Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 157 (2019) 345–352

Procedia Computer Science

www.elsevier.com/locate/procedia

4th International Conference on Computer Science and Computational Intelligence 2019 (ICCSCI), 12–13 September 2019

Enhancing the Performance of Library Book Recommendation System by Employing the Probabilistic-Keyword Model on a Collaborative Filtering Approach

Noor Ifada[a,*], Irvan Syachrudin[a], Mochammad Kautsar Sophan[a], Sri Wahyuni[b]

[a]*Informatics Department, University of Trunojoyo Madura, Bangkalan, Indonesia 69162*
[b]*Electrical Engineering Department, University of Trunojoyo Madura, Bangkalan, Indonesia 69162*

Abstract

This paper proposes the probabilistic-keyword CF method for a library book recommendation system. Our focus is to address the sparsity problem commonly occurs on the Collaborative Filtering (CF) approach. The framework of the method consists of four processes. First, building the circulation and keyword matrices respectively based on the book circulation records and the book keyword attribute data. Second, building the keyword model that takes into account both the book circulation records and the book keyword data. Third, building the probabilistic-keyword model that employs a probabilistic technique to calculate the probability of a user to borrow a book conditional to his/her keyword model. Fourth, generating the top-*N* book recommendations. Experiment results on a library dataset show that our probabilistic-keyword CF method outperforms the traditional user-based and item-based CF methods in terms of all evaluation metrics. This result conjectures that the probabilistic-keyword CF method that employs the probabilistic-keyword model can enhance the recommendation performance and is able to deal with the sparse dataset better than the traditional methods.

Keywords: Collaborative filtering; Library book recommendation; Probabilistic-keyword model;

* Corresponding author.
E-mail address: noor.ifada@trunojoyo.ac.id

1877-0509
10.1016/j.procs.2019.08.176

1. Introduction

The library plays an essential role in supporting the learning and research activities in a university. One of the key divisions of a library is the circulation division in which the borrowing book privilege service is accessible for its users, i.e., students, lecturers, and staffs. Given a vast number of book collections available in the library, the searching feature of a library catalog system is not sufficient to help users in finding books of his interest [1 2]. The implementation of the library book recommendation system could work as an efficient solution to this problem.

Recommendation system overcomes the excessive information offered to target users by recommending a personalized list of items of users' interests. In terms of the library book, the implementation of a system can enhance the user's satisfaction in searching books and reading interest [2], as well as the resource utilization and the user's experience [1].

The Collaborative Filtering (CF) approach is the most widely-used learning model implemented in the recommendation systems [3 4 5]. CF approach generates the list of recommendations by empowering the collaborative feedbacks, i.e., the known user-book relations, given by multiple users [6]. In this case, the collaborative feedbacks are modeled such that the similarities of users or items influence the predictions. Dealing with the sparse data is a common limitation in CF [6 7 8]. A CF-based recommendation system is prone to result in a poor recommendation performance due to the sparsity problem. The system is incapable of determining the users and items similarities given that most of the user-book relations are unknown [4 6 8 9]. One alternative solution to tackle the problem is by applying the attribute-based model, i.e., taking into account attribute data in the learning model [9 10 11 12]. However, existing methods did not consider the implementation of a probabilistic technique to control further the importance of item attribute for enhancing the recommendation performance [13 14 15].

The focus of our work is to deal with the sparsity problem on a library book recommendation system by employing a probabilistic item attribute model on a CF approach. In the proposed method, the probabilistic technique is implemented to predict the list of book recommendations for a target user conditional to the item attribute model, i.e., keyword model, built from the book circulation records and the book keywords data.

Experimental results on a library dataset show that our proposed method can outperform the traditional user-based CF (UBCF) and item-based CF (IBCF) methods. This finding confirms that the proposed method that employs probabilistic-keyword model can tackle the sparsity problem and therefore boost the performance of library book recommendations.

The rest of this paper is structured as follows. Section 2 describes the related work. Section 3 details our probabilistic-keyword CF method. Section 4 presents the experimental results based on a library dataset. Lastly, the conclusion is listed in Section 5.

