different number of items in each subgroup. For the different number of topics, we compare the recommendation results of our THBLR method with 10, 15, 20, 25, 30 and 35 topics. For the different number of items in each subgroup, we compare the recommendation results of our THBLR method with the number of items at 20, 40, 60, 80, 100, 120, 140, 160, 180 and 200. The THBLR_10, THBLR_15, THBLR_20, THBLR_25, THBLR_30 and THBLR_35 in Figure 3 represent our proposed THBLR under 10, 15, 20, 25, 30 and 35 topics, respectively. Moreover, we evaluate those methods on a different number of items in each subgroup.

From the results of the experiment, it can be seen that the best recommendation performance is achieved at the number of topics is 25, and the number of items in each subgroup is 140. For the number of topics, we found that both small (10 topics) and large (35 topics) values will cause the low recommendation performance. This is because when the number of topics is small, it easily leads to the problem of under fitting, and when the number of topics is large, it easily leads to overfitting. For the number of items in each subgroup, we found that both small (20 items) and large (200 items) values in each subgroup will cause the low recommendation performance. This is because when the number of items in each subgroup is small, the HBLR model for each subgroup cannot learn parameters accurately. When the number of items in each subgroup is large, the topics of items in each subgroup vary greatly, and sharing information among those items harm the niche item recommendation. Therefore, we compare our THBLR method with benchmark methods as the number of topics is 25 and the number of items in each subgroup is 140.

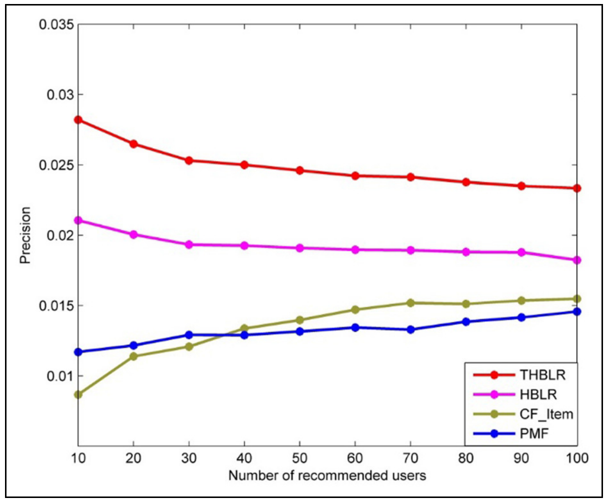

Figure 4. Comparison of the precision of THBLR with CF_Item, PMF and HBLR.

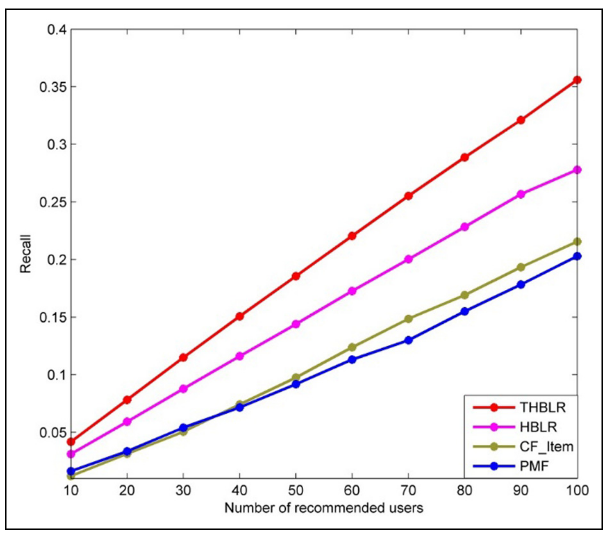

Figure 5. Comparison of the recall of THBLR with CF_Item, PMF and HBLR.

4.4.2. Comparison of our THBLR with the three benchmark methods. In this section, we compare the performance of THBLR method with three benchmark methods using Yahoo Movies dataset. We designed our experiments with the objective of recommending niche items. As our objective is to recommend niche items, we adopt precision and recall to evaluate recommender methods from the perspective of niche items and optimise the parameters of all the recommendation algorithms from the perspective of niche items. We compute the precision and recall for different number of recommended users ranging from 10 to 100 for each niche items. The recommendation performance of the THBLR with the three benchmark methods is shown in Figures 4 and 5. From the results, we can see that the proposed THBLR method has significantly higher precision and recall than CF_Item, PMF and HBLR, which indicates a better recommendation performance. Comparing THBLR with HBLR, the former has higher precision and recall, which confirms that sharing information among all items harms the recommendation performance. At the same time, comparing HBLR with CF_Item and PMF, the former has higher precision and recall, which indicates that HBLR is more effective than CF_Item and PMF in the sparsity environment. From these results, we can see that the best recommendation performance appears at $N = 10$ for the THBLR and HBLR while other recommendation methods achieve the best recommendation performance when N is bigger. Since the precision of the recommendation method should decrease with the

