Can we do better than Co-Citations? - Bringing Citation Proximity Analysis from idea to practice in research article recommendation

Petr Knoth and Anita Khadka

The Open University, UK
{petr.knoth, anita.khadka}@open.ac.uk

Abstract. In this paper, we build on the idea of Citation Proximity Analysis (CPA), originally introduced in [1], by developing a step by step scalable approach for building CPA-based recommender systems. As part of this approach, we introduce three new proximity functions, extending the basic assumption of co-citation analysis (stating that the more often two articles are co-cited in a document, the more likely they are related) to take the distance between the co-cited documents into account. Asking the question of whether CPA can outperform co-citation analysis in recommender systems, we have built a CPA based recommender system from a corpus of 368,385 full-texts articles and conducted a user survey to perform an initial evaluation. Two of our three proximity functions used within CPA outperform co-citations on our evaluation dataset.

Keywords: Citation Proximity Analysis, Co-Citation Analysis, Recommender System, Information Retrieval

1 Introduction

The number of scholarly articles is increasing exponentially every year, according to Gipp and Beel [1,2] by 3.7%, Bornmann and Mutz claim 8%- 9% (up-to 2010) [3]. This creates challenges for researchers to stay in touch with new relevant articles in their domain.

West et al. [4] state that searching for a particular paper knowing that it exists, has become trivial except for the pay-wall. Searching for (unknown) but relevant papers is a challenging task that is at the very centre of the research process (for example, the task of reviewing the state-of-the-art in a particular domain). To researchers, recommender systems can help them to stay in touch with the latest relevant papers in their field. To authors, recommender systems can help cater their papers to the relevant audiences resulting in an increased number of reads and therefore more effective dissemination of knowledge.

In the past, many academic recommender systems unless developed by publishers or corporations that negotiated access to scientific literature, faced several limitations. For instance, limitations of machine access to the full texts of papers and sometimes even the citation information. Consequently, full-text features have been so far relatively unexplored. Most of the recommender systems

that make use of citations do so purely by preferring articles with higher citation counts [5]. However, such approach makes it less likely to exhibit serendipity, as novel papers are rarely recommended. Hence, there is an opportunity for better exploitation of both citation information as well as the full-text in academic recommender systems. In this paper, we explore the use of citation proximity, following the assumption that papers that tend to be closely co-cited within the full-texts are likely to be related.

The concept of Citation Proximity Analysis (CPA) was developed in 2009 by Gipp and Beel [1] and is based on the Co-Citation Analysis (Co-Citation) approach. The main hypothesis of CPA is "the closer the documents are co-cited, the more related they are". In their work in [1], Citation Proximity Index (CPI) is computed as follows: if two documents are co-cited in a sentence level then the CPI value will be 1 and if they are cited in different paragraphs then their CPI value will be $\frac{1}{2}$. Similarly, if documents are co-cited in different chapters, same journals, same journals but different edition then CPI value will be $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$ respectively. Finally, the CPA value is the summation of all the proximities co-citations of the co-cited documents [1,6]. Despite this method being designed, it has not yet been implemented and evaluated fully in a research paper recommender system. Although, Beel et al. in [7] used the concept of CPA for web page recommendation using links in the websites.

In our work, we propose three new proximity functions, defined in Section 3.3, which use absolute CPI value, i.e character counts between co-cited documents rather than arbitrary CPI values as proposed in [1], to compute the CPA score. We implement and deploy these methods within a recommender system and evaluate these methods against the co-citation baseline, which does not use proximity information of citations.

Figure 1(a) shows CPA conceptualised by Gipp and Beel [1] while Figure 1(b) is our interpretation of CPA approach. In the Figure 1(b), *Document A*, *Document B* and *Document C* are the cited documents in the *Citing Document*; $d12$ is the distance between co-cited documents *Document A* and *Document C*, and $d23$ is the distance between *Document C* and *Document B*. Proximities between all the co-cited documents are calculated by counting the characters between each other.

This paper is structured as follows. We first introduce CPA and other relevant approaches to building academic recommender systems in Section 2. We then describe our method in detail, including the description of the tested proximity functions, in Section 3. Finally, in Section 4, we present the data we used to build the recommender, the evaluation experiment and the results.

