Website-oriented recommendation based on heat spreading and tag-aware collaborative filtering

Zi-Ke Zhang [a,b,*], Lu Yu [a,b], Kuan Fang [c], Zhi-Qiang You [a,b], Chuang Liu [a,b], Hao Liu [b,d], Xiao-Yong Yan [e]

[a] *Institute of Information Economy, Hangzhou Normal University, Hangzhou 311121, PR China*

[b] *Alibaba Research Center for Complexity Sciences, Hangzhou Normal University, Hangzhou 311121, PR China*

[c] *Web Sciences Center, University of Electronic Science and Technology of China, Chengdu 610054, PR China*

[d] *Department of Physics, University of Fribourg, Chemin du Musée 3, Fribourg 1700, Switzerland*

[e] *School of Systems Science, Beijing Normal University, Beijing 100875, PR China*

HIGHLIGHTS

- Discuss the mining webs with information extracted from search and browser logs.
- Propose the tag-aware KNN CF via SVD-based websites' keywords technique.
- Build a hybrid algorithm to provide accurate recommendations.

Recently, *Recommender Systems* has been widely applied in helping users find potentially interesting items from the era of big data. However, most of researches on this topic have focused on estimating the direct relationships between users and items, neglecting other available information. In this paper, we discuss about mining webs with information extracted from search and browser logs of users. In particular, we utilize the keywords correlated with corresponding websites by *Singular Value Decomposition* (SVD) technique to model users features and propose the tag-aware *k*-nearest neighbor *Collaborative Filtering* (CF). We then build a hybrid recommendation method to help people accurately find websites by employing *Heat Spreading* (HeatS) method. Experimental results demonstrate that the hybrid method outperforms baseline algorithms at the global ranking metric.

1. Introduction

With the development and popularization of Internet, online searching becomes one of the most commonly used approaches to obtain desired information. Thus, billions of *search query data* have been accumulated in the web servers [1]. Besides search data, browser plugins also record the *browser log* information. However, it still remains a big challenge to find efficient methods to model such massive amounts of data. In this paper, we study the challenges of mining the value of web search and browser logs with the recommendation technology. *Recommender Systems* [2] can automatically predict user preference and is becoming an essential part of various applications to provide personalized service. For instance, *Amazon.com* proposes item-to-item collaborative filtering system [3] to provide personalized book service; *Google* also builds its

News Recommender System by using user's click behaviors [4]; *Baidu*, the biggest search engine in China, pays much attention to researching recommender system for its Q&A community [5], and so on.

In the consequent years, *Collaborative Filtering* (*CF*) [6] is the mainstream method to mine user preference according to the historical behaviors. Zeng et al. [7] found that the top recommended items of CF were always with large degree, that is to say, CF tends to recommend popular items to users. Consequently, CF may give a low rank to the *majority* of items, but with few attention, that user may essentially prefer. To overcome the weakness of CF, various methods are proposed. Zhou et al. [8] provided a hybrid method to solve the accuracy–diversity dilemma by combining *probabilistic spreading* (*ProbS*) and *heat spreading* (*HeatS*) [9–12]. Zhang et al. [13–15] analyzed the impact of tag information [16] and raised a user–item–tag tripartite graph recommendation method, which not only improves on accuracy–diversity but also reports on a promising approach to solve the cold-start problem. Qiu et al. [17,18] presented that the item-based heterogeneous effect could significantly enhance the recommendation performance. Yang et al. pointed out that the anchoring bias effect of online voting behaviors could largely improve the recommendation performance [19]. Recently, many researchers used tag information to improve the performance of recommender systems, such as, personalized recommendation in social tagging systems [20], tagging-aware CF [21], hybrid content and tag-based profiles recommender system [22], and so on.

Though recommender systems have succeeded in various applications [3–5], it is still hard to design efficient recommendation models to handle log data because the web search and browser logs do not show the preference of users explicitly. Alternatively, webpages provide plenty of meta information to describe their contents, and the third-party social tagging systems (e.g., del.icio.us[1]) provide a flexible folksonomy of tags for different websites. Thus, the log data may provide us a possible access to analyze the users' features by extracting the tag information of webpages.

