PROCEEDINGS OF SCIENCE

A New-come Book Recommendation Algorithm Based on Features in University's Library

Yue Qi[1]
Library of Southwest University,Chongqing, 400715,China
Email: 19557834@qq.com

Wenwen Chen[2]
Library of Southwest University,Chongqing, 400715,China
E-mail: qqt.123@163.com

Huan Zhou[3]
Library of Southwest University,Chongqing, 400715,China
E-mail: tannbitous@sina.cn

The university's libraries have always had too many resources to be found by readers so the value of many books has been reduced for missing prescription. Although there have already been a lot of book recommendation researches in university library, few studies have been carried out in respect of the fact how to recommend the new-come books. New books should be more valuable to be recommended, but often overlooked by readers in the recommendation systems because of low weight. To solve this problem, a new-come book recommendation algorithm has been presented to help readers find their interested new books in time and improve all the new books' value in use. Through extracting the feature of new books and analyzing readers' borrowing behavior, effort have been devoted to get the readers' borrowing interest and the popular books recently, and calculate the similarity between books and readers through feature extraction so as to find out the new books which maybe popular and recommend them to all readers timely, and find right readers for other new books. With experiments carried out by using historical data of Southwest University Library, to some extent, it shows that the algorithm can find the new-come books which comply with the popular feature or may be interested by some readers, then get the personalized recommendation result effectively.

CENet2015
12-13 September 2015
Shanghai, China

[1] Speaker
[2] This study is supported by "Fundamental Research Funds for the Central Universities (XDJK2014C165)" and "Southwest University Research Project (SWULIB2015C02)".
[3] Correspongding Author

1.Introduction

The university libraries are always rich in resources. Taking the library of Southwest University as example, by 2014, there are about 3 million books with the number keeping growing. While satisfying the growing demons of information services, the "information overload" has become a deterrent to readers to find useful knowledge[1].

At present, many universities and research institutions are working on the library personalized recommendation research and practice [2], but few studies have been carried out in respect of the fact how to recommend new-books. Books have strong timeliness. The newly-existing books are more attractive for readers either by appearances or by contents, but as to the cold start problems of the recommendation systems[3], they often can't be recommended to readers by the low weight.

There have been many sophisticated recommended methods applied to a wide variety of systems. Personalized recommendation algorithm has extracted the users' feature according to their basic information and historical behavior, acquiring users' requirements, thereby helping users to get useful information [4], such as "content-based book recommendation algorithm" and "CF-based book recommendation algorithm" [5], but none of them worked on handling the new-come books. To solve this problem, this paper proposes a new-come book recommendation algorithm (hereinafter referred to as the algorithm), which extracts the features of the books to construct book feature vectors and readers' interest vector, and calculates the similarity between books and readers in the same vector space so as to find suitable readers for new books.

This paper, on the basis of a recommended algorithm for new book in the university library, tends to help readers find new books. The algorithm can find out the potential readers for the sake of the latest income books to the library, and recommend books to them at meanwhile.

2. New Book Recommendation Algorithm

The new book recommendation algorithm (hereinafter referred to as the algorithm) is hereby proposed to solve the problem of finding the recommended readers for the new books. It includes feature abstraction modeling, hot words induction and recommendation operation, as described in detail below.

2.1 Define theBbook Feature Vector

We express books in terms of vectors. Each item of the vector represents a value of book features. These values come from the digital library system directly. We believe the most important features of book are the subject thesaurus rather than classification. As the classification of books by the Chinese Library Classification is inflexible [6] to some extent, the clustering books only by classification code will make the recommendation result to be simplex; besides, the book has many other elements, such as author and press, etc.. In order to express the book features completely, the book feature vector should include subject thesaurus, classification code, author and press; thus the vector is defined as follows:

$$V_b=\{subject_1, subject_2, subject_3, subject_4, author, classification, press\} \quad (2.1)$$

To simplify the computation, we use the first four subject thesaurus of a book as four items of the vector. If a book has more than four subjects, we use the first four as features of the book; on the other hand, if there are less than four subjects, set the rest values of the items as void. To treat these features as the same, the book feature vector is defined as:

$$V_b=\{k_1, k_2, k_3, k_4, k_5, k_6, k_7\} \quad (2.2)$$

2.2 Define the Rreader Interest Vector

The features of reader interest are extracted from the features of books that the reader borrowed before. The follow discussion is made how to define the reader interest vector through the library records from the digital library system.

