A Mobile Services Collaborative Recommendation Algorithm Based on Location-Aware Hidden Markov Model

Mingjun Xin(✉), Shunxiang Li, Liyuan Zhou, and Guobing Zou

School of Computer Engineering and Science,
Shanghai University, Shanghai 200444, China
xinmj@shu.edu.cn

Abstract. Nowadays, location based services (LBS) has become one of the most popular applications with the rapid development of mobile Internet environment. More and more research is focused on discovering the required services among massive information according to the personalized behavior. In this paper, a collaborative filtering (CF) recommendation algorithm is presented based on the Location-aware Hidden Markov Model (LHMM). This approach includes three main stages. First, it clusters users by making a pattern similarity calculation of their historical check-in data. Then, it establishes the location-aware transfer matrix so as to get the next most likely service. Furthermore, it integrates the generated LHMM, user's score and interest migration into the traditional CF algorithm to generate a final recommendation list. The LHMM-based CF algorithm mixes the geographic factors and personalized behavior and experimental results show that it has more accuracy than other state-of-the-arts algorithms.

Keywords: Behavior prediction · LBS · LHMM · Collaborative recommendation

1 Introduction

With the rapid development of mobile Internet and spatial information processing technology, user's behavioral prediction stimulates considerable research interests. Collaborative filtering (CF) is an effective recommendation algorithm. But when it applies to the behavioral prediction, it has limitation [1]. Users' next action is greatly depends on their former choice, which are always not considered. Traditional CF algorithm is not as good as Hidden Markov Model (HMM) under LBS environment, which is frequently used to deal with states transition and predict the probability of services-to-service transfer.

As for HMM strategies, Blasiak and Rangwala [2] applied HMM to the classification, which completed the sequence classification by combining Baum-Welch, Gibbs sampling and change function together. Hamada et al. [3] used a modified BP-AR-HMM algorithm to predict user's driving behavior under multi-time series. Mathew and Raposo [4] completed the prediction of user's next location through putting labeled triangle into HMM learning model.

H. Yuan et al. (Eds.): GRMSE 2016, Part II, CCIS 699, pp. 297–306, 2017.
DOI: 10.1007/978-981-10-3969-0_34

Through mapping the geographic information and service categories, the location-aware HMM (LHMM) is presented in this paper. This model gives the occurring probability of each service, along with the most likely occurred area. Then, the CF-Behavior prediction algorithm combining LHMM and CF is proposed. It both considers the location and personalization factor. What's more, it can reduce the dimensions of similarity calculation.

2 Behavioral Sequence Prediction

2.1 Behavioral Prediction and Recommendation Framework

As shown in Fig. 1, the whole recommendation framework is divided into three key modules: similar behavior cluster, series forecast and CF-behavior prediction.

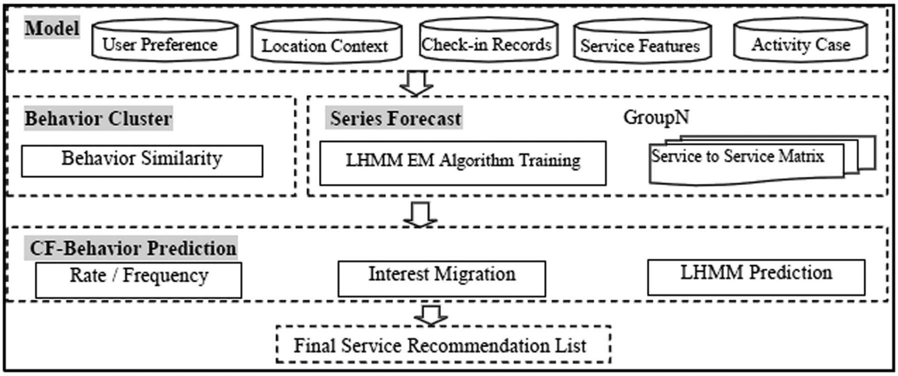

Fig. 1. The BP model to predict behavior and recommend services

(1) *Behavior cluster*. This module is design to group users who enjoy similar life rhythm together. It will generate the top-k users who have similar life pattern.
(2) *Series forecast*. This module trains similar user's check-ins to obtain initial probability matrix, transition probability matrix and emission probability matrix for LHMM model.
(3) *CF-behavior prediction*. This module provides more detailed recommendations, which combines the users' rate, visit frequency and interest migration.

