Research Paper Recommender Systems on Big Scholarly Data

Tsung Teng Chen[1] and Maria Lee[2](✉)

[1] National Taipei University, New Taipei City, Taiwan
timchen.ntpu@msa.hinet.net
[2] Shih Chien University, Taipei, Taiwan
Maria.lee@g2.usc.edu.tw

Abstract. Rapidly growing scholarly data has been coined Big Scholarly Data (BSD), which includes hundreds of millions of authors, papers, citations, and other scholarly information. The effective utilization of BSD may expedite various research-related activities, which include research management, collaborator discovery, expert finding and recommender systems. Research paper recommender systems using smaller datasets have been studied with inconclusive results in the past. To facilitate research to tackle the BSD challenge, we built an analytic platform and developed a research paper recommender system. The recommender system may help researchers find research papers closely matching their interests. The system is not only capable of recommending proper papers to individuals based on his/her profile, but also able to recommend papers for a research field using the aggregated profiles of researchers in the research field.

The BSD analytic platform is hosted on a computer cluster running data center operating system and initiated its data using Microsoft Academic Graph (MAG) dataset, which includes citation information from more than 126 million academic articles and over 528 million citation relationships between these articles. The research paper recommender system was implemented using Scala programming language and algorithms supplemented by Spark MLib. The performance of the recommender system is evaluated by the recall rate of the Top-N recommendations. The recall rates fall in the range of 0.3 to 0.6. Our recommender system currently bears the same limitation as other systems that are based on user-based collaborative filtering mechanisms. The cold-start problem can be mitigated by supplementing it with the item-based collaborative filtering mechanism.

Keywords: Big Scholarly Data · Recommender systems
Research paper recommender systems · Collaborative filtering

1 Introduction

Recommender Systems are software systems and techniques that suggest items to a user based on predicted user preference rating. As a subclass of information filtering systems, it tries to predict the “preference” or “rating” a user would give to an item. To tame the information explosion, recommender systems have become increasingly popular in

K. Yoshida and M. Lee (Eds.): PKAW 2018, LNAI 11016, pp. 251–260, 2018.
https://doi.org/10.1007/978-3-319-97289-3_20

recent years, and are applied in a variety of areas including entertainment content such as movies and music, knowledge and information acquirement such as news and research articles, and purchase suggestion for products in general. As a branch of recommender systems study, research paper recommender systems are more in the spotlight partially due to the already enormous and still fast growing research-related information available online. A research paper recommender system aims to mitigate the information overload and helps scholars to find relevant research papers suited to their interests. A research scholar may need to locate relevant papers to keep track in his or her field of study or to cite articles pertinent to an article s/he is working on.

Based on a literature survey study, at least 217 articles relevant to research paper recommendations had been published by 2013. About 120 different recommendation approaches were discussed in these articles. The recommendation approaches were categorized into seven main classes – Stereotyping, Content-based Filtering (CBF), Collaborative Filtering (CF), Co-Occurrence, Graph-based, Global Relevance, and Hybrid [1]. The Stereotyping approach was inspired by subject stereotyping found in the field of psychology that provides a mechanism of quickly judging people based on a few personal characteristics [2]. For example, a typical stereotype would be "woman is more interested in romance than man". Stereotypes could be constructed through a collection of personal traits and then applied in a recommendation setting. Content-based Filtering and Collaborative filtering are widely utilized recommendation mechanisms in various applications. CBF infers users' interests profile from the items the users interacted with, whereas an item is modeled and represented by its features. In the context of research paper recommender systems, word-based features are commonly used. Features of a paper are extracted from its textual content. The similarity of papers is calculated by comparing their features and used subsequently by the recommendation mechanism to recommend a paper that is similar to what the users like. CF tries to find like-minded users by preference ratings given by them. Two users are considered like-minded if they rate items alike. With the pool of identified like-minded users, items that rated positively by a user become recommending candidates for other users in the pool. The co-occurrence recommendation approach refers to the practice of recommending related items to a user. The relatedness between items may be established by items' co-occurrence, such as two papers are both cited by another paper, which creates a co-citation relationship between the two cited papers. The graph-based approach abstracts various relationships between entities into a graph and applies graph metrics, such as distance and centrality, to find recommendation candidates. The relationships used in a graph-based approach may include a citation or co-citation relationship between papers, co-authorship between authors, or venues of papers etc. The global relevance approach decides recommendation by utilizing some global metrics, such as the citation counts or the h-index of a publication or an author. The h-index is a metric that attempts to measure the impact of an author or a scholarly journal. The hybrid recommendation approach refers to combining two or more aforementioned methods into one. Despite various approaches that have been proposed, it remains unclear which one is more promising for many reasons [1]. One of the main reasons is dataset discrepancy, which refers to different datasets or different versions of a dataset that are used in the studies. Direct comparison between performance metrics calculated from different approaches are problematic since datasets may critically influence the performance of a recommender system. The scholarly datasets of research paper recommender systems have grown in size recently and are referred to as big scholarly data, which have been discussed in several articles [3–5]. The research paper recommender system is among one of the main applications in the analytics of big scholarly data.