2.2.1The Borrowing Feature Vector

Firstly, define a period of time as $T = Te - Ts$, and retrieve a reader's library records from Ts to Te. The core content of each record is a book and each book can be seen as V_b as defined in Equation 1. These vectors can form a matrix, which includes all the features of books that the reader has borrowed; thus the matrix is called the borrowing feature matrix, defined as follow:

$$M_r = \{V_{b1}, V_{b2}, \ldots, V_{bi}, \ldots, V_{bn},\} \tag{2.3}$$

Where Vb_i means the feature vector of a book and the reader borrowed in time period T.

Each element in the matrix is one item of a book feature vector. As not all books have no less than four subject thesaurus, some elements of M_r will be null. Traverse M_r, save all the non-empty elements in a linear array, then all elements of the array express the features of books the reader borrowed. The array is defined as a vector called borrowing feature vector, and the expression is shown as follows:

$$V_r = \{r | M_r \wedge (r \neq \phi)\} \tag{2.4}$$

2.2.2The Reader's Interest Vector

Sometimes there is blindness and randomness in readers' lending behavior, so V_r cannot be directly used as reader interest vector; but it is considered that words appearing more frequently in the borrowing feature vector can describe features of reader's interests. According to the idea of Content-based recommendation algorithm, a text can be created with all the items in V_r. As to the term frequency (TF), the number of times that each feature value appears in the text of a feature that values k_i can be calculated as:

$$TF_i = \frac{n_i}{|V_r|} \tag{2.5}$$

Where n_i is the number of items in the value of k_i. Thus, the lager TF_i values are, the more feature k_i can express reader's interests.

To choose the effective words, we set a threshold θ. When $TF_i > \theta$, it indicates that feature k_i appears frequently enough, and k_i can be treated as the reader's interest feature. Gathering all qualified feature to form a vector, we get the reader's interest vector V_i.

The exact value of thresh old θ should be determined by the actual situation. As to a new digital library system, θ should be set to a smaller value in case there are too few items in V_i; on the other hand, if the system has a large number of library records data, the value of θ should be appropriately increased.

2.3 Define the Hot Words

Many recommendation researches always focus on the personality of recommended contents [7], but ignore the commonality. In fact, readers' interests have a lot in common. If a feature value appears in many reader's interest vectors, then another reader, who doesn't have the value in his interest vector, is also likely to have interest in this feature, which is called the hot key word. The recommended algorithm should help readers explore their potential interests. One way is to recommend new books that contain the hot key word in the book feature vectors, which is an important way.

With statistics of all the items gathered from all reader's interest vectors in digital library system, we can get the features that have lager value of TF. These values of the features are the so called hot words. Put each item of all the reader's interest vectors as an element into a linear array, calculate TF of all different element values and sort them in descending order; then the top n elements of the array are the hot words. We record it as the vector below:

$$P = \{k_1, k_2, k_3, \ldots, k_n\} \tag{2.6}$$

2.4 The Recommendation Process

The basic idea of the algorithm is: when a new book data has been input in the digital library system, we acquire its book feature vector to find out whether there are hot key words in the items. If so, the book is considered as a popular book, we can recommend to all readers; otherwise, we compare the book feature vector to reader interest vectors. A higher similarity indicates that the reader is more likely to have interest in the book and the book can be one of the recommendation results to the reader. The similarity formula [8] is shown as follows:

$$\operatorname{sim}\left(V_b(i), V_i(r)\right)=\frac{\left|V_b(i)\cap V_i(r)\right|}{\left|V_b(i)\cup V_i(r)\right|} \tag{2.7}$$

Where $V_b(i)$ is the new book i's feature vector and $V_i(r)$ is the reader r's interest vector. The numerator of the formula is the number of items exist in the both vectors, and the denominator is the number of different items in the two vectors. The algorithm process is shown as follows:

