Recommending Citations for Academic Papers

Trevor Strohman, W. Bruce Croft, and David Jensen

University of Massachusetts Amherst

Abstract. Substantial effort is wasted in scientific circles by researchers who rediscover ideas that have already been published in the literature. This problem has been alleviated somewhat by the availability of recent academic work online. However, the kinds of text search systems in popular use today are poor at handling vocabulary mismatch, so a researcher must know the words used in relevant documents in order to find them. This makes serendipitous results unlikely.

We approach the problem of literature search by considering an unpublished manuscript as a query to a search system. With this approach, the entire text content of the paper can be used in the search process. We use the text of previous literature as well as the citation graph that connects it to find relevant related material. We evaluate our technique with manual and automatic evaluation methods, and find an order of magnitude improvement in mean average precision as compared to a text similarity baseline.

1 Introduction

Science continues to grow in both numbers of practitioners and depth of knowledge. Unlike the scientific landscape of even a hundred years ago, it is no longer possible for one researcher to understand everything about even a small corner of science. As science continues down this path of specialization, the role of the scientific literature becomes even more important. The scientific paper is the most effective tool that we have for disseminating scientific discovery. It allows specialized, geographically disparate scientists to work together in the process of discovery. In order to maintain the pace of scientific discovery, it is critical that researchers have easy access to past discoveries. Unfortunately, the constant growth of the scientific field makes literature access an increasingly difficult problem.

Searching the scientific literature by computer is not a new idea. The modern computer science researcher has access to general literature search engines like Google Scholar [5], CiteSeer [4] and Rexa [13], as well as publisher-specific engines like those offered by the ACM and IEEE. The best of these engines incorporate not just query text but also some citation features into the document ranking process. The bibliometrics literature provides a series of ideas for ranking the importance and similarity of scientific works.

Most current literature search systems concentrate on short queries that are unlikely to describe the nuance of the user's true information need. In this work, we instead suppose that the user is able to provide the system with a very long query; we assume that the user has already written a few pages about the topic, and is able to submit this document to the search system as the query. With this additional information, we suppose that we can improve the effectiveness of the ranked list of documents.

In this recasting of the problem, we no longer assume that the user wants documents that are topically similar to the query; instead, we assume the user wants documents that the query document might cite. The distinction is important, because relevant citations may take many different forms. Some relevant citations are for recent, topically related work. Other citations are for seminal works or survey articles; these papers show a weaker topical similarity, but provide context about the field to the reader. Finally, authors sometimes cite textbooks and other reference materials as sources of information about algorithms and standard methods.

We have built a system to explore this citation recommendation problem. In the process, we have found that simple text similarity computation is not enough to excel at this task. We show that it is necessary to use graph-based features in the retrieval process to achieve high quality retrieval results, although many seemingly useful features offer little benefit. Our evaluation and results are preliminary, but we provide both automatic and manual evaluations to show the effectiveness of our model. In conclusion, we call for a fresh look at this task by the information retrieval community.

2 Related Work

This work depends on the idea that there is some notion of a correct or optimal set of citations for a paper, or at least there is some partial ordering among bibliographies. Wouters [20] cites evidence that scientists are intensely personal about their referencing behavior, and that the accepted model of citation may change drastically between disciplines. For example, Wouters notes that mathematicians are known for very brief citation lists, while biologists are not afraid to cite exhaustively. Wouters shows how the process of academic citation has resisted many attempts to build a comprehensive theory explaining it.

If citing references is a matter of personal taste, then perhaps we can determine something about the author of a paper from the references they cite. Hill and Provost prove that this is true; they find that a classifier can determine the author of a paper with as much as 60% accuracy using only the list of references at the end of the paper [6]. As may be expected, accuracy rates improve with more training data, so more prolific authors tend to be easier to identify.

Our goal is to represent the scientific process of citation just well enough that we can attempt to influence it. Wouters approaches citation as an anthropologist, trying to define what people do and why. Hill and Provost take a different approach and show that the citation behavior of different authors is different enough to act as a kind of fingerprint. We do not attempt to understand or model this kind of personal behavior in citation. Our ultimate goal is to affect some kind of change in the scientific process, which we believe may be an easier task than fully understanding the present state of scientific writing.

The first step toward emulating the citation process is analysis, and the pioneer of citation analysis is Garfield and the Science Citation Index [3]. Garfield's Science Citation Index was, in effect, a paper version of the publication graph that our system uses. This index linked authors and their papers, and papers with the papers they cite. Once this data was made available, the next logical step was to analyze patterns in the data and to make inferences from that data. In the article we cite here, Garfield discusses the use of citation analysis to judge the quality of academic journals. Garfield suggests that citations per work published is a reasonable measure of the quality of a journal, and that influence in science is concentrated in a few highly influential journals. This work is not so different from what we attempt to do; we want to use this citation data to help us determine what documents might be cited by a query document. Influence is clearly an important latent variable in that equation.