We tried to achieve several objectives in this study. One was to build a research paper recommender system utilizing recommendation mechanisms architected by open source projects. Another objective was to use publicly available big scholarly datasets to have a common basis. We hope to make the study of research paper recommender systems reproducible by utilizing publicly available architectures and datasets.

2 Current Status of Research Paper Recommender Systems

2.1 Research Paper Recommender Systems Related Studies

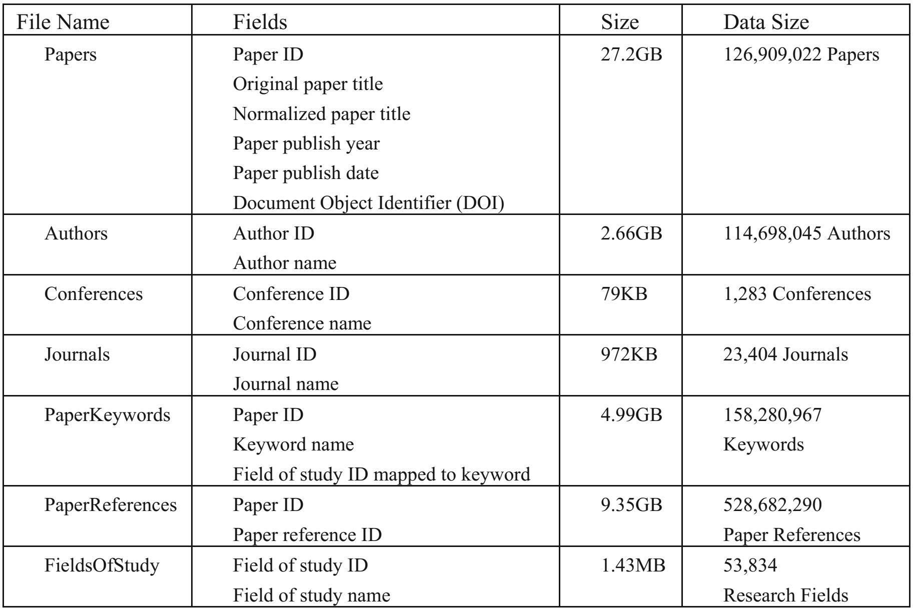
At least 217 research paper recommender systems related papers had been published by 2013 [1]. The main drawbacks of these researches are the unreproducible and incomparable results. The problems of reproducibility and comparability are due to several commonly found issues. The foremost issue is different datasets are used in the recommender systems that make the comparison between studies impractical. The datasets or data sources commonly used include CiteSeer and CiteULike, which account for 43% of all studies reviewed [1]. Most of the other datasets are taken from data sources that are often not publicly available. Another issue is that only a few papers disclose the architecture of their recommender systems. Two architectures for academic information collecting and pre-processing were discussed – system architecture for retrieval papers' PDF files by CiteSeer and the architecture for aggregating data usage from multiple academic data sources [1]. Another study describes an architecture platform that is capable of harvesting big scholarly information and hosting related applications such as citation recommendation [3]. Some recently published research paper recommender systems related articles utilized the afore-mentioned Graph-based and Global Relevance approaches [6] or used the collaborative filtering approach [7].

2.2 Research Paper Recommender Systems from the Perspective of Big Scholarly Data

Big scholarly data may be utilized in literature (research papers') recommendation, collaboration recommendation, and venue recommendation [4, 5]. An architecture platform tailored for big scholarly data analytics has been explored by the CiteSeer research team [3]. A recommender system utilizing Hadoop and Apache Mahout was introduced in a digital library recommender system [8], which makes recommendations based on roughly 2.2 million publications extracted from DBLP dataset.