2 Related Work

Recommender systems suggest relevant and useful information to match the need of its users. These systems are popular in both commerce and academia. Over the years, various metrics and approaches such as Collaborative Filtering (CF) [8], Content Based Filtering (CBF) [9] [10], Graph based recommendations [11][12]

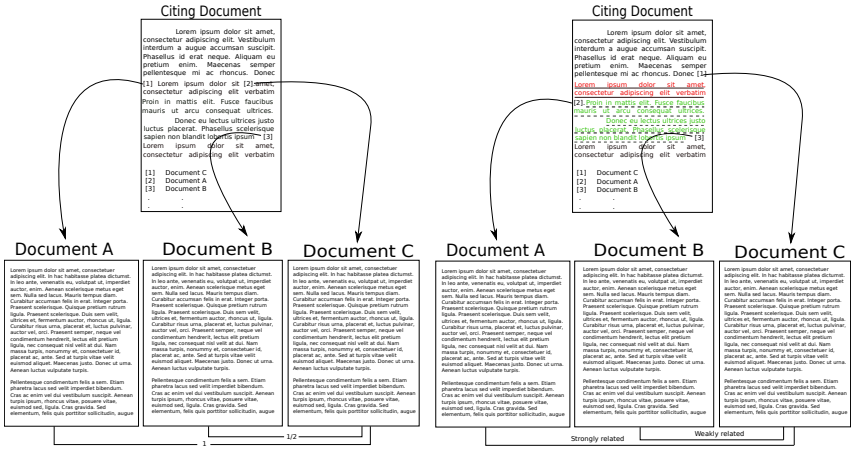


Fig. 1. (a) Citation Proximity Analysis (CPA) conceptualised by Gipp and Beel [1] (b) Citation Proximity Analysis (CPA). Length of solid underlined Red text signifies d12 and Length of dashed underlined Green text signifies d23. (Best viewed in colour)

and different citation based concepts have been implemented and evaluated for scholarly recommendation, either using full-texts or with meta-data. While CF is the state-of-the-art approach for recommending items in commerce, it is also prone to some limitations such as the cold start problem, i.e. the need to have good coverage of ratings. In the domain of scholarly papers, it is typically difficult to obtain ratings. Agarwal et al. [13] argue that the research paper domain has relatively less users compared to the large number of online research papers. This introduces high dimensionality and sparsity, which performs poorly when algorithms such as k nearest neighbour and CF are applied. To combat this issue, their Subspace Clustering Algorithm (SCuBA) approach reduces the dimensionality of the subspace [13]. Sugiyama and Kan in [14] used CF to discover the potential citation papers for a document by creating user profiles from the researcher's list of publication. They also showed that "Conclusion" section weight more than other section for computing effectiveness of the paper. However, Nascimento et al. [9] argued that title of any document weighs more.

Another popular approach is citation based approach, such as, Co-Citation which was proposed by Small [15] and Marshakova [16] separately in the 70s. This approach is a well-established in scholarly recommendation system by now. West et al. [4] have proposed an automation system which has remarkable success over the system "Co-download" which uses collaborative filtering and the "Co-Citation" system. Although, results are remarkable it may differ in different databases and result could be different in cross-disciplinary database. Furthermore, Tran et al. [17] followed the concept of CPA and used graph based similarity measure to demonstrate that documents are more related in Sentence Level

Co-Citation than Paper Level Co-Citation. Gipp et al. [6] introduced a hybrid research paper recommender system by using the concept of in-text Impact factor (ICFA) and in-text citation distance analysis (ICDA). Similarly, Schwarzer et al. [18] used the concept of CPA for articles recommendation using links (*SeeAlso* links from wikipedia articles) instead of citations for article recommendations. They claimed that citation based approaches have different strength to text-based approach like More Like This (MLT) and suggested that combining them could supersede one's caveat with other's advantages. And, Gipp et al. [19] evaluated and analysed citation based approach and compared with character based approach to detect plagiarism and showed citation based approach provided preferable results than character based approach.