2. Related Work

The attribute-based models have shown to able to tackle the sparsity problem and enhance the recommendation performance in CF. Attribute models can alternatively be built by using: the combination of the users and items similarities to their corresponding attributes [9], the combination of the items similarities to their attributes and the time-weight score [11], the combination of the rating data and the item-attribute similarities [10], and the folksonomy network [12]. However, existing methods basically implemented the models on the traditional CF approach, i.e., user-based or item-based, and did not consider the implementation of a probabilistic technique that has proven can enhance the recommendation performance in the Social Tagging systems [13 14 15]. Those probabilistic-attribute methods used the attributes, i.e., tags, that were linked to the users and items and formed the ternary relationships.

In this paper, we propose to combine the advantage of both the attribute and probabilistic models, then frame them as a probabilistic-attribute model. Unlike the existing probabilistic-attribute methods [13 14 15], our method use attributes, i.e., keywords, that are attached initially to the items/books only. The keyword model that depicts the relationship of users, books, and keywords are built through the utilization of user-keyword and keyword-book frequencies.

3. Probabilistic-keyword CF Method

Our proposed probabilistic-keyword CF method is focusing on a library book recommendation system. To solve the sparsity issue of CF approach, we build a keyword model that takes into account both the book circulation records and book keyword data. As depicted in Fig. 1, the framework of our method is decomposed into four processes: (1) building the circulation and keyword matrices, (2) building the keyword model, (3) building the probabilistic-keyword model, and (4) generating the top-*N* book recommendations.

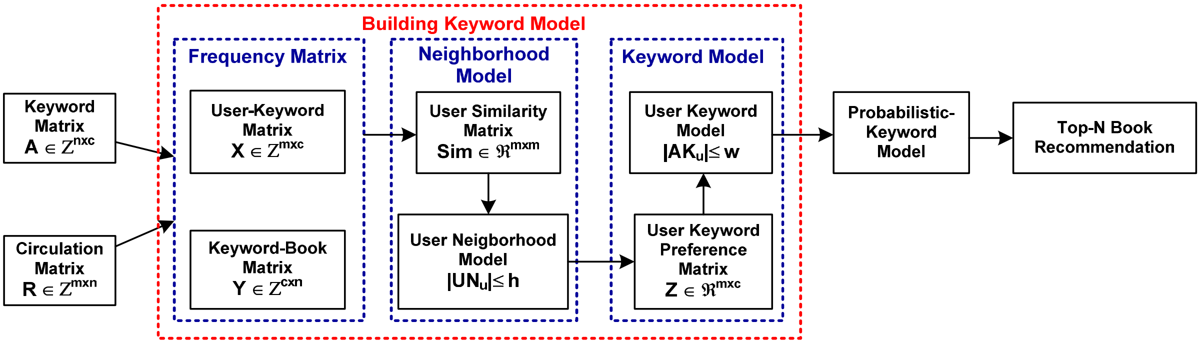


Fig. 1. Framework of our probabilistic-keyword CF method

3.1. Building the Circulation and Keyword Matrices

The circulation and keyword matrices are respectively constructed from the book circulation records and the book keyword attribute data. Assume that we have $U = \{u_1, u_2, u_3, \dots, u_m\}$, $I = \{i_1, i_2, i_3, \dots, i_n\}$, and $K = \{k_1, k_2, k_3, \dots, k_c\}$ as the set of m users, n books, and c keyword attributes, respectively. The book circulation records represent the binary correlation between user u and book i that can be modeled as a circulation matrix $R \in \mathbb{Z}^{m\times n}$ where $r_{u,i}$ is set to 1 if user u has borrowed book i or 0 otherwise. The book keyword data represents the list of keyword attributes of each book that can be modeled as a keyword matrix $A \in \mathbb{Z}^{n\times c}$ where $a_{i,k}$ is denoted as 1 if book i has keyword k or 0 otherwise. Fig. 2(a) and (b) respectively show the toy examples of a circulation $R \in \mathbb{Z}^{3\times 4}$ and a keyword $A \in \mathbb{Z}^{4\times 5}$ matrices where $U = \{u_1, u_2, u_3\}$, $I = \{i_1, i_2, i_3, i_4\}$, and $K = \{k_1, k_2, k_3, k_4, k_5\}$.