In this paper, we aim at constructing a recommender system to push websites that users might be interested in. Firstly, we use a tag-based k-Nearest-Neighbor (kNN) CF with the tag information of webpages. The experimental results indicate that, compared with user-based CF, tag-based CF improves in accuracy. In addition, we propose a new approach to combine heat spreading [8] with the tag-based CF, and improve the accuracy of recommendation.

2. Method

In the present model, there are three kinds of relations, $(U_{n \times m}, O_{m \times r}, T_{n \times r})$, n, m and r represent the number of users, items and tags, respectively. $U_{n \times m}$ is the user–item matrix, $U_{i\alpha} = 1$ if user i collects item α, otherwise $U_{i\alpha} = 0$. $O_{m \times r}$ is the item–tag matrix, where $O_{\alpha t} = 1$ if item α contains tag t, otherwise $O_{\alpha t} = 0$. In order to measure the correlations between users and tags, we firstly construct the user–tag matrix $D_{n \times r}$:

$$D = U \times O. \tag{1}$$

Subsequently, we employ a term-wighting approach, so-called TF–IDF [23,24], to measure the preference of an arbitrary user for different tags. We then reconstruct the matrix D to obtain another user–tag matrix $T_{n \times r}$. Here the *term frequency* (TF) measures the degree of user's preference for each tag and the *inverse document frequency* (IDF) reflects whether the tag is common or rare for users.

In this paper, we employ two types of recommendation methods, tag-based collaborative filtering and heat spreading. In the area of recommender system, generally, CF is classified into two types, Memory-based and Model-based methods. Memory-based CF mainly includes user- or item-based [3,24] CF. According to Ref. [25], Model-based CF contains various types, such as matrix factorization (MF) [26], Bayesian network [27], and so on. Here, we mainly discuss the difference between the traditional user-based CF and the variant of model-based CF based on matrix factorization.

2.1. User-based k-Nearest-Neighbor (kNN) collaborative filtering

In the user-based CF, we assume that we can predict the scores of uncollected items for the target user based on the preferences of her/his neighbors, which are defined as a group of people with similar interests. The most common approach is to find the k Nearest Neighbors (kNN) [28,29] to produce recommendation for the target user. Basically, the similarity s_{ij} between user i and j is measured by cosine similarity [8]:

$$s_{ij} = \frac{\sum_{\alpha=1}^{m} U_{i\alpha} U_{j\alpha}}{\sqrt{k_i k_j}}, \tag{2}$$

where $U_{i\alpha}$ or $U_{j\alpha}$ denotes whether user i or j collects item α, k_i and k_j respectively denote the number of items simultaneously collected by user i and j. The predicted score of uncollected item α for the target user i is generated as [8]:

$$r_{\alpha i} = \frac{\sum_{j=1}^{k} s_{ij} U_{j\alpha}}{\sum_{j=1}^{k} s_{ij}}. \tag{3}$$

---

[1] http://del.icio.us/.

The final recommendation list of target user i is generated by ranking the scores of the uncollected items in descending order.

2.2. Tag-based k-Nearest-Neighbor (kNN) collaborative filtering

In the area of *Information retrieval* (IR) [30], one document is denoted as a set of tokens, which could reduce the computational cost and improve the efficiency of searching the documents related to a query. Each token has its own latent semantic feature [31,32]. The weight of tokens in each document is determined by TF–IDF [24] in *Latent Semantic Indexing* (LSI) [31]. In the present model, Tags also have their own latent semantic information. Inspired by this, we propose the tag-based kNN CF and build a user–tag matrix $T_{n \times r}$, based on the tag information of each item. Each entry of the matrix weighs the preference degree of the user to a tag. However, for the large size of $T_{n \times r}$, which is usually with millions of entries, it is hard to find the latent semantic information. Deerwester et al. [31] discussed the implementation of LSI by *Singular Value Decomposition* (SVD). It not only reduces the dimensions and the computation, but also detects the latent semantic information in the same topic model [33]. Refs. [26,29,34,35] also extended the approaches of applying SVD to build collaborative filtering system.