Researchers have tried and tested citation-based approaches which are proving in some situations, such as for expert users, more powerful than purely text based approaches. Consequently, it is worthwhile to further explore the role of citation proximity in recommender systems based on co-citations.

3 Method

The design of our CPA-based recommender system consists of five modules (Extracting Citation Information, Citation Information Normalisation, Sparse Matrix, Citation Proximity Analysis, Recommendation) depicted in Figure 2. We start by extracting citation information, including the positions of citation anchors in the body of the full-texts. We then normalise the extracted reference strings (typically found at the end of each paper) trying to detect and merge those referring to the same canonical document. The output of this process is a sparse square matrix where rows and columns correspond to unique references found in the full-texts of research papers in the original collection. These unique references form the set of recommendable items. Each cell of the matrix contains all the co-occurrences of the corresponding references in any of the research papers in the original collection, including their character positions. The information stored in each cell is passed to the CPA component which applies a proximity function to produce a CPA value.

To produce recommendations for a given paper reference, it is necessary to look up a corresponding row (or column), calculate CPA values for each non-empty cell and select n references with the highest CPA scores as the recommendations. Next sections describe the process in more detail.

3.1 Extracting citation information

The aim of this component is to:

- extract reference strings, typically appearing at the end of research papers,
- identify and parse the reference structure, such as article title, authors, publication year or DOI, of each reference and
- detect and extract the character offset of each citation occurrence (citance) on the body of the research paper.

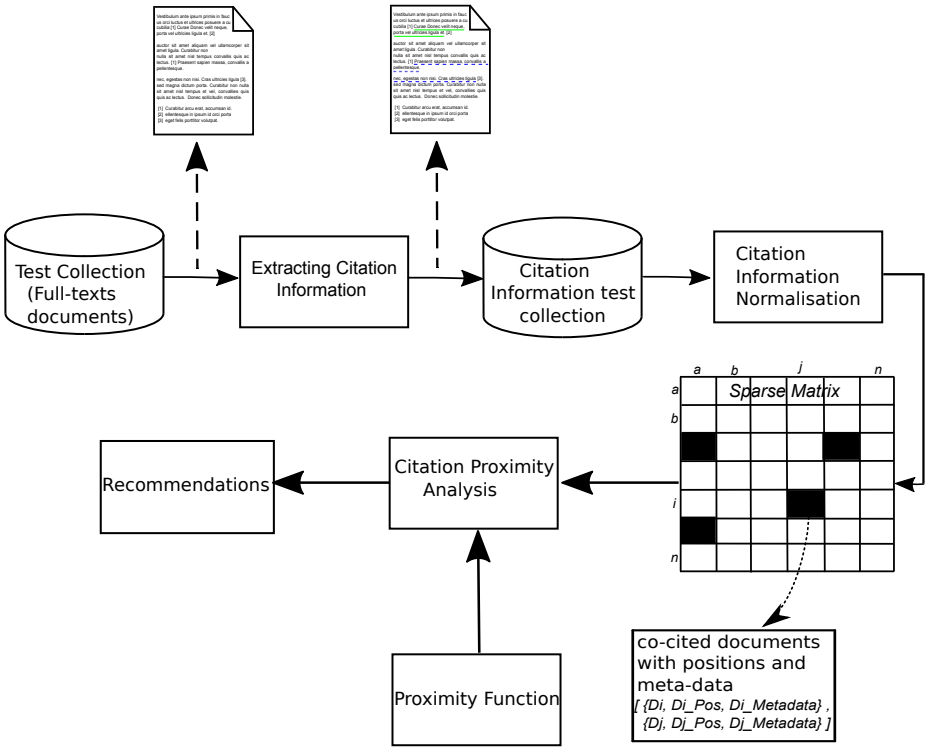


Fig. 2. Model diagram of the process carried out for recommending research articles

The input of this component is the full text of a research paper and the output is a tuple:

(referenceId, title, authors [], characterOffsets [], yearPublished, sourceId) (1)

where *referenceId* is an identifier of each reference string, *title* is the title of the referenced article, *authors* is an array of author name strings, *characterOffsets* is an array of offsets of the citances in the body of the full text for the given reference, *yearPublished* is the year of publication and *sourceID* is a unique identifier of the full-text research paper from which this reference was extracted.