		Book			
		i_1	i_2	i_3	i_4
User	u_1	1	0	1	0
	u_2	1	1	1	0
	u_3	0	0	1	1

(a)

		Keyword				
		k_1	k_2	k_3	k_4	k_5
Book	i_1	1	0	0	0	0
	i_2	0	1	1	0	0
	i_3	0	0	1	1	1
	i_4	0	0	1	0	0

(b)

Fig. 2. Toy examples: (a) Circulation matrix $R \in \mathbb{Z}^{3\times 4}$ and (b) Keyword matrix $A \in \mathbb{Z}^{4\times 5}$

3.2. Building the Keyword Model

3.2.1. User-Keyword and Keyword-Book Frequency Matrices

The user-keyword and keyword-book frequency matrices are constructed out of the circulation R and keyword A matrices. The user-keyword frequency matrix $X \in \mathbb{Z}^{m\times c}$ lists the keyword usage on books of each user u, where $x_{u,k}$ denotes the number of books that a user u has borrowed with a keyword k. Meanwhile, the keyword-book frequency matrix $Y \in \mathbb{Z}^{c\times n}$ lists the popularity of each keyword k amongst users, where $y_{k,i}$ denotes the number of users who have borrowed a book i with a keyword k. Fig. 3(a) and (b) respectively show the toy examples of a user-keyword $X \in \mathbb{Z}^{3\times 5}$ and a keyword-book $Y \in \mathbb{Z}^{5\times 4}$ frequency matrices constructed based on the circulation $R \in \mathbb{Z}^{3\times 4}$ and keyword $A \in \mathbb{Z}^{4\times 5}$ matrices of Fig. 2(a) and (b).

(a)

		Keyword				
		k_1	k_2	k_3	k_4	k_5
User	u_1	1	0	1	1	1
	u_2	1	1	2	1	1
	u_3	0	0	2	1	1

(b)

		Book			
		i_1	i_2	i_3	i_4
Keyword	k_1	2	0	0	0
	k_2	0	1	0	0
	k_3	0	1	3	1
	k_4	0	0	3	0
	k_5	0	0	3	0

Fig. 3. Toy examples: (a) User-Keyword frequency matrix $X \in \mathbb{Z}^{3\times5}$ and (b) Keyword-Book frequency matrix $Y \in \mathbb{Z}^{5\times4}$

3.2.2. User Neighborhood Model

The concept of building the user neighborhood model is that users are considered to have a higher similarity with a target user u when they have used a significant number of same keywords. Meanwhile, keywords that are used by many users are considered to have less similarity contribution compared to those used by a small number of users. In this research, the user neighborhood model is generated based on the similarity of the keyword usage on books and the keyword popularity amongst users. The similarity between user u and v is computed by using the cosine similarity:

$$sim_{u,v} := \frac{\sum_{k=1}^{c}(x_{u,k}\cdot f_k)(x_{v,k}\cdot f_k)}{\sqrt{\sum_{k=1}^{c}(x_{u,k}\cdot f_k)^2}\sqrt{\sum_{k=1}^{c}(x_{v,k}\cdot f_k)^2}} \tag{1}$$

where $x_{u,k}$ and $x_{v,k}$ denote the keyword usage similarity of user u and v in the user-keyword frequency matrix X. While f_k refers to the dissimilarity of keyword popularity amongst users in the keyword-book frequency matrix Y:

$$f_k := \log\left(m/\sum_{i=1}^{n} y_{k,i}\right) \tag{2}$$

The similarity score between user u and v is in the range of $[0,1]$. The higher the score, the more similar user u to user v. For m users, the users' similarities form a similarity matrix $Sim \in \mathbb{R}^{m\times m}$. The user neighborhood UN_u is then built as a set of top-h nearest neighbors of user u generated based on the similarity values of $Sim_{u,*}$ sorted in descending order such that $UN_u \subseteq U$ and $|UN_u| \le h$.