2.2 Behavioral Sequence Model and Similarity Calculation

In order to reduce the sparseness calculate the transfer probability between two time states for different user groups, the check-in data is then to be divided into six different stages, namely {1–5, 6–10, 11–13, 14–16, 17–19, 20–24}.

Definition 1 (Check-in): Check-in (*CK*) indicates that a user *U* check in at shop *S* at a certain time *T*. User rates shop *S* with *R*, R ∈ {0, 10, 20, 30, 40, 50}.

CK = (userId, userName, user City, time, shopId, star, comment)

Definition 2 (Shop): Shop *S* represents the place where user participates in an activity with check-in.

Shop = (shopId, shopName, address, city, district, area, category, subcat, lat, lon)

Definition 3 (Score Function): *x* and *y* are two check-in record. σ(x, y) represents the output through function σ. The bonus points are designed as follow:

$$\sigma = \begin{cases} \sigma + 4x.district = y.district \\ \sigma + 2x.service = y.service \\ \sigma + 1x.area = y.area \\ \sigma - 3x \neq y \end{cases} \tag{1}$$

Definition 4 (Sequence Similarity): Given two sequences $S = S_1$ -> … -> $S\text{-}_n$, $T = T_1$ -> … -> $T\text{-}_n$. $|S|$ denotes the length of the sequence *S*. The similarity between two behavioral sequences is calculated by Eq. 2 as below:

$$\mathrm{Score} = \sum_{i=1}^{m} \sigma(S_i, T_i) \text{ where } m = |S| = |T| \tag{2}$$

Definition 5 (Behavior Similarity): $A(U_1, U2)$ shows the highest similarity after one-to-one sequences comparison between two users. Based on ClustalW [5] sequences match theory, the sequences similarity one-on-one between two different groups can be calculated.

3 Behavioral Prediction Based Collaborative Recommendation Algorithm

3.1 LHMM (Location-Aware Hidden Markov Model) Generation

A set of hidden states S = $\{S_1, S_2 \ldots S_M\}$ represents user's current location, along with a set of observations O = $\{O_1, O_2 \ldots O_N\}$ which represents the activities participated by the users. Figure 2 shows an illustration of relationship between hidden states and observations.

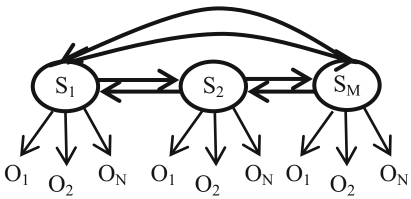

Fig. 2. The relationships between hidden states and observations of LHMM

There are three important parameters defined as follows: The initial probability matrix indicates the probability of each hidden state $S_i \in S$. Transition probability matrix indicates the probability from hidden state S_i to hidden state S_j. Emission probability matrix indicates the probability of observed $O_i \in O$ under a given state S_i.

Expectation-maximization algorithm (EM) can be used to search a set of LHMM parameters. Baum-Welch [6], also known as forward-backward algorithm, is the most widely used method for solving HMM learning. The basic idea is using a random initialization λ, assuming that this λ is the optimal solution.

What's more, in view of occurrence frequency of the sequence, we improve the Eqs. 8–10 listed in paper [7] with weight factor, so that periodic sequences can be better handled in new model. The improved formula is shown below.

$$L(i) = \sum_{O_k=O_1}^{O_n} \gamma_{i,O_k}(1) * C(O_k)\overline{\pi_i} = \frac{L(i)}{\sum_{i=1}^{|S|} L(i)} \tag{3}$$

$$M(i,j) = \sum_{O=O_1}^{O_n} \sum_{t=1}^{L-1} \varepsilon_{i,j,O_k}(t) * C(O_k)\overline{A_{i,j}} = \frac{M(i,j)}{\sum_{j=1}^{|S|} M(i,j)} \tag{4}$$

$$N(i,E_p) = \sum_{O=O_1}^{O_n} \sum_{t=1}^{L} \delta_{o_t,o_k}\gamma_i(t) * C(O_k)\overline{B_i(E_p)} = \frac{N(i,E_p)}{\sum_{E_p \in O} N(i,E_p)} \tag{5}$$

Where $C(O_k)$ represents the weight of observed sequence O_k. γ_i represents the probability of generating sequence Z under the state i. $\varepsilon_{i,j}$ shows the probability of generating sequence Z during the transition from state i to state j. The original equation multiplied by the weight $C(O_k)$, L, M, N formula is obtained which help specify the certain occurred frequency of sequences.