3.2 Citation information normalisation

The citation normalisation component takes as an input, a dataset of the reference tuples defined in the Equation 1, deduplicates them and represents them as a sparse matrix. As deduplication is not the key focus of this paper, we use a naive deduplication method that targets precision at the expense of recall. Firstly, we removed the special characters and spaces from the title of each documents and grouped records with the same title, publication date and at least

one matching author and singled out only one record from the group. By doing this, we ended up only unique documents in the dataset.

After normalisation, we represent the output as a *Citation-Positions* matrix. Each cell $V_{i,j}$ in this matrix contains character offset information of all the citances where a normalised reference i co-occurs with a normalised reference j in a given source document. This is all that is needed as input for the CPA module which then calculates the CPA value based on a given proximity function.

3.3 Proximity functions

The CPA proximity function takes as an input, a set of character offset distances and produces a single value. The higher the value the higher the relevance. In a research paper, a reference can be cited multiple times leading to a set of pair distances for the co-cited pair. Additionally, references can be co-cited in multiple source documents. The intuition behind the proximity metric is that higher number of co-citations as well as closer proximity should lead to an increased relevance.

In this work, we have defined and experimented with the following proximity measures: *MinProx*, *SumProx* and *MeanProx* described below. Our baseline co-citation method, which does not use any proximity information, can be in this framework defined as:

$$cocit^{ab}_{baseline} = |Doc|_{a \in Doc \wedge b \in Doc}, \tag{2}$$

This means that the number of co-citations is defined by the number of source documents where references a and b co-occur.

3.3.1 MinProx: Uses only the distance of the closest co-occurrence in the denominator. During distance computation, one of the hypothesis is the distance between co-cited documents are never Zero or One. For example, the extreme case of citations being cited together will be like $[X, Y]$ and this will always have a separator character between them. If reference "X" has character offset 102 and reference "Y" has character offset 104 then the distance will be 104-102=2.

As, in the following proximity functions (3, 4, 5), logarithm is applied for smoothing large distances. The nominator is equal to the baseline co-citation measure.

$$prox^{ab}_{Min} = \frac{|Doc|_{a \in Doc \wedge b \in Doc}}{\log(min\{d^{ab}_1, ..., d^{ab}_n\})} \tag{3}$$

where d^{ab}_1 denotes the first distance between the co-cited references a and b and d^{ab}_n denotes the last distance between them.

3.3.2 SumProx: Uses the sum of the logs of all the co-cited distances in the denominator.

$$prox^{ab}_{Sum} = \frac{|Doc|_{a \in Doc \wedge b \in Doc}}{\sum_{i=1}^{n} \log(d_i^{ab})} \quad (4)$$

where d_i is the i^{th} distance between the co-cited documents a and b

3.3.3 MeanProx: Uses the log of the mean of all the co-cited distances in the denominator.

$$prox^{ab}_{Mean} = \frac{|Doc|_{a \in Doc \wedge b \in Doc}}{\log(mean\{d_1^{ab}, ..., d_n^{ab}\})} \quad (5)$$

where d_n is the last distance between the co-cited documents a and b.

4 Experiments

To evaluate CPA against the co-citations baseline, we have developed and "trained" a recommender system, as described in Section 3, on a sample collection scientific documents from CORE [20] [1]. The evaluation dataset consisted of a set of recommendations retrieved by each evaluated variation of the recommender in response to different queries (research papers for which recommendations should be produced). Several human judges were asked to provide binary relevance judgments which form the evaluation ground truth. We will now provide more details of the experimental setup, evaluation dataset and results.

4.1 Experimental system

We used GeneRation Of BIbliographic Data (GROBID) [21] to convert research papers in the PDF format into the Text Encoding Initiatives (TEI) format from which we extracted the required citation information as specified in Section 3.1. The processing of this collection took about an hour and a half on a quad core system with 20 GB of memory.