3.2.3. Keyword Model

For generating the keyword model, first, the preference score of each target user u for a keyword k must be computed. The score is achieved by summing the keyword k usage of the neighbor users UN_u weighted by the conforming similarity score between the target user u and his/her each neighbor $v \in UN_u$. The formulation of how much the target user u prefers a keyword k is given by:

$$z_{u,k} := \sum_{v\in UN_u} x_{v,k}.\, sim_{u,v} \tag{3}$$

For m users and c keywords, the users keyword preference form a preference matrix $Z \in \mathbb{R}^{m\times c}$. The keyword model AK_u is then built as a set of top-w preferred keywords of user u generated based on the preference scores of $Z_{u,*}$ sorted in descending order such that $AK_u \subseteq K$ and $|AK_u| \le w$.

3.3. Building the Probabilistic-Keyword Model

The probabilistic-keyword model is created by considering the CF approach as a classification problem. Hence, the classifier predicts, for each target user u, the probability of a target book i_y belongs to class book given the user u keyword model AK_u used as the feature variables. By using the multinomial event model distribution assumption for the Naïve Bayes [16], the posterior probability of user u for a book i_y given keyword model AK_u is formulated as:

$$p_{u,i_y} := p(i_y|AK_u) := p(I = i_y)\prod_{j=1}^{w} p(k_j|I = i_y)^{(X_{u,j}+1)} \tag{4}$$

where $p(I = i_y)$ dan $p(t_j|I = i_y)$ are determined as :

$$p(I = i_y) := \frac{\sum_{u=1}^{m} R_{u,y}}{\sum_{y=1}^{n}\sum_{u=1}^{m} R_{u,y}} \tag{5}$$

$$p(k_j|I = i_y) := \frac{1 + Y_{j,y}}{c + \sum_{k=1}^{c} Y_{k,y}} \tag{6}$$

To avoid zero values on Equation (4) and (6), we apply the Laplacian estimate [17] as a smoothing technique by adding "1" to those equations.

3.4. Generating the Top-N Book Recommendations

For the target user u, the list of top-N book recommendation is an ordered set of books, $TopN_u$, obtained by descending order the posterior probability scores p_{u,i_y}. The complete algorithm of the proposed probabilistic-keyword CF method is shown in Fig. 4.

Algorithm : Probabilistic-keyword CF Method

Input : Book circulation records and book keyword data, size of user neighborhood UN h, size of keyword model AK w

Output : top-N list of book recommendation for target user u $TopN_u$

1. Construct the circulation $R \in \mathbb{Z}^{m\times n}$ and keyword $A \in \mathbb{Z}^{m\times c}$ matrices from the book circulation records and keyword attribute data
2. Build the keyword model:
   a. Construct the user-keyword $X \in \mathbb{Z}^{m\times c}$ and keyword-book $Y \in \mathbb{Z}^{c\times n}$ frequency matrices
   b. Based on the user similarity matrix $Sim \in \mathbb{R}^{m\times m}$ constructed using Equation (1), generate the top-h nearest neighbors of user u UN_u for target user u where $UN_u \subseteq U$ and $|UN_u| \le h$
   c. Based on the user keyword preference matrix $Z \in \mathbb{R}^{m\times c}$ constructed using Equation (3), generate the top-w keyword model AK_u for target user u where $AK_u \subseteq K$ and $|AK_u| \le w$
   d. Build the probabilistic-keyword model by calculating the posterior probability using Equation (4)
3. Generate the top-N list of book recommendation for target user u $TopN_u$ based on the posterior probability values

Fig. 4. Algorithm of probabilistic-keyword CF method

4. Empirical Analysis

4.1. Dataset and Evaluation Method

For the experiments, we use the dataset of UTM Library (https://library.trunojoyo.ac.id/) collected in 2016-2017. Table 1 lists the description of the dataset. Note that the $Circulation$ and $BookCollection$ data are respectively used to construct the circulation matrix R and the keyword matrix A. As shown in Table 2, the dataset is very sparse since 99.7383% of $r_{u,i}$ of the circulation matrix R are unknown, i.e., labeled as "0".