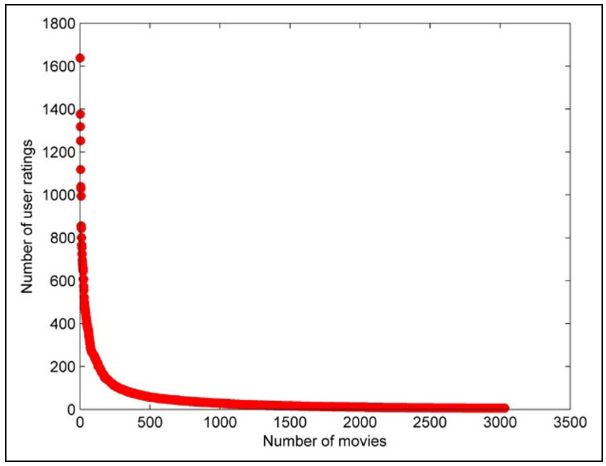

Figure 6. The number of user ratings on movies.

Table 3. The number of long-tail items and niche items selected in different long-tail division.

	The proportion of long-tail items in all items				
	50%	60%	70%	80%	90%
The number of long-tail items	1517	1820	2123	2426	2729
The number of niche items	845	1006	1193	1354	1505

increase in the number of recommended users, it can be seen that our proposed THBLR and HBLR have a better stability for niche item recommendation.

4.4.3. Comparison on the different long-tail division. Since there is no public collection of studies that specifies the division of long-tail items and popular items, we designed the experiments to compare our proposed THBLR with the baselines under different long-tail division. As shown in Figure 6, a large proportion of movies in a market only generate a small proportion of ratings. The long-tail items in the tail of the curve have low popularity. Table 3 shows the number of long-tail items and niche items selected in the different long-tail division. While the proportion of long-tail items in all the items is bigger, the more the number of long-tail items, and also the more the number of niche items. Thus, the dataset containing niche items becomes sparser when the proportion of long-tail items in all the items is smaller. The recommendation performance of the THBLR along with the baselines under different long-tail division is shown in Figure 7. The values 0.5, 0.6, 0.7, 0.8 and 0.9 in the abscissa indicate that the proportion of long-tail items in all items is 50%, 60%, 70%, 80% and 90%, respectively. The results clearly indicate that the precision of the THBLR and the baselines increased with the increment of the proportion of long-tail items in all items, and the THBLR outperforms the other three recommendation methods CF_Item, PMF and HBLR with the different long-tail division.

5. Conclusion and future work

With the increase in the number of items in the market, more and more items in the sales distribution can bring high profit. Niche items are the items aimed at satisfying specific market needs and sold in a niche market. It is important to implement a niche marketing strategy for a specialised firm. However, research on recommender systems has not paid much attention for the niche item recommendation. This article addressed an important problem in the recommender systems domain by designing a strategy for these niche item recommendations. Based on the niche item features and the niche marketing strategy, a new recommendation approach and evaluation metric from the perspective of niche items is proposed. In the proposed methods, the problem of sparsity has been addressed via the THBLR models. Thus, this research has important implications for researchers interested in niche market and recommendation systems. As potential incoming and the return on investment of niche markets are bigger than the return on investment of mass markets, it is

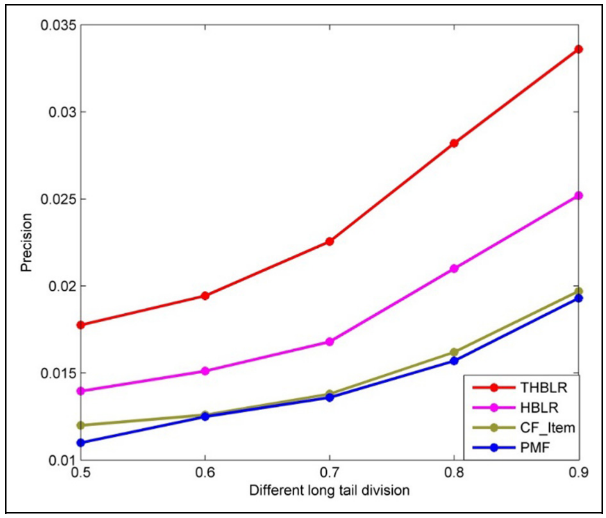

Figure 7. Comparison of the precision of THBLR with the baselines under different long-tail proportions.

important to implement a niche marketing strategy. Since there is a lack of specific methods for niche marketing, this article presents a practical way of deriving high-quality recommendations for niche items.