As expected, GROBID could not successfully process all the documents (due to PDFs that were scans, badly encoded PDF files or citation extraction failing on valid PDFs). We were confident of 368,385 documents which yielded citations information along with their positions and used these in our experiment. The resulting set consisted of 6,609,147 references. This means we have obtained on average 18 references per document. Figure 3(a) shows the probability distribution of the number of citation mentions (citances) in a document while Figure 3(b) shows the probability distribution of the number of references in a document.

For an efficient implementation of the subsequent components, i.e. normalisation and CPA calculation, we made use of the MapReduce paradigm and implemented the solution using the data flow language Apache Pig. The normalization step took approximately 2 minutes with 100 parallel processes and

[1] `https://core.ac.uk/services#datasets`

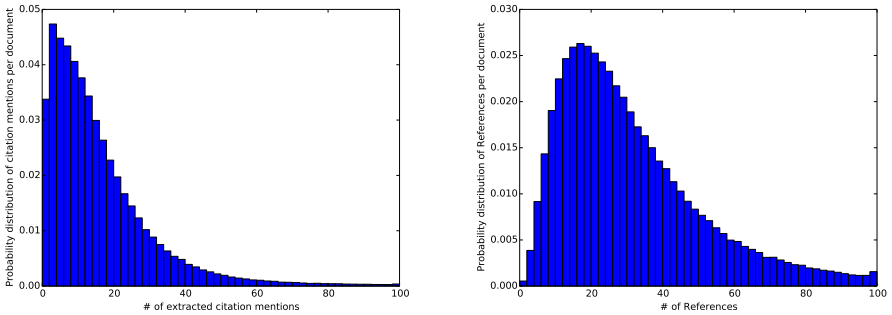


Fig. 3. (a) Probability distribution of the number of extracted citation mentions (citances) in a document, (b) Probability distribution of the number of references in a document.

resulted in a citation-positions sparse matrix containing 142,157,561 co-cited pairs. Figure 4 shows the distribution plots of relevance score produced by each proximity functions, we have defined along the baseline standard's results.

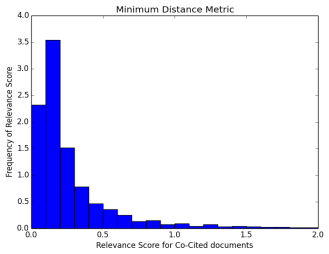
4.2 Evaluation dataset

Evaluating recommender system is generally a challenging task due to the variety of criteria and goals recommendation methods can be developed with, such as recency, serendipity, relevance, etc. Consequently, some evaluation datasets can work better with one metric or task while other's may not [22,23]. We created an evaluation dataset on the basis of the relevance of recommendations to the target evaluator's expertise. As CPA is still in its relative infancy, our goal was to create a small-scale pilot evaluation initially. If encouraging results are produced by the tested method, this will be a signal for us to extend this study and re-evaluate on a larger dataset.

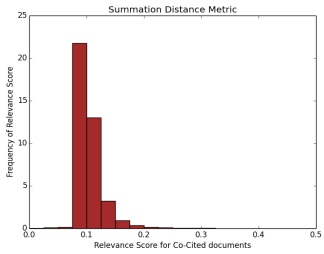
For the evaluation, we randomly selected 6 sample documents from the dataset from the area of "Computer Science" (specially "Data mining" and "Information Retrieval") with which the annotators were familiar with. Ten annotators (survey participants) from computer science department (working on "Data mining" and "Information Retrieval") were asked to provide binary relevance judgments on each recommendation offered by each evaluated system. As, we had 4 evaluated metrics, 5 recommendations for each sample document and ten participants, this yields $6 * 5 * 4 * 10 = 1,200$ individual relevance judgments.