Table 1. Description of UTM library dataset

Data	Description
$Circulation$	The book borrowing records: ID_Member, ID_Col, Date_Borrow, Date Should Return, Date Returned
$BookCollection$	Information about the book collections in the library: ID_Col, Classification, Author, Title, Year Published, Publisher, City Published, Keyword
$Member$	Information about the users of the library: ID, Department, Name, Sex, Category

Table 2. Statistic of UTM library dataset

Total number of users (m)	Total number of books (n)	Total number of keywords (c)	Total number of book circulation records $(\lvert r_{u,i} == 1\rvert)$	Density $\left(\frac{\lvert r_{u,i} == 1\rvert}{m \cdot n} \cdot 100\%\right)$	Sparsity $(100\% - Density)$
1007	3585	1854	9448	0.2617%	99.7383%

As the evaluation method, we apply the 5-fold cross-validation approach such that we arbitrarily split every fold as two sets: 80% training data D_{train} and 20% test data D_{test}. Thus, the performance results are presented as the average of five folds results.

4.2. Evaluation Metric

The task of book recommendation is to generate the top-N books prediction for the target users in D_{test}. For the evaluation, the top-N list of book recommendations for a user u, $Top_u(N)$, are then matched to his/her ground-truth books, GT_u, recorded in D_{test}. We use various evaluation metrics commonly used for assessing the recommendation performance of a target user u, i.e., Average Precision (AP), F1-Score, and Normalized Discounted Cumulative Gain (NDCG):

$$AP_u(N) := \sum_{n=1}^{N} Precision_u(N) \cdot \mathbb{I}(Top_u(n) \in GT_u) \tag{7}$$

$$F1\text{-}Score_u(N) := \frac{2 \cdot Precision_u(N) \cdot Recall_u(N)}{Precision_u(N) + Recall_u(N)} \tag{8}$$

$$NDCG_u(N) := \frac{DCG_u(N)}{IDCG(N)} \tag{9}$$

where $\mathbb{I}(\cdot)$ results 1 when the condition within the bracket is satisfied, or else 0. The Precision, Recall, DCG, and IDCG of each target user u are calculated as:

$$Precision_u(N) := 100 \cdot \frac{|Top_u(N) \cap GT_u|}{N} \tag{10}$$

$$Recall_u(N) := 100 \cdot \frac{|Top_u(N) \cap GT_u|}{|GT_u|} \tag{11}$$

$$DCG_u(N) := \sum_{n=1}^{N} \frac{1}{\log_2(1+n)} \cdot \mathbb{I}(Top_u(n) \in GT_u) \tag{12}$$

$$IDCG(N) := \sum_{n=1}^{N} \frac{1}{\log_2(1+n)} \tag{13}$$

The reported performance scores are the average results of all target users in D_{test}.