2.3 Public Available Big Scholarly Datasets

As discussed earlier, the size of datasets used in previous research paper recommender studies ranges from ten thousand to a few million [1, 7–10]. In KDD Cup 2016, Microsoft granted Microsoft Academic Search (MAS) [11] dataset to be used freely in the KDD competition. The MAS dataset includes the Meta information of 126 million academic papers, 114 million authors, and over 528 million citations relationship between these papers. Our recommender system is built on the MAS dataset whose schema is shown in Fig. 1. Open Academic Society (OAS) [12] also has archived a more recent copy of MAS dataset, which includes bibliographical information of over 166 million academic papers. OAS also archived the A Miner dataset, which includes information on more than 154 million academic papers. Semantic scholar, which is funded by Microsoft cofounder Paul Allen, also has made their 20 million+ bibliographical dataset publicly available.



File Name	Fields	Size	Data Size
Papers	Paper ID Original paper title Normalized paper title Paper publish year Paper publish date Document Object Identifier (DOI)	27.2GB	126,909,022 Papers
Authors	Author ID Author name	2.66GB	114,698,045 Authors
Conferences	Conference ID Conference name	79KB	1,283 Conferences
Journals	Journal ID Journal name	972KB	23,404 Journals
PaperKeywords	Paper ID Keyword name Field of study ID mapped to keyword	4.99GB	158,280,967 Keywords
PaperReferences	Paper ID Paper reference ID	9.35GB	528,682,290 Paper References
FieldsOfStudy	Field of study ID Field of study name	1.43MB	53,834 Research Fields

Fig. 1. Partial schema of the MAS dataset

3 Research Paper Recommender Systems for Big Scholarly Data

3.1 The Author/Paper Utility Matrix for Recommender Systems

We utilized the widely used CF mechanism in our recommender systems for several reasons. Firstly, it is a mechanism that has been implemented on many platforms, including the Spark's scalable machine learning library (MLib), which was adopted by us. CF mechanism requires the interaction data between users and items to make recommendations. The traditional interaction data between users and items in a CF-based system are the explicit rating scores given to items by users. However, the implicit ratings given by users are usually infrequent and sparse, making the CF mechanism inoperative in some circumstances. To mitigate the problem of data sparsity, implicit interaction data between users and items are utilized. The implicit interaction data generally refers traces of data left unconsciously by users when they interact with items, such as web browsing logs or purchasing records [13]. In the context of research paper recommender systems, we postulate the behavior of citing or referencing academic papers approximates the explicit rating behavior. A citation is a conscious action made by an author. However, when an article is cited multiple times, it does not necessarily mean the article is regarded highly by an author. The main motivations of citing a paper were categorized as: (1) Perfunctory- an acknowledgement of some other relevant works have been performed; (2) Organic- facilitating the understanding of the citing article; (3) Conceptual- connecting a concept or theory that is used in the citing article; (4) Operational- referring the tools and techniques used in the citing article; (5) Evolutionary- the citing articles built on the foundations provided by the cited article [14]. It is fair to say that a cited article provides some utility to an author just like the enjoyment utility an entertaining item (e.g., a movie) to a viewer. Although the motivation for citing an article may differ, the aggregated citation count recorded by a paper is still regarded as a reliable measure of academic impact [15]. In line with this, we take the accumulated citation counts to an article as the proxy of the preference rating of an item. A higher citation count is equivalent to a higher preference rating. Analogous of a user/item preference rating matrix required by the CF recommender algorithm, an author/paper utility matrix is built, whereas authors as rows and articles as columns entries, respectively. The citation count an article received from an author is listed in the corresponding preference entry in the matrix. A simplified author/paper utility matrix is shown in Fig. 2. The paper-citation bibliographical data also has been utilized differently in other studies. For instance, a paper is regarded as a user and a citation is treated as an item in several studies to construct a paper-citation relations matrix [7, 16] for CF processing. However, instead of the more informative citation counts, only a binary relationship (a paper is cited or not) could be represented by this approach. Another study built the recommender mechanism using some graph-based operations over the citation network, which is derived from the paper-citation data [10].