4.3 Results

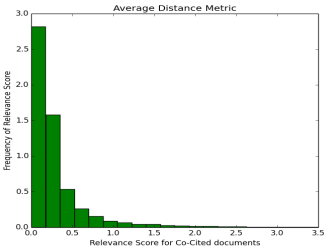
We have calculated precision at 3 different precision levels as shown in Table 1. Our experimental results indicate that proximity information helps in producing



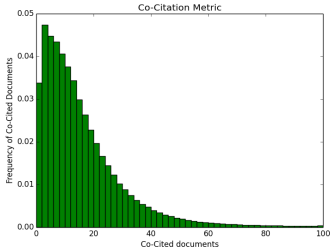
(a) proximities using minimum co-cited Distance metric



(b) proximities using summation of co-cited Distances metric



(c) proximities using mean of co-cited Distances metric



(d) proximities using frequency of co-citation metric

Fig. 4. Probability Distribution Functions of the proposed proximity functions and baseline measure.

better recommendations than the baseline co-citation approach. More specifically, out of the three proximity functions two, $SumProx$ and $MeanProx$, outperform the $Baseline$. The improvement over the tested dataset over the baseline for P@5 (from 0.27 to 0.34) corresponds to a more than 25% improvement.

To assess the subjectivity of the task, we have also calculated inter-rater reliability statistic to weight the agreement between the contributors. To do so, we have used Fleiss's κ as follows:

$$\kappa = \frac{\bar{P} - \bar{P}_e}{1 - \bar{P}_e} \tag{6}$$

where, $\bar{P}_e$ denotes the observed agreement and $\bar{P}$ denotes the probability of chance agreement. Hence, $(1 - \bar{P})$ is the degree of agreement which is obtainable by chance and $\bar{P} - \bar{P}_e$ gives the degree of agreement which is actually obtained. For all the sample data and its recommendations, we have observed $\kappa = 0.25$ suggesting a fair agreement.

Metric	Precision@1	Precision@3	Precision@5
Baseline	0.29	0.27	0.27
MinProx	0.20	0.25	0.25
SumProx	**0.32**	0.33	**0.34**
MeanProx	**0.32**	**0.34**	0.30

Table 1. Precision at three different recall levels

5 Discussion and future work

There are several things we would like to address in our future work as this is an initial practical observation of the CPA concept. Firstly, our current proximity functions are based on working with absolute character offsets, i.e. our distance measure is a character distance of the citances. In the future, we would like to extend our work by experimenting with distance measures that reflect the lexico-syntactic structure of language. For example, we could use information on whether two papers have been co-cited in the same sentence clause, sentence, paragraph, section, table, etc. Additionally, we would also like to compute the CPI values as conceptualised by Beel and Gipp in [1] and compare the results by extending the combinations of the metrics like multiplication of CPA and Co-Citation.

Secondly, our current method treats all co-citations equal. However, it would be interesting to explore how the concept of "authority" might be applied in this problem. For example, if one paper is co-cited (or even compared/contrasted) with another highly significant work (e.g. famous authors, policy document or a high value determined by any particular scientometric method) or if the co-citation is present in a highly significant work, this information should influence the strength of this co-citation evidence and be effectively used in the recommender.

Thirdly, more work is needed to understand the impact of document length on co-citation analysis approaches as long documents, like theses or books, produce significantly higher numbers of co-citations than shorter documents (short, long papers, demo), hence they have a higher impact on the results of the recommender.

Finally, we would also like to use machine learning algorithms using position of the citation as one of the features to improve the weighing process. However, to do so, we will be needing big dataset with the ground truth of recommendation results for training the system so collecting such dataset will be one of the major hurdles.

6 Conclusion

In this paper, we performed an experiment to convert the concept of CPA into practice and benchmark this approach against co-citation Analysis. We introduced three different proximity functions used within the CPA method and

developed a highly scalable CPA implementation that runs on a cluster using the MapReduce paradigm. Our initial results suggest that CPA can provide better performance in recommender systems than the co-citation method. More specifically, two of our proximity functions outperformed the baseline co-citation approach on our dataset, the SumProx function by a margin of more than 25% for precision@5. However, a larger evaluation dataset is needed to confirm these results.