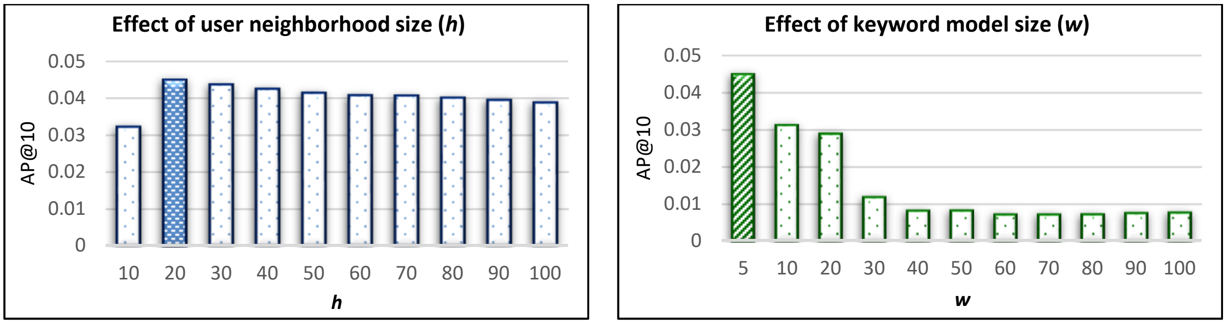
4.3. Experiment Result

4.3.1. Sensitivity Analysis

We conduct the sensitivity analysis of our proposed method to the size of the user neighborhood and keyword models:

- Effect of the user neighborhood UN size (h). The h is the size of the user neighborhood to generate UN_u for each target user u that is used in Equation (3). We conduct the experiments on a variety of user neighborhood size $h = \{10,20,30,40,50,60,70,80,90,100\}$. Fig. 5(a) shows that the performance of recommendation is in lines to h until $h = 20$ and then deteriorating when $h > 20$. This finding concludes that the number of target user's best neighbor is 20 users.

- Effect of the keyword model AK size (w). The w is the size of the keyword model to generate AK_u for each target user u that is used in Equation (4). We conduct the experiments on a variety of user neighborhood size $w = \{5,10,20,30,40,50,60,70,80,90,100\}$. Results in Fig. 5(b) show that our method performs the best at $w = 5$ and is declining on larger w. In other words, a target user's interest is best represented with five keywords.



(a) (b)

Fig. 5. Sensitivity analysis: (a) Effect of user neighborhood size h and (b) Effect of keyword model size w

4.3.2. Comparison of Recommendation Performance

The performance of our method (probabilistic-keyword CF method) is benchmarked with the following:

- UBCF [18]: The traditional user-based CF method which conjectures that the users that have similar rating pattern with a target user are influencing his/her preference and, therefore, the list of recommendations are generated based on the users' similarities. To obtain its best performance, the tuned parameter for UBCF is $user\_neighborhood\_size = 80$.
- IBCF [4]: The traditional item-based CF method which conjectures that a user inclines to pick similar items that he/she liked in the past and, therefore, the list of recommendations are generated based on the similarities of the items. To obtain its best performance, the tuned parameter for IBCF is $item\_neighborhood\_size = 100$.

Table 3 lists the comparisons of recommendation performance of our proposed probabilistic-keyword CF method and the benchmarking methods in terms of AP, F1-Score, and NDCG at various top-N, where $N = \{10,20,30\}$. We obtain two observation based on those results.

First, our method outperforms all of the benchmarking methods in terms of all evaluation metrics. Note that the higher the N, the higher the AP score is. On the contrary, the higher the N, the lower F1-Score and NDCG scores are. All of these results confirm that our probabilistic-keyword CF method that employs the probabilistic-keyword model can enhance the performance of recommendations.

Two, given the 99.7383% sparsity of the dataset (see Table 2), we can establish that our proposed method can deal with the sparse dataset and therefore it is superior in handling such problem commonly occurred in CF-based methods, in compared to all benchmarking methods.

Table 3. Comparison of recommendation performance

Method	AP			F1-Score			NDCG		
	@10	@20	@30	@10	@20	@30	@10	@20	@30
Probabilistic-keyword CF	**0.044893**	**0.048274**	**0.049809**	**0.031924**	**0.026006**	**0.022246**	**0.026632**	**0.020544**	**0.01739**
UBCF	0.038961	0.042022	0.043249	0.029564	0.02433	0.020299	0.023672	0.018481	0.015493
IBCF	0.009884	0.011858	0.012935	0.010671	0.011138	0.010833	0.006991	0.006598	0.006249

5. Conclusion and Future Work

Our proposed probabilistic-keyword CF method is developed to solve the sparsity problem of a library book recommendation system. That is by building the keyword model that takes into account both the book circulation records and book keyword data, and the probabilistic-keyword model that employs a probabilistic technique to predict the list of book recommendations for a target user that is conditional to the keyword model.