	Paper 1	Paper 2	Paper 3	Paper 4	...	Paper m
Author 1		5	1	7	...	
Author 2	1		1	6	...	1
Author 3		1	8	2	...	1
Author 4	1		1	1	...	
...	...	...	...	...	...	...
Author n			5	4	...	

Fig. 2. An Author/Paper Utility Matrix. The citations count a paper received from an author stored in a cell in the matrix. The citation count is obtained by summing the total number of times a paper is cited by an author. Empty cells indicate no citation received.

3.2 The Architecture of BSD Capable Recommender Systems

Since the already massive scholarly data is expected to grow at an even faster pace, the capability of processing large dataset is now essential for recommender systems. The proposed platform should be capable of hosting massive and fast accumulating data, and supplementing mechanisms to facilitate efficient BSD analytics. In light of the considerations above, the Berkeley Data Analysis Stack (BDAS) [17] was selected as the main constituent for our BSD recommender systems. The BDAS stack includes a computer cluster manager (Mesos) that enables efficient resource virtualization and sharing across distributed applications and frameworks. Mesos supports Hadoop, Spark, and other applications through a dynamically shared pool of computing and storage resources. The architecture of our BSD-based recommender systems is shown in Fig. 3. The MAS dataset is parsed from its original text format and stored in HDFS format. The recommender system is implemented in Scala programming language utilizing the Alternating Least Squares (ALS) [18] algorithm provided by the Spark MLib. The data in the author/paper utility matrix are divided into 80/20% for training and test data, respectively. ALS is then applied to the training data iteratively to derive the low-rank matrices combination that have a minimum Root Mean Square Error (RMSE). The RMSE of test data is then calculated by applying the resulting low-rank matrices. The RMSE values computed from the training and test data are compared to check if an overfitting occurred. We may adjust the regularization hyper-parameter lambda and rerun ALS to fix the overfitting problem. To recommend research papers for a designate scholar, we just need to locate his corresponding row in the ALS-processed author/paper utility matrix. The values in the selected row correspond to the utility/preference rankings of the scholar. From this row, we then choose N entries with the highest values, which correspond to N highest-ranked candidate papers for Top-N [19] recommendation. The Top-N recommendations for a research field are obtained by summing the values from columns corresponding to papers in the research field from the utility matrix. The research field attribute of a paper is derived from the PaperKeywords and FieldOfStudy files in the MAS dataset.