3.2 Next Service and Location Prediction

After the three important parameters of LHMM are generated, it is time to predict user's future possible behavior. Assuming that $CK = \{O_1, O_2 \ldots O_t\}$ is a check-in activity sequence, where O_i represents the check-in activity at time i, corresponding to the check-in place sequence $S = \{S_1, S_2 \ldots S_t\}$. Now, in order to derive the activity O_{t+1} at time $t+1$, it can analyze the most probable hidden state S_{t+1} at time $t+1$ by applying Eqs. 6–7 in this paper.

$$p(O_{t+1}|O_{1..t}) = \sum_{S_{t+1}} p(O_{t+1}|S_{t+1}) \sum_{S_t} p(S_{t+1}, S_t|O_{1..t}) \tag{6}$$

$$\sum_{S_t} p(S_{t+1}, S_t|O_{1..t}) = \frac{1}{\sum_{S_t} \alpha(S_t)} \sum_{S_t} p(S_{t+1}|S_t)\alpha(S_t) \tag{7}$$

The $p(O_{t+1}|S_{t+1})$ represents the occurrence probability of observation O at time $t+1$ under given hidden state S_{t+1}. Moreover, $p(S_{t+1}|S_t)$ shows the state transition probability from time t to time $t + 1$. $\alpha(S_t)$ represents the probability of observing $O_{1 \ldots t}$ at time t under hidden state S.

Each calculated pair $<S_i, O_j>$ forms an array of service-location probability. By sorting this array, the most probable activity and its corresponding location will be easily worked out with the LHMM which user may take in the next period.

3.3 CF-Behavior Prediction Algorithm (LHCF)

If it is just stopped at previous module, the recommendations within a group are the same result. In fact, although users share similar routine of the day, it doesn't imply that they enjoy similar interests. For instance, users *A* and *B* usually have lunch during 11:00–13:00 at District *X*. User *A* prefers restaurant *a* while user *B* prefers *b*. To solve this problem, the CF-Behavior prediction based on LHMM model (LHCF) is proposed.

Combining with the output from the previous sections, the large matrix can be easily divided into several sub-matrixes at space (S_i) and category (C_i) level, shown as Fig. 3.

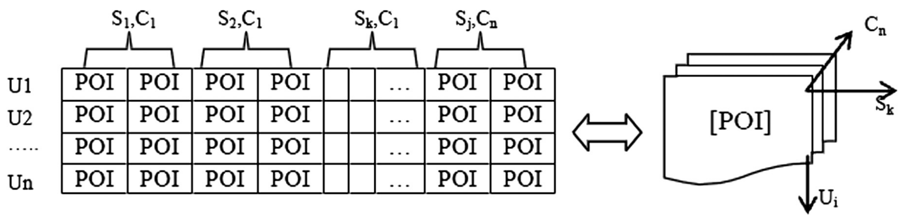

Fig. 3. Divide dataset into subset by space and category information

According to the above dataset, user's preferences can be affected by three important factors: score, visiting frequency and the time user visited. Therefore, a time transfer function is added into final formula, so that users can be clustered in a more proper way. Equations 8 and 9 are the definition of user's point of interest (POI):

$$\mathrm{POI}(\mathrm{U_i}, \mathrm{S_j}) = \left(\mathrm{a}*\frac{\mathrm{avg(score)}}{\mathrm{maxscore}} + \mathrm{b}*\frac{\mathrm{count}(\mathrm{s_j})}{\sum_{\mathrm{s_k} \in \mathrm{S}} \mathrm{count}(\mathrm{s_k})}\right) * \mathrm{t}(\mathrm{U_i}, \mathrm{S_j}) \tag{8}$$

$$\mathrm{t}(\mathrm{U_i}, \mathrm{S_j}) = 1/\left(\frac{\mathrm{currenDate} - \mathrm{maxDate}}{7}\right) \tag{9}$$

Where t function is an interest migration function, the more frequently a user visit a shop, the higher score it will be. Attributes a and b are two fit parameters in order to calculate the POI in a more flexible way. For check in records without user's rate, an average score will be assigned according to user's historical records.