In this paper, we intend to represent user-interests in the low-rank latent feature space. Thereby, we employ the SVD method to decompose the user–tag matrix T into two feature matrices: $\hat{T} = P_{n \times s} \times Q_{r \times s}$, where s is the number of topics. Each row of P stands for user's feature vector p_u and each row of Q stands for tag's feature vector q_i. Each dimension of the user or tag feature vector corresponds with the same topic. The similarity between user i and j is computed by [24]:

$$s_{ij} = \cos\left(\overrightarrow{p_u^{(i)}}, \overrightarrow{p_u^{(j)}}\right) = \frac{\left(\overrightarrow{p_u^{(i)}}, \overrightarrow{p_u^{(j)}}\right)}{\left\|\overrightarrow{p_u^{(i)}}\right\|_2 * \left\|\overrightarrow{p_u^{(j)}}\right\|_2}, \tag{4}$$

where $p_u^{(i)}$ and $p_u^{(j)}$ respectively denote the feature vector of user i and j. We then predict the uncollected items' score for the target user i with Eq. (3).

2.3. Heat spreading

Heat Spreading (HeatS) [8] could generate novel recommendation list without other information but only the user-object bipartite network. HeatS works by assigning an initial resource to the objects connected to the target user, which is denoted as the vector f (f_β represents the resource on object β). Then, resources are distributed to different objects via the transformation of $\overrightarrow{f} = W^H f$ as following:

$$W_{\alpha\beta}^H = \frac{1}{k_\alpha} \sum_{j=1}^{n} \frac{U_{j\alpha} U_{j\beta}}{k_j}, \tag{5}$$

where $W_{\alpha\beta}^H$ describes how many resources item α obtains from item β. k_j and k_α respectively represent the degree of the user j and the item α. $\overrightarrow{f}$ is a column vector, and each dimension is the predicted score for the corresponding item. We then sort items by the descending order of scores. The higher the rank is, the more likely the corresponding item will be recommended.

2.4. Hybrid method

Many pioneering research in this area have been done to investigate the effect of combining a few recommender algorithms to comprise a hybrid recommender system [36–38]. Experimental results indicate that tag-based kNN CF method has a better performance in accuracy than user-based kNN CF while the heat spreading method favors the diversity but not good in accuracy. Therefore, we combine the tag-based CF method with HeatS method in a linear form. We define the score of uncollected item i of the target user generated by method X and Y as X_i and Y_i, respectively. Then, the final score of object i, r_i, is measured by:

$$r_i = \omega \cdot X_i + (1 - \omega) \cdot Y_i, \quad \omega \in [0, 1]. \tag{6}$$

3. Data & metrics

3.1. Data

The data set is distributed from the competition hosted by CNNIC[2] and a data sharing system Datatang.[3] Raw data from CNNIC consists of 1000 users' personal behaviors, including browser log data, search data and software behavior records. The

---

[2] CNNIC stands for China Internet Network Information Center. Information about the competition can get from http://cnnicdata.datatang.com/.

[3] http://www.datatang.com/.

Table 1

Performance of three methods. The k of each CF method is set as 50.

Method	RS	Precision	Novelty
User-based kNN CF	0.1574	**0.1052**	2.389
Tag-based kNN CF	**0.1323**	0.105	2.3885
Heat spreading	0.2443	0.00041	**9.8677**

data were collected between 2012-05-07 and 2012-08-12. Here, we focus on studying the behaviors of visiting the website. In the data set, url structures that are provided by users are diverse. It is essential to filter out the unusable information. After preprocessing the raw data, we obtain a data set R, which contains 1000 users, 38,242 websites, 117,742 records and the time users visit the websites.

As introduced in the Section 2.2, tag-based CF method needs additional information to construct user–tag matrix. Web designers usually carefully polish the words contained in the meta information making sure their web sites rank higher in the search results. So retrieving the meta information of those webpages could help us capture the user interests, and then analyze the relationship between users. Therefore, we crawl the meta information of websites before implementing the *tokenization* [39] to obtain the keywords or tags for each website.

To evaluate the present method, The data set is divided into two parts, 20% of which are randomly chosen as the testing set, and the rest are regarded as training set.

3.2. Metrics

We utilize three metrics to evaluate the quality of the proposed methods.