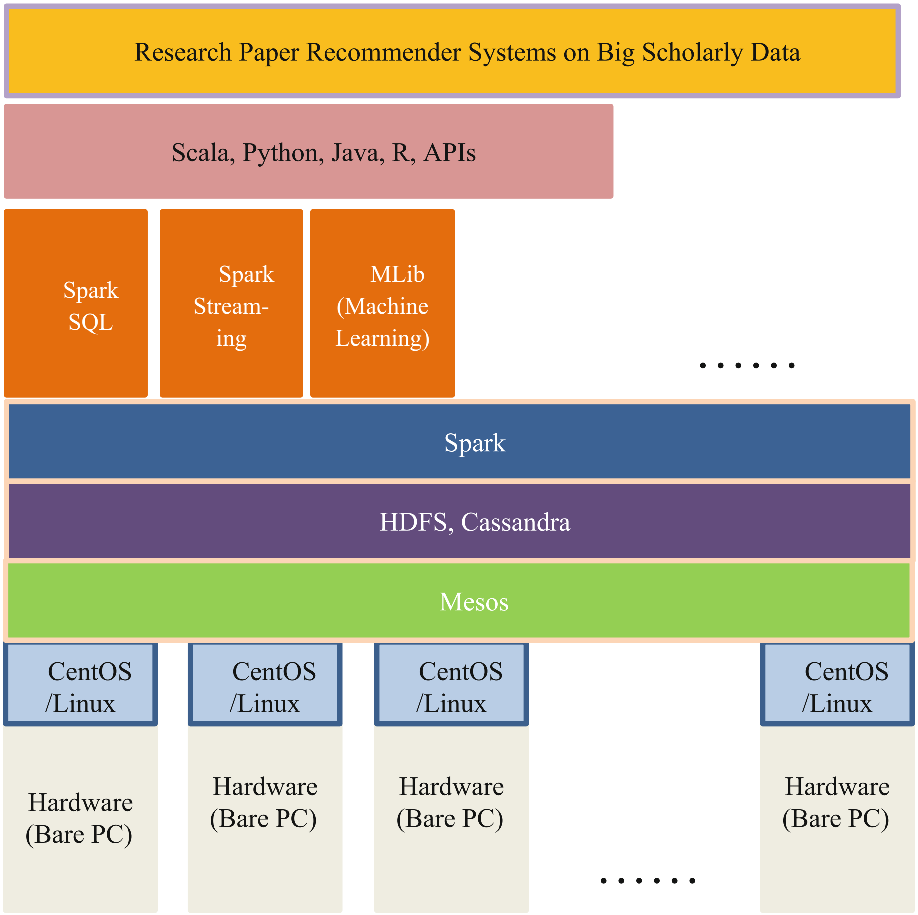


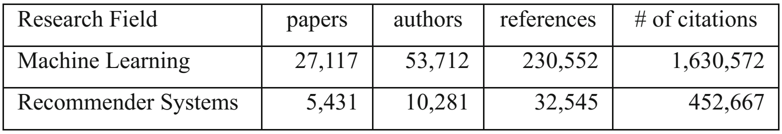
Fig. 3. The architecture of the recommender systems

4 Performance Evaluation of the Research Paper Recommender Systems

We evaluate the system performance using the offline metrics – recall. Since recall only considers the positively rated articles (cited articles in our case) within the Top-N, a high recall rate with lower N signified a better system [20]. For each author, the recall is calculated as follows:

$$recall = \frac{\text{number of articles the author cited in TopN}}{\text{total number of articles the author cited}}$$

We extracted datasets for two research fields from the MAS dataset, the information on the two datasets are shown in Fig. 4.



Research Field	papers	authors	references	# of citations
Machine Learning	27,117	53,712	230,552	1,630,572
Recommender Systems	5,431	10,281	32,545	452,667

Fig. 4. The paper column stores the number of papers published in the research field of machine learning and recommender systems, respectively. Taking the recommender systems field as an example, it includes 5,431 papers that were authored or co-authored by 10,281 distinct scholars and contained 32,545 references. There are 452,667 citations recorded between authors and papers (including papers' references) in the recommender systems research field.

Table 1. The recall rate of recommender systems research field

Author ID	Number of hits	Recall rate
76666523	7	0.35
72694593	12	0.6
77527215	6	0.3
76545162	7	0.35
71946686	8	0.4

We then randomly selected five authors from each field and calculated the recall rate. The recall rate here is the percentage of overlap between the top 20 recommended papers and the 20 most cited papers by the author. The recall rates range from 0.3 and 0.6 as shown in Table 1.

With the completely filled author/paper utility matrix computed by ALS, we are able to find the Top-N most recommended papers of a research field. The 20 most recommended papers in the recommender systems research field (Table 2) are obtained by summing the columns of the utility matrix of the recommender systems research field and retrieving the 20 columns with the highest summation values.

Table 2. The Top 20 papers in the recommender systems research field

Paper ID	Paper Title
10C9E0EA	Hybrid Recommender Systems: Survey and Experiments
7A283611	Latent Semantic Models for Collaborative Filtering
7A6FB77C	Matrix Factorization Techniques for Recommender Systems
7C54E0A8	An Algorithmic Framework for Performing Collaborative Filtering
7DC9036C	Empirical Analysis of Predictive Algorithms for Collaborative Filtering
7EA2B2D5	Social Information Filtering: Algorithms for Automating Word of Mouth
7537398E	Using Collaborative Filtering to Weave an Information Tapestry
757BB126	Evaluating Collaborative Filtering Recommender Systems
7FAE89BB	Item-based Top- N Recommendation Algorithms
7F3B2BC5	Explaining Collaborative Filtering Recommendations

(*continued*)

Table 2. (*continued*)

Paper ID	Paper Title
7F6B27CB	A Framework for Collaborative, Content-Based and Demographic Filtering
76FCDFDA	Analysis of Recommendation Algorithms for E-commerce
77270A42	GroupLens: Applying Collaborative Filtering to Usenet News
79018AC7	Recommending and Evaluating Choices in a Virtual Community of Use
79BABCCB	Item-based Collaborative Filtering Recommendation Algorithms
79CBDC59	Fab: Content-based, Collaborative Recommendation
80A853E8	Methods and Metrics for Cold-start Recommendations
80B12C04	Amazon.com Recommendations: Item-to-item Collaborative Filtering
80745098	GroupLens: An Open Architecture for Collaborative Filtering of netnews
81757DC2	Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-art and Possible Extensions