The cosine similarity calculation formula is applied to calculate the similarity between the different mobile users.

$$\mathrm{Sim}(U_i, U_j) = \frac{\sum_{S_k \in S}(\mathrm{POI}(U_i, S_k) - \overline{\mathrm{POI}(U_i)})(\mathrm{POI}(U_j, S_k) - \overline{\mathrm{POI}(U_j)})}{\sqrt{(\mathrm{POI}(U_i, S_k) - \overline{\mathrm{POI}(U_i)})^2}\sqrt{(\mathrm{POI}(U_j, S_k) - \overline{\mathrm{POI}(U_j)})^2}} \tag{10}$$

$\mathrm{POI}(U_i, S_k)$ means the points of interest of user *i* for service *k*. $\overline{\mathrm{POI}(U_i)}$ presents the average points of interest the user *i* for all service categories.

Algorithm *CF-Behavior prediction*

```
Input: User: U, prevObservList<location,activity>, HMM parameters
Output: ProbList<location,activity>, RecommendList
Algorithm:
pastObserv = buildPastObserv(prevObservList,time) //predict t+1 service through <O1…Ot>
stateList = getSortedProbHiddenState(pastObserv); // probability calculation of next state
for stateI in stateList do // For each predicted state, calculating the probability of next observation sequence
observNext<<state,observe>,prob> = insertAndCalculateProb(pastObserv,stateI);
getTop5Prob(observNext);// Probability values are sorted and selected the top five combinations for <region,activity> in <state,observe> do// Iteratively predicted next combination
  for userU in users do
    for CkR in userU.checkinRecords do
if userU.checkinRecord.region = region and checkinRecord.categories = activity
POI[userU][userU.CkR.shop] = calculatePOI();
UserSim = calculateCosSim(POI) // Calculate the cosine similarity
topSimUserList[] = topUserSim(userI,5) // Get nearest 5 users
RecommendList<region,activity> = findHigherPOIShop(topSimUserList[])
// From the similar users group, find out recommendations users might like shops
```

Based on the service sequences which users have participated in, a numeric probability list is calculated, indicating the likelihood of each possible service and its corresponding occurred places. After the most possible service is determined, the cosine similarity between two users is calculated based on user's historical rating behavior, visit frequency and interest shift. Finally the recommendation list is gotten according to the improved CF-behavior prediction algorithm.

4 Experimental Evaluation

4.1 Data Analysis

Dianping website (http://www.dianping.com) is a famous leading third-party website that provides detail business information, consumer reviews and other O2O trading services. Through its open API, our dataset has nearly 6000 shops in Shanghai, along with 60,000 check-ins records from 3000 distinct users from 2010 to 2014, shown as Fig. 4.

The 5-fold cross-validation method is used to in this paper. The check-ins is divided into 5 subsets. Every time, one of the 5 subsets is used as the test set S_{test} and the other 4subsets are put together to form training set $S_{training}$. The detail is as follow:

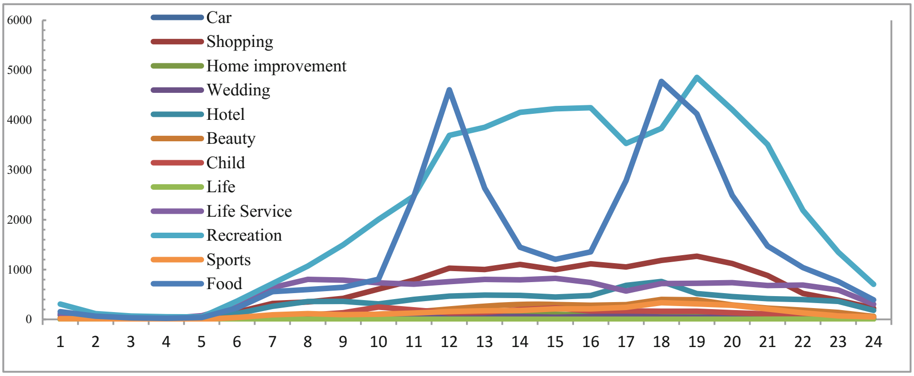

Fig. 4. Different services check-in frequency chart

4.2 Behavioral Prediction Evaluation

The purpose of this experiment is to calculate the prediction accuracy on TopK-LHMM algorithm compared with other traditional methods.