This is not to say that citation data has not been used before; in fact, the field of bibliometrics continues to thrive. White and McCain [19] write that "Bibliometrics is the study of literatures as they are reflected in bibliographies." In bibliometrics, scientific literature is seen as a graph that connects authors, journals, conferences, schools, companies and papers together. By analyzing this graph, researchers hope to determine which authors and journals are most influential, which scientific fields are segregated from one another, and which papers seem to be similar. It is this last category that interests us most.

Much of the recent work on link-based features in retrieval has been focused on web pages. PageRank is the best known link-based web retrieval feature [15]. Page et al. consider a mythical web surfer, randomly moving through the web graph by either following links, or jumping directly to other pages (presumably by typing the address of a page into a web browser). Since this random walk is an ergodic Markov chain, it has a unique stationary distribution; the probability mass of each page in this stationary distribution is its PageRank.

PageRank is a global metric; it does not depend on the query. In contrast, Kleinberg [8] proposes a query-dependent measure known as HITS. The algorithm starts with an initial set of supposed relevant documents, then searches the document graph to find those papers that point to the document set, and those documents that the document set points to. After doing this, the algorithm iterates to determine which papers are hubs, and which are authorities. The authority scores generated are analogous to PageRank, but are query-dependent.

The task we consider can be considered a traditional retrieval task, in which case the link metrics shown above are directly applicable. Another way to consider our task is as a link prediction task; we ask the whether the query document will cite documents in the collection in the future.

Liben-Nowell and Kleinberg [12] consider a variety of graph-based measures in order to predict new links in a social network. Like us, they focus on the graph of academic publications. However, instead of predicting the bibliography of a new document, they try to predict collaborations between existing researchers.

The authors consider a variety of known measures, including PageRank, common neighbors (similar to bibliographic coupling) and the Katz measure. The authors find that while their best algorithms perform far better than their baselines, it is difficult to achieve high accuracy on this collaboration task.

The closest system to the one we describe is CiteSeer [4]. The primary focus of the CiteSeer research was the automatic extraction of citation information from research papers found on the web. However, CiteSeer is a successful academic search engine, and as such is related to our work. The initial query processing done by CiteSeer is a traditional Boolean search. However, CiteSeer also offers a similar documents search, which can either use textual similarity (cosine similarity of TF-IDF vectors), header similarity, or a citation similarity (cosine similarity of TF-IDF vectors of citations, with each citation treated as a term). These different metrics are distinct searches in the CiteSeer interface, and are not combined to form one single ranking metric. The authors do not attempt to evaluate the effectiveness of their similarity searches.

Salton and Buckley [16] survey spreading activation methods for information retrieval. The spreading activation idea comes from models of cognition, where concepts occur on a graph, which are proxies for linked neurons. If one graph node is activated, that activation spreads through graph connections to activate other nearby nodes. In the document retrieval case, if one document is assumed to be relevant, documents similar to that document are also assumed to be relevant.

A different take on the same general idea comes from recent work by Diaz [1]. While Salton and Buckley envision document scores propagating through a graph structure, Diaz instead considers the problem of smoothing a function on a graph. The result is similar; a high score at one graph node is smoothed to its neighboring nodes.

3 Model

Text similarity is the basis of our retrieval model, as in most text retrieval systems. Matching the term distribution between two documents is the basis of document clustering and classification methods, and therefore gives us a strong base to build from. However, we hypothesize that text similarity will not be enough to perform well in this task for two reasons. First, authors are known to create new terminology when writing about new ideas. It stands to reason that two researchers working independently on the same idea will describe that idea using different words and concepts. In order for this kind of search system to be successful, we need to account for this potential change in terminology between papers. Second, text similarity cannot adequately account for important paper attributes like quality and authority.

Because of this, our model also exploits the citation information between papers in the collection. We can think of the papers in the collection as nodes in a directed graph, where the edges are citations of one paper by another. Since we assume the query consists of only text and not citations, it is natural to think of it as a node with no incoming or outgoing edges. We use the text similarity measure as a proxy for actual citation information only for this node; this allows us to approximately place the query in the citation graph.

To do this, our system uses a two stage process to find a set of documents to rank. Let R be the initially empty set of documents to rank. In the first step, the system retrieves the top 100 most similar papers to the query document and adds them to R. In the second step, all papers cited by any paper in R are added to R. In general, this process concludes with a set R that contains 1000 to 3000 documents. Initial experimentation with real academic papers suggested that over 90% of papers that researchers actually cite would be in R at this point. Expanding R with a third step (again adding all papers that are cited by some paper in R) did not appear improve recall.