(1) *RankingScore* (RS) [8]: For a good algorithm, we assume that the preferable items will rank higher than the uncollected ones. For a certain item α linking to the arbitrary user i in the test set with rank r in the recommendation list, its RS value is calculated as:

$$RS_{i\alpha} = \frac{r}{m - k_i}, \tag{7}$$

where m is the number of items and k_i presents the number of items collected by user i. Averaging all the RS of items in the testing set, we obtain the RS of the observed algorithm. The smaller RS is, the better accuracy of the algorithm will be.

(2) *Precision* [40]: Precision is defined as the proportion of truly liked items contained in the Top-L places of the recommendation list for the user i:

$$P = \frac{\sum_{i=1}^{n} h_i}{nL}, \tag{8}$$

where n is the number of users who appear both in the testing set and training set, L is the length of recommendation list, and h_i is the number of items recovered in top-L recommendation list. By averaging over all individuals', we obtain the precision of overall recommender system.

(3) *Novelty* [8]: Given an item O_α, the degree of its novelty for the target user is denoted as the self-information, is:

$$I_\alpha = \log_2 \left(\frac{n}{k_\alpha} \right), \tag{9}$$

where n is the number of users. We then compute the self-information of the top-L items and average them to get the novelty for the target user. Finally, we average all the users' top-L items' self-information to obtain the overall top-L novelty. This metric measures alternative capability of the recommender systems of recommending less popular items. Thereby, the recommendation method with larger value of Novelty could bring the recommender systems with more serendipity.

4. Results

In order to evaluate the performance of presented algorithm, we conduct several experiments to assess three aforementioned metrics.

The results for three algorithms are showed in Table 1. The number of nearest neighbors k for each user is set to 50. Both user- and tag-based CF have better accuracy than Heat Spreading, but worse in novelty. The precision of tag-based kNN CF and user-based kNN CF are very close to each other. However, tag-based CF outperforms user-based CF in RS, which suggests that tag-based CF has higher ability to recommend preferable item to the target user.

Fig. 1 shows that different values of k (the length of nearest neighbor list) have an impact on the performance of CF. The two types of CF have similar performance for precision and novelty. However, user-based CF has much worse performance at the RS than tag-based CF with different k value.
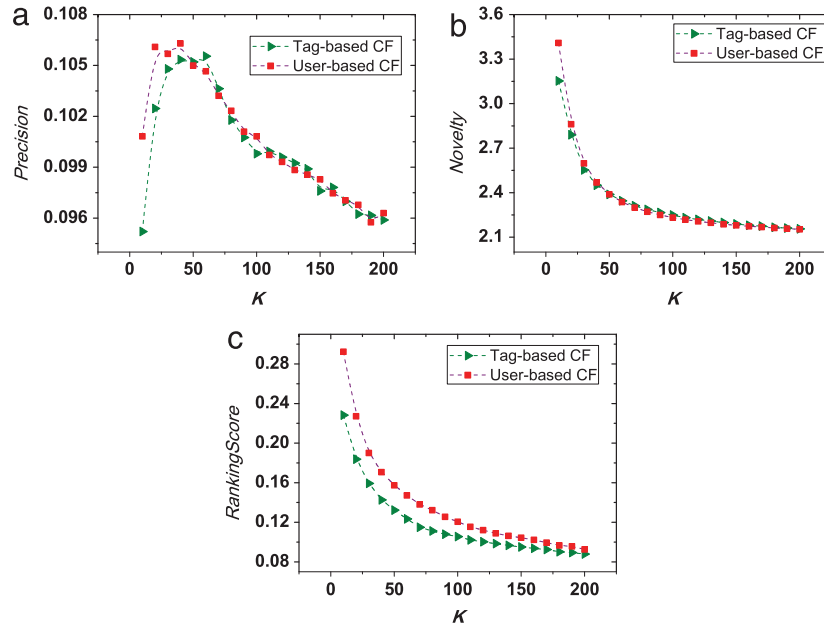
Fig. 1. Comparison between user-based kNN CF and tag-based kNN CF: (a) Precision, (b) Novelty and (c) RankingScore. k ranges from 10 to 200 with the step of 10.
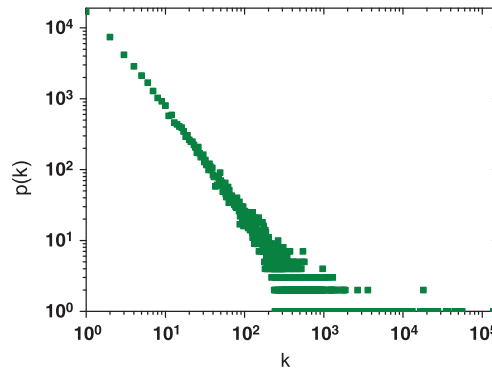
Fig. 2. Distribution of the click times of websites.