To demonstrate the prediction accuracy of the algorithm, the concept of N-hit is introduced. If the biggest probability value is the one user participates in reality, so this scenario is defined as 1-Hit. All in all, N-hit means that the *n*th value in the prediction array is match with the next step that happens in reality. Obviously, the smaller *n* is, the higher accuracy the algorithm indicates (Table 1).

Table 1. Behavioral sequence similarity calculation

User Check-In	A:-,-,100102, 100102,-,-	A:-,-,100105,-,100105,090103
B:-,-,020104,-,020204,-	2(phase3) = 2	2(phase3)−3(phase5) = -1
B:-,-,-,020203,-,-	−3(phase4) = −3	0
C:-,-,-,100110,130110,-	6(phase4) = 6	2(phase5) = 2
C:-,-,-,100203,100105 ,09020	4(phase4) = 6	7(phase5) + 5(phase6) = 12

Figure 5 shows the experiment result of different nearest *k* value. The performance is very close when *k* is 5 or 10, but when *k* is bigger than 10, the prediction of users' next action will drop. The user behavior is similar, not exactly the same. Expanding the number of similar users will reduce the accuracy of the model prediction. Thus, we assign *k* as 10 in follow-up experiments.

As the Fig. 6 shows, the accuracy of action prediction for the first two hit is over 50%, which has better impact than traditional HMM. If Top-K feature adds into LHMM, the new model helps improve additional 5% better accuracy on the 1-hit to 3-hit in average.

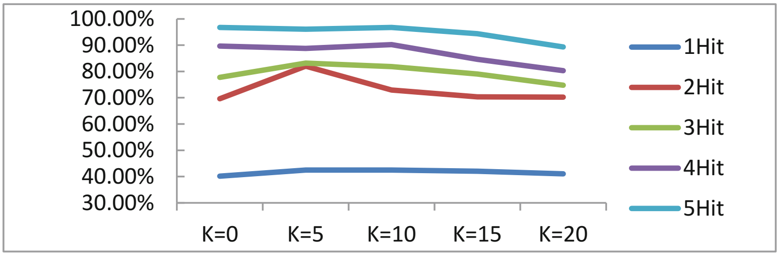

Fig. 5. Experiment on nearest K value

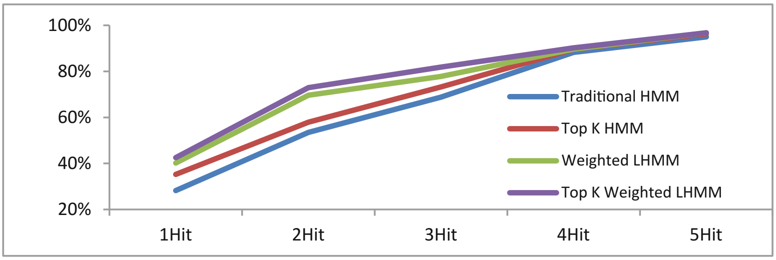

Fig. 6. Behavioral prediction comparison

4.3 Interests Recommendation Evaluation

Different from traditional collaborative filtering, the LHCF use the output <service, area, probability> from LHMM. For evaluating the recommendation algorithms, we compare ours with the following five state-of-the-arts recommendation methods. Location-based Collaborative Filtering (LCF), User interest and Proximity (UP) [8], User interest and geographical influences (UG) [9], Spatio-Temporal Collaborative Filtering (STCF) [10], Location-aware Hidden Markov Model (LHMM):

It is the evaluation indicators *hit@N* and strategies in papers [11, 12] that we use to evaluate the effectiveness of different recommendation algorithms.

$$hit@N = \frac{|S_{success}|}{|S_{test}|} \tag{11}$$

The N represents the number of recommended services. The $S_{success}$ represents the number of success in S_{test}. The S_{test} is a test case set. As for an individual test case $(u, s, l) \in S_{test}$, the u represents user, s represents service, l represents location.