The experiment results show that our proposed method outperforms its benchmarks, i.e., the traditional user-based and item-based CF methods. This result conjectures that the probabilistic-keyword CF method that employs the probabilistic-keyword model can enhance the recommendation performance and is able to deal with the sparse dataset better than the benchmarking methods. In the near future, we are planning to study further the potential of building the attribute model by combining the attributes data of both the users and items to solve the sparsity problem.

Acknowledgments

This work is supported by the Ministry of Research, Technology and Higher Education (Indonesia) under the "Penelitian Dasar" grant scheme, for the financial year 2019, contract number: 221/SP2H/LT/DRPM/2019.

References

1. Chen C, Zhang L, Qiao H, Wang S, Liu Y, Qiu X. Book recommendation based on book-loan logs. In The 2012 International Conference on Asian Digital Libraries; 2012; Taiwan, China. p. 269-78.
2. Jomsri P. Book recommendation system for digital library based on user profiles by using association rule. In The 4th edition of the International Conference on the Innovative Computing Technology (INTECH 2014); 2014; Luton, UK. p. 130-134.
3. Konstan JA, Riedl J. Recommender systems: From algorithms to user experience. User Modeling and User-Adapted Interaction. 2012; 22(1-2): p. 101-123.
4. Sarwar B, Karypis G, Konstan J, Riedl J. Item-based collaborative filtering recommendation algorithms. In The 10th International Conference on World Wide Web; 2001; Hongkong. p. 285-295.
5. Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering. 2005; 17(6): p. 734-749.
6. Aggarwal CC. Recommender systems: The textbook: Springer; 2016.
7. Ifada N, Nayak R. How relevant is the irrelevant data: Leveraging the tagging data for a learning-to-rank model. In The 19th ACM International Conference on Web Search and Data Mining; 2016; San Francisco, California, US. p. 23-32.
8. Guo G, Zhang J, Thalmann D. Merging trust in collaborative filtering to alleviate data sparsity and cold start. Knowledge-Based Systems. 2014; 57: p. 57-68.
9. Gong SJ. Employing user attribute and item attribute to enhance the collaborative filtering recommendation. Journal of Software. 2009; 4(8): p. 883-890.
10. Ifada N, Prasetyo EH, Mula'ab. Employing sparsity removal approach and fuzzy c-means clustering technique on a movie recommendation system. In The 2010 International Conference on Computer Engineering, Network and Intelligent Multimedia (CENIM); 2018; Surabaya, Indonesia. p. 157-162.
11. Chen Q, Li W, Liu J. Collaborative filtering algorithm based on item attribute and time weight. In The 2016 International Conference on Automatic Control and Information Engineering; 2016; Hongkong. p. 12-15.
12. Luo X, Ouyang Y, Xiong Z. Improving neighborhood based collaborative filtering via integrated folksonomy information. Pattern Recognition Letters. 2012; 33(3): p. 263-270.
13. Kim HN, Ji AT, Ha I, Jo GS. Collaborative filtering based on collaborative tagging for enhancing the quality of recommendation. Electronic Commerce Research and Applications. 2010; 9(1): p. 73-83.
14. Ifada N, Nayak R. A two-stage item recommendation method using probabilistic ranking with reconstructed tensor model. In User modeling, adaptation, and personalization.: Springer; 2014. p. 98-110.
15. Ifada N, Nayak R. Tensor-based item recommendation using probabilistic ranking in social tagging systems. In The 23rd International Conference on World Wide Web Companion; 2014; Seoul, Korea. p. 805-810.
16. Baker LD, McCallum AK. Distributional clustering of words for text classification. In The 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval; 1998; Melbourne, Australia. p. 96-103.
17. Lops P, Gemmis M, Semeraro G. Content-based recommender systems: State of the art and trends. In Recommender systems handbook.: Springer; 2011. p. 73-105.
18. Ekstrand MD, Riedl JT, Konstan JA. Collaborative filtering recommender systems. Human–Computer Interaction. 2010; 4(2): p. 81-173.