Fig. 2 shows that most websites are visited by only a few users, in other words, only a few websites are frequently visited. Therefore, we need an efficient method to help users find interested websites with low attention. Consequently, we combine tag-based CF with HeatS by linear mixture. For the tunable parameter ω, when $\omega = 1$, the hybrid method degenerates to the HeatS method and obtains the optimal novelty. When $\omega = 0$, it degenerates to the pure tag-based CF. Fig. 3 shows that, though the hybrid system does not have a obvious improvement in precision or novelty, the RS is significantly improved by about 25% at $\omega = 0.05$. Considering the heterogeneous property and mass number of webpages online, this overall improvement can help users put more attention to relatively low-attention websites.

5. Conclusions and discussion

Researchers from various fields have discussed how to mine the essential value of web [41,42] by personalized recommender systems and achieved good performances [43–45]. In this paper, we discuss about mining webs with information extracted from search and browser logs. Firstly, we utilize the keywords correlated with corresponding websites to model user's features and propose the tag-based kNN CF. Secondly, we build a hybrid recommender method to help people accurately find websites by additional employing HeatS method. Experimental results demonstrate that the hybrid method outperforms both baseline algorithms at the global ranking metric.

This paper provides a simple method to explore the value of online access records. However, the raw data sets may also include demographics records, such as age, job, education, and so on. Our future work will try to improve our method of keyword extraction to get a higher quality of the description of the webpages. In addition, we shall also explore the value of demographics to improve the performance of the recommender system both on accuracy and diversity.
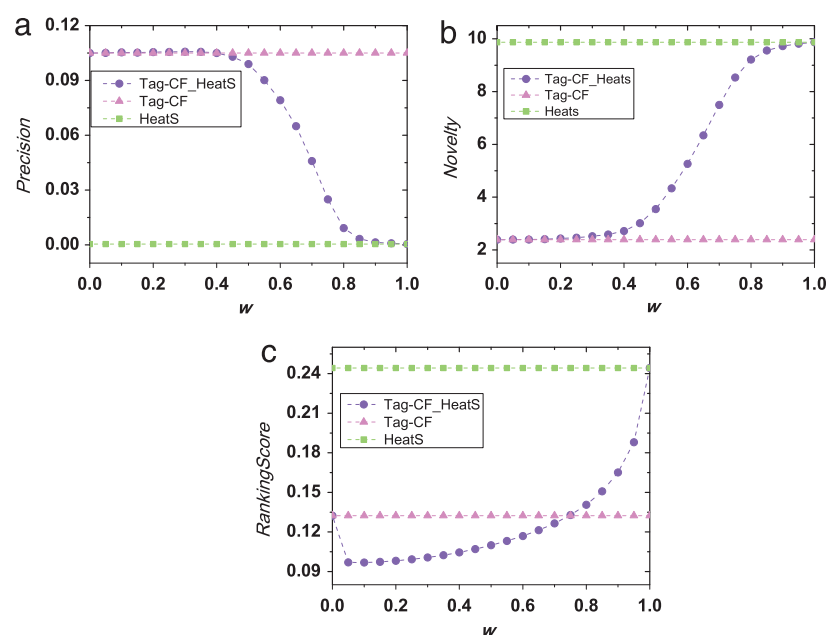
Fig. 3. Performance of combining the HeatS with Tag-based kNN CF on (a) Precision, (b) Novelty, and (c) RankingScore.