However, our work still has some limitations. The classification of popular items, long-tail items and niche items should be verified by conducting more experiments. Moreover, we only use the U-I ratings matrix and movie descriptive content information. Future works can utilise more information to distinguish niche items and improve the effectiveness of niche item recommendation. In addition, other methods to obtain the niche item subgroup are another research direction.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship and/or publication of this article.

Funding

This work is supported by the Major Programme of the National Natural Science Foundation of China (71490725), the Foundation for Innovative Research Groups of the National Natural Science Foundation of China (71521001), the National Natural Science Foundation of China (71501057, 71722010, 91546114, 91746302), the National Key Research and Development Programme of China (2017YFB0803303), the Fundamental Research Funds for the Central Universities (JZ2017HGTB0185) and Anhui Provincial Natural Science Foundation (1608085QG166).

References

[1] Ricci F, Rokach L and Shapira B. *Introduction to recommender systems handbook*. New York: Springer, 2011.
[2] Bu J, Shen X, Xu B et al. Improving collaborative recommendation via user-item subgroups. *IEEE T Knowl Data En* 2016; 28: 2363–2375.
[3] Lee J, Lee D, Lee Y-C et al. Improving the accuracy of top-N recommendation using a preference model. *Inform Sciences* 2016; 348: 290–304.
[4] Huang Z, Zeng D and Chen H. A comparison of collaborative-filtering recommendation algorithms for e-commerce. *IEEE Intell Syst* 2007; 22: 68–78.
[5] Sarwar B, Karypis G, Konstan J et al. Item-based collaborative filtering recommendation algorithms. In: *Proceedings of the 10th international conference on world wide web*, Hong Kong, China, 1–5 May 2001, pp. 285–295. New York: ACM.
[6] Koren Y, Bell R and Volinsky C. Matrix factorization techniques for recommender systems. *Computer* 2009; 42: 30–37.
[7] Hu L, Cao L, Wang S et al. Diversifying personalized recommendation with user-session context. In: *Proceedings of the 26th international joint conference on artificial intelligence*, 2017, pp. 1858–1864, https://www.ijcai.org/proceedings/2017/0258.pdf
[8] Muter I and Aytekin T. Incorporating aggregate diversity in recommender systems using scalable optimization approaches. *INFORMS J Comput* 2017; 29: 405–421.
[9] Kawamae N. Serendipitous recommendations via innovators. In: *Proceedings of the 33rd international ACM SIGIR conference on research and development in information retrieval*, Geneva, 19–23 July 2010, pp. 218–225. New York: ACM.
[10] Yang Y, Xu Y, Wang E et al. Improving existing collaborative filtering recommendations via serendipity-based algorithm. *IEEE T Multimedia*. Epub ahead of print 1 December 2017. DOI: 10.1109/TMM.2017.2779043.