References

1. Bela Gipp and Jöran Beel. Citation proximity analysis (cpa)-a new approach for identifying related work based on co-citation analysis. In *Proceedings of the 12th International Conference on Scientometrics and Informetrics (ISSI'09)*, volume 2, pages 571–575. Rio de Janeiro (Brazil): International Society for Scientometrics and Informetrics, 2009.
2. Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. Research-paper recommender systems: a literature survey. *International Journal on Digital Libraries*, 17(4):305–338, jul 2015.
3. Lutz Bornmann and Rüdiger Mutz. Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. *Journal of the Association for Information Science and Technology*, 66(11):2215–2222, 2015.
4. Jevin D. West, Ian Wesley-Smith, and Carl T. Bergstrom. A recommendation system based on hierarchical clustering of an article-level citation network. *IEEE Transactions on Big Data*, 2(2):113–123, jun 2016.
5. Norman Meuschke, Bela Gipp, and Mario Lipinsk. Citrec: An evaluation framework for citation-based similarity measures based on trec genomics and pubmed central. *iConference 2015 Proceedings*, 2015.
6. Bela Gipp, Jöran Beel, and Christian Hentschel. Scienstein : A research paper recommender system. In *Proceedings of the International Conference on Emerging Trends in Computing*, pages 309–315, 2009.
7. Bela Gipp, Adriana Taylor, and Jöran Beel. Link Proximity Analysis-Clustering Websites by Examining Link Proximity. *Proceedings of the 14th European Conference on Digital Libraries (ECDL'10)*, 6273(September):449–452, 2010.
8. Chong Wang and David M. Blei. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 448–456, New York, NY, USA, 2011. ACM.
9. Cristiano Nascimento, Alberto H.F. Laender, Altigran S. da Silva, and Marcos André Gonçalves. A source independent framework for research paper recommendation. In *Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries*, JCDL '11, pages 297–306, New York, NY, USA, 2011. ACM.
10. Qi He, Jian Pei, Daniel Kifer, Prasenjit Mitra, and Lee Giles. Context-aware citation recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 421–430, New York, NY, USA, 2010. ACM.
11. Yicong Liang, Qing Li, and Tieyun Qian. *Finding Relevant Papers Based on Citation Relations*, pages 403–414. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

12. Ding Zhou, Shenghuo Zhu, Kai Yu, Xiaodan Song, Belle L. Tseng, Hongyuan Zha, and C. Lee Giles. Learning multiple graphs for document recommendations. In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pages 141–150, New York, NY, USA, 2008. ACM.
13. Nitin Agarwal, Ehtesham Haque, Huan Liu, and Lance Parsons. Research paper recommender systems: A subspace clustering approach. In *Proceedings of the 6th International Conference on Advances in Web-Age Information Management*, WAIM'05, pages 475–491, Berlin, Heidelberg, 2005. Springer-Verlag.
14. Kazunari Sugiyama and Min-Yen Kan. Exploiting potential citation papers in scholarly paper recommendation. In *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '13, pages 153–162, New York, NY, USA, 2013. ACM.
15. Henry Small. Co-citation in the scientific literature: A new measure of the relationship between two documents. *Journal of the Association for Information Science and Technology*, 24(4):265–269, 1973.
16. Irina V Marshakova. System of document connections based on references. *Nauchno-Tekhnicheskaya Informatsiya Seriya 2-Informatsionnye Protsessy I Sistemy*, (6):3–8, 1973.
17. Nam Tran, Pedro Alves, Shuangge Ma, and Michael Krauthammer. Enriching pubmed related article search with sentence level co-citations. In *AMIA Annual Symposium Proceedings*, volume 2009, page 650, 2009.
18. Malte Schwarzer, Moritz Schubotz, Norman Meuschke, Corinna Breitinger, Volker Markl, and Bela Gipp. Evaluating link-based recommendations for wikipedia. In *Digital Libraries (JCDL), 2016 IEEE/ACM Joint Conference on*, pages 191–200. IEEE, 2016.
19. Bela Gipp, Norman Meuschke, and Corinna Breitinger. Citation-based plagiarism detection: Practicability on a large-scale scientific corpus. *Journal of the Association for Information Science and Technology*, 65(8):1527–1540, 2014.
20. Petr Knoth and Zdenek Zgrahal. Core: Three access levels to underpin open access. *D-Lib Magazine*, 18(11/12), nov 2012.
21. Patrice Lopez. *GROBID: Combining Automatic Bibliographic Data Recognition and Term Extraction for Scholarship Publications*, pages 473–474. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
22. Paul Resnick and Hal R Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.
23. Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, January 2004.