References

[1] D. Jiang, J. Pei, H. Li, Mining search and browse logs for web search: a survey, ACM Trans. Intelligent Systems Technology 4 (4) (2013) 1–42.
[2] L. Lü, M. Medo, C.H. Yeung, Y.-C. Zhang, Z.-K. Zhang, T. Zhou, Recommender systems, Phys. Rep. 519 (1) (2012) 1–49.
[3] G. Linden, B. Smith, J. York, Amazon. com recommendations: item-to-item collaborative filtering, Internet Computing, IEEE 7 (1) (2003) 76–80.
[4] A.S. Das, M. Datar, A. Garg, S. Rajaram, Google news personalization: scalable online collaborative filtering, in: Proceedings of the 16th International Conference on World Wide Web, 2007, pp. 271–280.
[5] Q. Liu, T. Chen, J. Cai, D. Yu, Enlister: baidu's recommender system for the biggest chinese Q&A website, in: Proceedings of the Sixth ACM Conference on Recommender Systems, 2012, pp. 285–288.
[6] J. Herlocker, J. Konstan, J. Riedl, Explaining collaborative filtering recommendations, in: Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work, 2000, pp. 241–250.
[7] W. Zeng, M.S. Shang, Q.M. Zhang, L.Y. LÜ, T. Zhou, Can dissimilar users contribute to accuracy and diversity of personalized recommendation? Internat. J. Modern Phys. C 21 (10) (2010) 1217–1227.
[8] T. Zhou, Z. Kuscsik, J.-G. Liu, M. Medo, J.R. Wakeling, Y.-C. Zhang, Solving the apparent diversity-accuracy dilemma of recommender systems, Proc. Natl. Acad. Sci. USA 107 (10) (2010) 4511–4515.
[9] J.-G. Liu, T. Zhou, H.-A. Che, B.-H. Wang, Y.-C. Zhang, Guo. Effects of high-order correlations on personalized recommendations for bipartite networks, Physica A 389 (4) (2010) 881–888.
[10] J.-G. Liu, T. Zhou, Q. Guo, Information filtering via biased heat conduction, Phys. Rev. E 84 (3) (2011) 037101.
[11] J.-G. Liu, K. Shi, Q. Guo, Solving the accuracy-diversity dilemma via directed random walks, Phys. Rev. E 85 (1) (2012) 016118.
[12] J.-G. Liu, T. Zhou, Q. Guo, Heat conduction information filtering via local information of bipartite networks, Eur. Phys. J. B 85 (8) (2012) 286.
[13] Z.-K. Zhang, C. Liu, Y.-C. Zhang, T. Zhou, Solving the cold-start problem in recommender systems with social tags, EPL 92 (2) (2010) 28002.
[14] Z.-K. Zhang, T. Zhou, Y.-C. Zhang, Personalized recommendation via integrated diffusion on user-item-tag tripartite graphs, Physica A 389 (1) (2010) 179–186.
[15] Z.-K. Zhang, T. Zhou, Y.-C. Zhang, Tag-aware recommender systems: a state-of-the-art survey, J. Comput. Sci. Tech. 26 (5) (2011) 767–777.
[16] Z.-K. Zhang, L. Lü, J.-G. Liu, T. Zhou, Empirical analysis on a keyword-based semantic system, Eur. Phys. J. B 55 (4) (2008) 557–561.
[17] T. Qiu, G. Chen, Z.-K. Zhang, An item-oriented recommendation algorithm on cold-start problem, EPL 389 (5) (2011) 58003.
[18] T. Qiu, Z.-K. Zhang, G. Chen, Information filtering via a scaling-based function, PLoS ONE 8 (5) (2013) e63531.
[19] Z. Yang, Z.-K. Zhang, T. Zhou, Anchoring bias in online voting, EPL 100 (6) (2012) 68002.
[20] A. Shepitsen, J. Gemmell, B. Mobasher, R. Burke, Personalized recommendation in social tagging systems using hierarchical clustering, in: Proceedings of the 2008 ACM Conference on Recommender Systems, 2008, pp. 259–266.
[21] K.H.L. Tso-Sutter, L.B. Marinho, L. Schmidt-Thieme, Tag-aware recommender systems by fusion of collaborative filtering algorithms, in: Proceedings of the 2008 ACM Symposium on Applied Computing, 2008, pp. 1995–1999.
[22] D. Godoy, A. Amandi, Hybrid content and tag-based profiles for recommendation in collaborative tagging systems, in: Latin American Web Conference, LA-WEB'08. IEEE, 2008, pp. 58–65.
[23] G. Salton, C. Buckley, Term-weighting approaches in automatic text retrieval, Inform. Process. Management 24 (5) (1988) 513–523.
[24] B. Sarwar, G. Karypis, J. Konstan, J. Reidl, Item-based collaborative filtering recommendation algorithms, in: Proceedings of the 10th International Conference on World Wide Web, 2001, pp. 285–295.