[11] Kaminskas M and Bridge D. Diversity, serendipity, novelty, and coverage: a survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM Trans Interact Intell Syst* 2016; 7: 2.
[12] Anderson C. *The long tail: why the future of business is selling less of more*. New York: Hachette Books, 2006.
[13] Wang SF, Gong MG, Li HL et al. Multi-objective optimization for long tail recommendation. *Knowledge-Based Syst* 2016; 104: 145–155.
[14] Wang YQ, Wu JJ, Wu Z et al. Popular items or niche items: flexible recommendation using cosine patterns. In: *IEEE international conference on data mining workshop*, Shenzhen, China, 14 December 2014, pp. 205–212. New York: IEEE.
[15] Kotler P, Saliba SJ, Turner RE et al. *Marketing management: analysis, planning, implementation and control*. Canadian 6th ed. Scarborough, ON, Canada: Prentice-Hall Canada, 1989.
[16] Parrish ED, Cassill NL and Oxenham W. Niche market strategy for a mature marketplace. *Market Intell Plann* 2006; 24: 694–707.
[17] Parrish ED, Cassill NL and Oxenham W. Niche market strategy in the textile and apparel industry. *J Fash Mark Manag* 2006; 10: 420–432.
[18] Toften K and Hammervoll T. Niche marketing and strategic capabilities: an exploratory study of specialised firms. *Market Intell Plann* 2010; 28: 736–753.
[19] Toften K and Hammervoll T. Niche firms and marketing strategy: an exploratory study of internationally oriented niche firms. *Eur J Marketing* 2009; 43: 1378–1391.
[20] Melville P, Mooney RJ and Nagarajan R. Content-boosted collaborative filtering for improved recommendations. In: *Eighteenth national conference on artificial intelligence*, Edmonton, AB, Canada, 28 July–1 August 2002, pp. 187–92. Menlo Park, CA: American Association for Artificial Intelligence.
[21] Godoy-Lorite A, Guimerà R, Moore C et al. Accurate and scalable social recommendation using mixed-membership stochastic block models. *Proc Natl Acad Sci USA* 2016; 113: 14207–14212.
[22] Zhang Y and Koren J. Efficient Bayesian hierarchical user modeling for recommendation system. In: *Proceedings of the 30th annual international ACM SIGIR conference on research and development in information retrieval*, Amsterdam, 23–27 July 2007, pp. 47–54. New York: ACM.
[23] Gopal S, Yang Y, Bai B et al. Bayesian models for large-scale hierarchical classification. In: *Advances in neural information processing systems*, Lake Tahoe, NV, 3–6 December 2012, pp. 2411–2419. Red Hook, NY: Curran Associates Inc.
[24] Zhang Q, Wu D, Lu J et al. A cross-domain recommender system with consistent information transfer. *Decis Support Syst* 2017; 104: 49–63.
[25] McKenna R. Marketing in an age of diversity. *Harvard Bus Rev* 1988; 66: 88–95.
[26] Michaelson GA. Niche marketing in the trenches. *Market Commun* 1988; 13: 19–24.
[27] Wu C, Ho GT, Lam CH et al. An online niche-market tour identification system for the travel and tourism industry. *Internet Res* 2016; 26: 167–185.
[28] Murray DW and O'Neill MA. Craft beer: penetrating a niche market. *Brit Food J* 2012; 114: 899–909.
[29] Tassiopoulos D and Haydam N. Golf tourists in South Africa: a demand-side study of a niche market in sports tourism. *Tourism Manage* 2008; 29: 870–882.
[30] Mooney RJ and Roy L. Content-based book recommending using learning for text categorization. In: *Proceedings of the fifth ACM conference on digital libraries*, San Antonio, TX, 2–7 June 2000, pp. 195–204. New York: ACM.
[31] Aligon J, Gallinucci E, Golfarelli M et al. A collaborative filtering approach for recommending OLAP sessions. *Decis Support Syst* 2015; 69: 20–30.
[32] Yao L, Sheng QZ, Ngu AH et al. Unified collaborative and content-based web service recommendation. *IEEE T Serv Comput* 2015; 8: 453–466.
[33] Shi L. Trading-off among accuracy, similarity, diversity, and long-tail: a graph-based recommendation approach. In: *Proceedings of the 7th ACM conference on recommender systems*, Hong Kong, China, 12–16 October 2013, pp. 57–64. New York: ACM.
[34] Johnson J and Ng Y-K. Enhancing long tail item recommendations using tripartite graphs and Markov process. In: *Proceedings of the international conference on web intelligence*, Leipzig, 23–26 August 2017, pp. 761–768. New York: ACM.
[35] Yin HZ, Cui B, Li J et al. Challenging the long tail recommendation. *Proc VLDB Endow* 2012; 5: 896–907.
[36] Blei DM, Ng AY and Jordan MI. Latent Dirichlet allocation. *J Mach Learn Res* 2003; 3: 993–1022.
[37] Yu K, Tresp V and Yu S. A nonparametric hierarchical Bayesian framework for information filtering. In: *Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval*, Sheffield, 25–29 July 2004, pp. 353–360. New York: ACM.
[38] Kadir SN, Goodman DF and Harris KD. High-dimensional cluster analysis with the masked EM algorithm. *Neural Comput* 2014; 26: 2379–2394.
[39] Yu K, Tresp V and Schwaighofer A. Learning Gaussian processes from multiple tasks. In: *Proceedings of the 22nd international conference on Machine learning*, Bonn, 7–11 August 2005, pp. 1012–1019. New York: ACM.
[40] Valcarce D, Parapar J and Barreiro A. Item-based relevance modelling of recommendations for getting rid of long tail products. *Knowledge-Based Syst* 2016; 103: 41–51.
[41] Mnih A and Salakhutdinov R. Probabilistic matrix factorization. In: *Advances in neural information processing systems*, Vancouver, BC, Canada, 3–6 December 2007, pp. 1257–1264. Red Hook, NY: Curran Associates Inc.