Firstly, we simulate user's current temporal and spatial properties, which are close to the test check-in. Secondly, different algorithm works out its Top-N recommendation list. Finally, if a Top-N recommendation list includes the testing service, it is successful and the number of $S_{success}$ increases one. The *hit@N* can reflect the quality of recommendation algorithm.

Figure 7 shows the performance of each algorithm. The experiment shows that LHCF algorithm is greatly superior to the others, where it takes time, space and user interest into consideration.

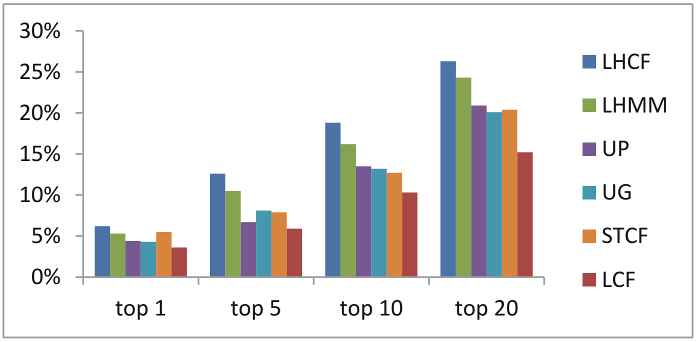

Fig. 7. Comparison experiments

5 Conclusions

In this paper, a user behavior prediction and recommendation framework for location based services is proposed under mobile Internet environment. Based on the users' activity behavior sequences clustering module for location aware mobile services, a Top K-LHMM algorithm is proposed and implemented to do a better prediction for different kind of services and regions under a certain user's status. Under the extensive experiments designed in this paper, the improved system gives us more accurate results. Especially it overcomes the weakness of perception of location and time context in traditional collaborative filtering algorithm and obtains a high efficiency on mobile service prediction. The improved system has strong scalability to adapt to different services recommendation environment. In the future, our research work is mainly focused on how to extend the framework in several directions.

Acknowledgments. This work is partially supported by National Natural Science Foundation of China (61074135, 61303096, 71101086) and Shanghai Leading Academic Discipline Project (J50103). We also would like to show our great appreciations to all of our hard working fellows in the projects above.

References

1. Natarajan, N., Shin, D.: Which app will you use next: collaborative filtering with interactional context. In: Proceedings of the 7th ACM Conference on Recommender Systems, pp. 201–208 (2013)
2. Blasiak, S., Rangwala, H.: A hidden Markov model variant for sequence classification. In: IJCAI 2011, Proceedings of the International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, pp. 1192–1197. DBLP, July 2011

3. Hamada, R., Kubo, T., Ikeda, K.: Towards prediction of driving behavior via basic pattern discovery with BP-AR-HMM. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2805–2809 (2013)
4. Mathew, W., Raposo, R.: Predicting future locations with hidden Markov models. In: ACM Conference on Ubiquitous Computing, pp. 911–918. ACM (2012)
5. Thompson, J.D., Gibson, T.J.: Multiple sequence alignment using ClustalW and ClustalX. Curr. Protoc. Bioinf (2002). UNIT 2.3
6. Welch, L.R.: Hidden Markov models and the Baum-Welch algorithm. IEEE Inf. Theory Soc. Newslett. **53**(4), 10–13 (2003)
7. Zukerman, I., Albrecht, D.,W., Nicholson, A.,E.: Predicting users' requests on the WWW. In: Kay, J. (ed.) UM99 User Modeling. CICMS, vol. 407, pp. 275–284. Springer, Heidelberg (1999). doi:10.1007/978-3-7091-2490-1_27
8. Ference, G., Ye, M., Lee, W.C.: Location recommendation for out-of-town users in location-based social networks. In: ACM International Conference on Information and Knowledge Management, pp. 721–726. ACM (2013)
9. Ye, M., Yin, P., Lee, W.C., et al.: Exploiting geographical influence for collaborative point-of-interest recommendation. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 325–334. ACM (2011)
10. Yuan, Q., Cong, G., Ma, Z., et al.: Time-aware point-of-interest recommendation. In: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 363–372. ACM (2013)
11. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, pp. 426–434, August 2008
12. Yin, H., Cui, B., Li, J., et al.: Challenging the Long Tail Recommendation. Proc. VLDB Endow. **5**(9), 896–907 (2012)