[25] X. Su, T. Khoshgoftaar, A survey of collaborative filtering techniques, Adv. Artif. Intellig. 4 (2009) (2009).
[26] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, IEEE Computer 42 (8) (2009) 30–37.
[27] X. Su, T.M. Khoshgoftaar, Collaborative filtering for multi-class data using belief nets algorithms, in: 18th IEEE International Conference on Tools with Artificial Intelligence. IEEE, ICTAI'06, 2006, pp. 497–504.
[28] J. Gemmell, T. Schimoler, M. Ramezani, B. Mobasher, Adapting K-nearest neighbor for tag recommendation in folksonomies, in: Intelligent Techniques for Web Personalization & Recommender Systems, 2009.
[29] G. Takcs, I. Pilszy, B. Nmeth, Matrix factorization and neighbor based algorithms for the netflix prize problem, in: Proceedings of the 2008 ACM Conference on Recommender Systems, 2008, pp. 267–274.
[30] T. Landauera, P. Foltzb, D. Lahamc, An introduction to latent semantic analysis, Discourse Processes 25 (2–3) (1998) 259–284.
[31] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, R. Harshman, Indexing by latent semantic analysis, J. Am. Soc. Inf. Sci. 41 (6) (1990) 391–407.
[32] T. Hofmann, Latent semantic models for collaborative filtering, ACM Trans. Inf. Syst. (TOIS) 22 (1) (2004) 89–115.
[33] D. Blei, J. Lafferty, Dynamic topic models, in: Proceedings of the 23rd International Conference on Machine Learning, 2006, pp. 113–120.
[34] S. Zhang, W.-H. Wang, J. Ford, F. Makedon, Using singular value decomposition approximation for collaborative filtering, in: E-commerce Technology 2005. CEC 2005. Seventh IEEE International Conference, 2005, pp. 257–264.
[35] S. Rendle, Factorization machines, in: Data Mining (ICDM), 2010 IEEE 10th International Conference on. IEEE, 2010, pp. 995–1000.
[36] M. Pazzani, A framework for collaborative, content-based and demographic filtering, Artificial intelligence review (1999) 393–408.
[37] T. Tran, R. Cohen, Hybrid recommender systems for electronic commerce, in: Knowledge-Based Electronic Markets, Papers from the AAAI Workshop, Technical Report WS-00-04, AAAI Press, 2000.
[38] J. Kleinberg, M. Sandler, Using mixture models for collaborative filtering, in: Proceedings of the 36th Annual ACM Symposium on Theory of Computing, 2004, pp. 569–578.
[39] Chu-Ren Huang, P. Simon, Shu-Kai Hsieh, L. Prevot, Rethinking chinese word segmentation: tokenization, character classification, or wordbreak identification, in: Proceeding ACL '07 Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, 2007, pp. 69–72.
[40] J.L. Herlocker, J.A. Konstan, L.G. Terveen, J.T. Riedl, Evaluating collaborative filtering recommender systems, ACM Trans. Inf. Syst. 22 (1) (2004) 5–53.
[41] R. Burke, Hybrid web recommender systems, in: The Adaptive Web, Springer, Berlin, Heidelberg, 2007, pp. 377–408.
[42] G. Shani, M. Chickering, C. Meek, Mining recommendations from the web, in: Proceedings of the 2008 ACM Conference on Recommender Systems, 2008, pp. 35–42.
[43] M. Ge, C. Delgado-Battenfeld, D. Jannach, Beyond accuracy: evaluating recommender systems by coverage and serendipity, in: Proceedings of the Fourth ACM Conference on Recommender Systems, 2010, pp. 257–260.
[44] N. Lathia, S. Hailes, L. Capra, X. Amatriain, Temporal diversity in recommender systems. in: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2010, pp. 210–217.
[45] S. McNee, J. Riedl, J. Konstan, Being accurate is not enough: how accuracy metrics have hurt recommender systems, in: CHI '06 Extended Abstracts on Human Factors in Computing Systems, 2006, pp. 1097–1101.