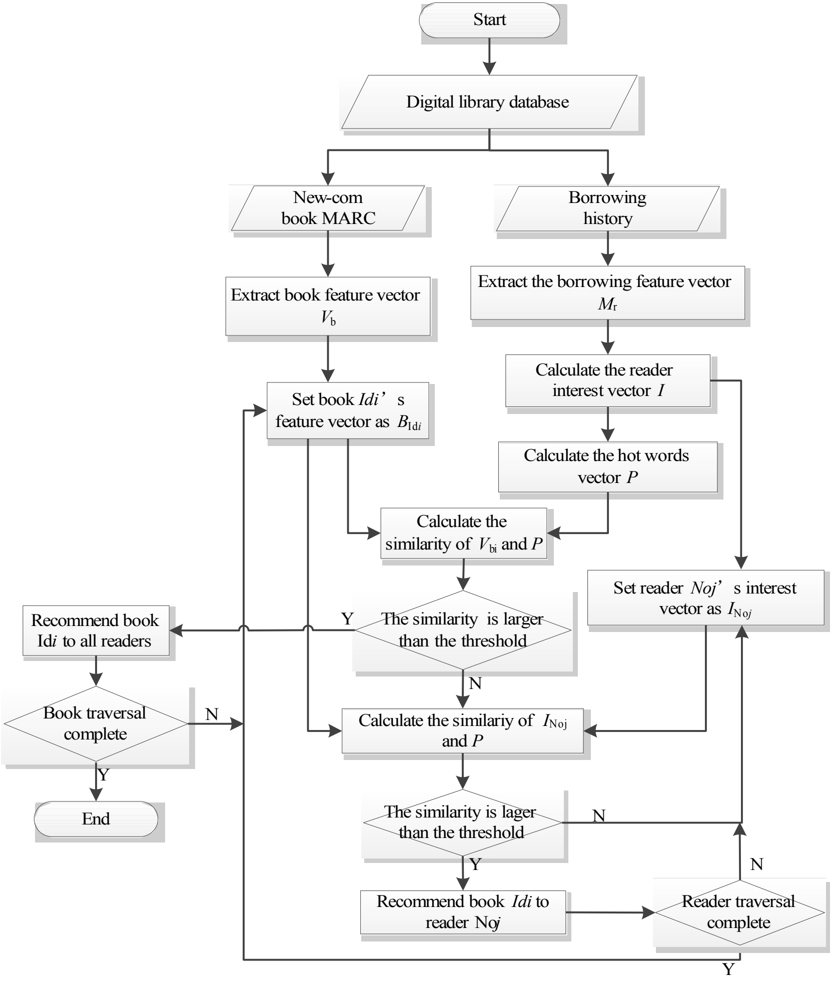

Start
Digital library database
New-com book MARC
Borrowing history
Extract book feature vector V_b
Extract the borrowing feature vector M_r
Calculate the reader interest vector I
Set book Idi' s feature vector as B_{Idi}
Calculate the hot words vector P
Calculate the similarity of V_{bi} and P
Set reader Noj' s interest vector as I_{Noj}
Recommend book Idi to all readers
Y
The similarity is larger than the threshold
N
Book traversal complete
N
Y
End
Calculate the similariy of I_{Noj} and P
The similarity is lager than the threshold
N
Y
N
Recommend book Idi to reader Noj
Reader traversal complete
Y

Figure 1: The Flowchart of the Algorithm

Set a time interval as statistics period, in which, the system collects data from the borrowing history record, analyzes the readers' interests, calculate hot words and keeps these results in data tables. When a number of new-come books change their status to "circulation" in the database, the system starts the algorithm to find out the popular ones and recommend them to all readers and match others with the reader's interests by means of the similarity formula, and then recommends them to those who may be interested in it.

3. Experiment And Analysis

3.1 Dataset

The quality of the algorithm is evaluated by an offline experiment and the raw data comes from Southwest University digital library system. In order to simplify the calculation, we ignore the canceled readers in the latest two years to make the number of readers unchanged. Multiple identical copies are considered as one book. In this context, the system owns 61604 registered readers, 61316 new books and 781,894 library records from January 2013 to December 2014.

The library records are divided into 8 groups quarterly, marked as M_1, M_2, ……, M_8. Firstly, set a variable i = 1, take M_1 to M_i as a training set and M_{i+1} as test set; secondly, take the newest n = 100 books and establish the book feature vectors and reader interest vectors from the training set; thirdly, get the recommendation results with the algorithm. And then track the library record of these n books on the test set, to calculate the evaluation index of the algorithm. Then, set i=2, repeat the steps above, calculate another evaluation index. Until i=7, we have seven evaluation indexes to get a comprehensive evaluation result.

3.2 The Evaluation Index

The Precision/Recall algorithm [9] is used here to evaluate the quality of the algorithm. The value of Precision can show how many recommended books are borrowed by readers in all recommendation results, and Recall shows the proportion of recommendation to readers' interests. The computation equations are as follow:

$$Precision = \frac{1}{n}\sum_{b=1}^{n}\frac{|R(b)\cap R_b(u)|}{|R(b)|}\times 100\% \tag{3.1}$$

$$Recall = \frac{1}{n}\sum_{b=1}^{n}\frac{|R(b)\cap R_b(u)|}{|R_b(u)|}\times 100\% \tag{3.2}$$

Where *R(b)* is the data set of recommended readers of book b on the training set, and the $R_b(u)$ is the data set of readers who have borrowed book b on the test set.