5 Conclusion

We demonstrated the feasibility of developing a BSD-capable recommender system that is capable of making personalized recommendations. We may recommend the Top-N papers for an author based on his profile, which is derived from references listed in the articles authored by him or her. In addition, we are able to recommend the Top-N papers in a research field based on the aggregated authors' profiles in the field. The study could not have been done without the MAS dataset contributed by Microsoft. The rich meta-information included in the MAS dataset has made feasible many previously unthinkable analyses. Instead of painstakingly harvesting the vast scholarly data ourselves (as seen in many previous studies), academia should vigorously utilize the vast and rich datasets donated by the industry. We could better use our time and effort to develop novel applications from the publicly available big scholarly datasets compiled by the Open Academic Society or Semantic Scholar.

References

1. Beel, J., et al.: Research-paper recommender systems: a literature survey. Int. J. Digit. Libr. **17**(4), 305–338 (2016)
2. Rich, E.: User modeling via stereotypes. Cogn. Sci. **3**(4), 329–354 (1979)
3. Wu, Z., et al.: Towards building a scholarly big data platform: challenges, lessons and opportunities. In: Proceedings of the 14th ACM/IEEE-CS Joint Conference on Digital Libraries, London, United Kingdom, pp. 117–126. IEEE Press (2014)
4. Khan, S., et al.: A survey on scholarly data: From big data perspective. Inf. Process. Manag. **53**(4), 923–944 (2017)
5. Xia, F., et al.: Big scholarly data: a survey. IEEE Trans. Big Data **3**(1), 18–35 (2017)
6. Sesagiri Raamkumar, A., Foo, S., Pang, N.: Using author-specified keywords in building an initial reading list of research papers in scientific paper retrieval and recommender systems. Inf. Process. Manag. **53**(3), 577–594 (2017)

7. Haruna, K., et al.: A collaborative approach for research paper recommender system. PLoS ONE **12**(10), e0184516 (2017)
8. Ismail, A.S., Al-Feel, H.: Digital library recommender system on Hadoop. In: Proceedings of the 2015 IEEE 4th Symposium on Network Cloud Computing and Applications, pp. 111–114. IEEE Computer Society (2015)
9. Xia, F., et al.: Scientific article recommendation: exploiting common author relations and historical preferences. IEEE Trans. Big Data **2**(2), 101–112 (2016)
10. Son, J., Kim, S.B.: Academic paper recommender system using multilevel simultaneous citation networks. Decis. Support Syst. **105**, 24–33 (2018)
11. Sinha, A., et al.: An overview of microsoft academic service (MAS) and applications. In: Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, pp. 243–246. ACM (2015)
12. Open Academic Society (2017). https://www.openacademic.ai/. Accessed 3 Jan 2018
13. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, pp. 263–272. IEEE Computer Society (2008)
14. Cano, V.: Citation behavior: classification, utility, and location. J. Am. Soc. Inf. Sci. **40**(4), 284–290 (1989)
15. Lutz, B., Hans-Dieter, D.: What do citation counts measure? A review of studies on citing behavior. J. Doc. **64**(1), 45–80 (2008)
16. McNee, S.M., et al.: On the recommending of citations for research papers. In: Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work, New Orleans, Louisiana, USA, pp. 116–125. ACM (2002)
17. Singh, D., Reddy, C.K.: A survey on platforms for big data analytics. J. Big Data **2**(1), 8 (2014)
18. Zachariah, D., et al.: Alternating least-squares for low-rank matrix reconstruction. IEEE Signal Process. Lett. **19**(4), 231–234 (2012)
19. Cremonesi, P., Koren, Y., Turrin, R.: Performance of recommender algorithms on top-n recommendation tasks. In: Proceedings of the Fourth ACM Conference on Recommender Systems. ACM (2010)
20. Wang, C., Blei, D.M.: Collaborative topic modeling for recommending scientific articles. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, California, USA, pp. 448–456. ACM (2011)