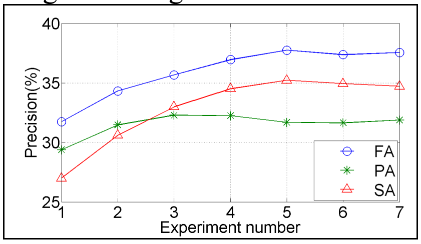
3.3 Results of the Experiment and Analysis

We use the algorithm here (hereinafter called FA) to compare with readers' borrowing preference based algorithm [10] (hereinafter called PA) and the CF recommending model based on borrowing-time scores [11] (hereinafter called SA). Because of the data sparsity, we set the recommendation threshold λ = 0.4. The results of the experiments sorted by numbers are shown as below in Table 1.

No.	FA(%)		PA(%)		SA(%)	
	Precision	Recall	Precision	Recall	Precision	Recall
1	31.74	20.50	29.37	19.36	26.99	18.21
2	34.32	21.75	31.47	20.37	30.59	19.95
3	35.66	22.39	32.28	20.76	32.99	21.11
4	36.94	23.01	32.24	20.74	34.50	21.83
5	37.74	23.40	31.68	20.57	35.22	22.17
6	37.37	23.26	31.64	20.36	34.93	22.03
7	37.55	23.31	31.88	20.51	34.71	22.11

Table 1: The Precision/Recall results

Upon drawing the data above in to a chart, the evaluation results can be compared more intuitively. See Fig. 2 and Fig. 3 below:



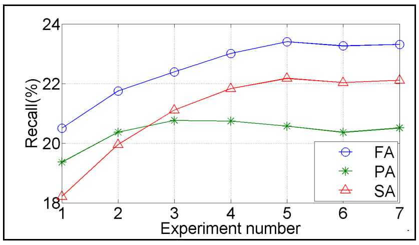


Figure 2: The Precision comparison **Figure 3:** The *Recall* comparison

With the number of the experiments increases, the value of evaluation index is higher when training set become bigger, which indicates that a richer set of data can get more effective results; besides, the experiment results show that the new book recommendation algorithm has a higher value of both Precision and Recall than the other two algorithms. It indicates the former has better performance on new book recommendation, but the scientifically offline experiment cannot completely reflect the effect of recommendation results to readers. the Precision value is no more than 40%, and the Recall value is about 23%.

4. Conclusion

In order to improve the utilization of new books in university library, a feature based recommendation algorithm has been studied in this paper. During every designated period, the system analyzes the readers' borrowing interests, calculates the hot word through the data of digital library database and keeps them in the data tables. The recommendation behavior is triggered by completion of a number of new-come books' data entry. With the algorithm, the system firstly extracts features of all these new books, determines the hot books to be recommend to all readers, and then recommends the others which are similar to reader interests through similarity algorithm so as to find suitable readers for them.

At last, using the dataset of Southwest University, the comparion of the algorithm with other two recommendation methods shows that the algorithm dose get personalized recommendation effectively.

References

[1] R. Zhou. *Internet age, information overload solution preliminary*[J]. Union BBS: journal of Shandong union management cadre institute. 17(5): 76-77(2011) (In Chinese)

[2] G. Adomavicius, A. Tuzhilin. *Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions*[J]. IEEE Tram on Knowledge and Data Engineering. 17(6): 734-749(2005)

[3] Y. G. Guo, G. S. Deng. *Collaborative filtering system project cold start hybrid recommendation algorithm*[J]. Computer engineering. 34(23): 11-13(2008) (In Chinese)

[4] G. X. Wang, H. P. Liu. *Personalized recommendation system review*[J]. Computer engineering and application. 13(7), 66-76 (2012) (In Chinese)

[5] P. Kantor, F. Ricci, L. Rokach, B. Shapira. *Recommender Systems Handbook*[M]. Springer, London. 74-78(2011)

[6] Y. P. Duan, *The Development History of Books' Classification*[J]. Sci-Tech Information Development & Economy. 21(8): 186-187(2011) (In Chinese)

[7] Z. Nie, H. Q. Wang, J. Zhou. *The personalized recommendation technology application in library service*[J]. modern intelligence. 33 (9): 95-102(2003) (In Chinese)

[8] J. S. Zhang, Y. C. Sun, H. L. Wang Huilin, Y. Q. He. *Calculating statistical similarity between sentences* [J]. Journal of Convergence Information Technology, 6(2): 22-34(2011)

[9] L. Xiang. *Recommendation system practice*[M]. Posts and Telecom Press, Beijing. 23-26(2011) (in Chinese)

[10] K. C. Li, D. M. Lan, X. E. Ling. *A personalized books recommend university readers borrowing preference*[J]. Modern intelligence. 33 (8): 68-72(2013) (In Chinese)

[11] M. C. Jing, Y. H. Yu. *CF Recommending Model Based on Borrowing-time Scores and Its Application* [J]. Library and Information Service. 56 (3): 117-120(2012) (In Chinese)