Publication Year	The year the document was published (normalized by subtracting 1950)
Text Similarity	The similarity of the text of this candidate with the query, as measured by the multinomial diffusion kernel
Co-citation Coupling	The fraction of documents that cite this candidate that also cite documents in the base set
Same Author	Binary feature; true if this document is written by the same author that wrote the query
Katz	The Katz graph distance measure
Citation Count	Number of citations of this document from all documents in the corpus

Table 1. A list of the features used in our experimental model

We then rank the documents in R by the features shown in Table 1. Neither text-based nor citation-based features performed well in isolation. Text-based features are good for finding recent related work, since papers will use the same sort of vocabulary. However, text features are not as good at finding conceptually related work that uses different vocabulary. Textual features are also poor at establishing authority of documents. Citation features are useful for these things, but may do a poor job at coverage (since recent documents may have no citations).

The features are combined linearly to provide a final document score. As shown in both Joachims [7] and Metzler [14], maximizing data likelihood is not an effective way to train a model for high performance on the mean average precision measure. Instead, we use coordinate ascent to find feature weights for our model.

We use two different evaluations to show the effectiveness of our technique. In the manual evaluation, we show that the system is capable of finding reasonable citations for a sample paper from our collection. In the automatic evaluation, we use real research papers with their citation lists stripped, and evaluate our system based on its ability to find the true citation list using the mean average precision metric.

3.1 Model Features

We use publication year as a feature since academic citation tends to focus on recent literature. There is a legitimate debate about whether this is good or not; a focus on recent literature allows a discipline to move forward quickly, but this focus can turn to myopia if important discoveries from older literature are ignored.

Citation count is also used as a feature in our model. Here we use the total count of documents that cite a particular document. This is very similar to an inlink count feature that might be used for web retrieval. Here we are assuming that heavily cited pages are likely to be cited again.

In the bibliometrics literature, bibliographic coupling and co-citation coupling are used as indicators that two documents are similar. Traditionally these are compared using Pearson's r to find a correlation score between the two documents. We instead use the multinomial diffusion kernel recommended by Lafferty and Lebanon [10]. In using this kernel, we assume that bibliographies are generated by multinomial distributions of citations, and then ask whether the bibliographies of two papers are generated by the same distribution. The probability of this is the multinomial diffusion kernel:

$$K(\theta_i, \theta_j) = (4\pi t)^{-\frac{|V|}{2}} e^{-\frac{1}{t}\arccos^2(<\sqrt{\theta_i}, \sqrt{\theta_j}>)}$$

We use this kernel to measure the textual similarity between two documents as well, as done in Diaz [1]. Our decision for using this feature was motivated both by arguments from Lafferty and Lebanon [10] and the availability of code to generate this feature.

The Katz measure [12] is a measure of distance on a graph. Under the Katz measure, two graph nodes are considered close if there are many short paths between them. This gives the following formula:

$$\sum_i \beta^i N_i$$

where N_i is the number of unique paths of length i between the two nodes, and β is a decay parameter between 0 and 1. In this work, we train both the β parameter and the weight of this feature on the document score. All paths are assumed to start from the query document, then go through one of the base set of textually similar documents, then on to documents that those base documents cite.

We chose the Katz measure because Liben-Nowell and Kleinberg found it to be the most useful of the measures they tested. It also gives us some intuition about what the influential works in a discipline are. The citation count metric is useful, but perhaps too coarse. For instance, computer science textbooks may be highly cited sources by students, but are less important sources to researchers.

The Katz measure is computed within the retrieved set of documents, and can be seen as measuring the authority that this set of documents place on each other.

4 Experimental Platform

We built a retrieval system to evaluate the effectiveness of our approach. We created a custom parser for the Indri [17] indexing and retrieval system in order to parse the Rexa [13] corpus, which is described in the next section. All documents were stemmed with the Krovetz stemmer [9]. As documents were indexed by Indri, the system also extracted citation data from the corpus and stored it in a MySQL database.

We computed the text similarity feature using the data stored in the Indri index, and stored these similarity values in the database. The citation count and publication year features were also stored in the database for efficient retrieval by the ranking component. The rest of the features, such as the Katz measure, author feature, and co-citation coupling were computed at system runtime.

All ranking experiments were performed on a 3GHz Pentium 4 desktop running Linux. Training a model on 900 documents takes about an hour, while running an average query takes about a second.

4.1 Rexa

This research is dependent on a corpus of data collected by the Rexa project at the University of Massachusetts [13]. The public face of Rexa is a search engine of scientific literature, like CiteSeer [4] or Google Scholar [5]. Like both of these systems, Rexa finds literature on the Internet in the form of PDF or PostScript documents, and uses probabilistic parsing techniques to extract metadata from these documents, such as title, author and citation information [18]. The result of the extraction process is a database of papers with extracted metadata, as well as automatically extracted author and citation graphs. While most research in the Rexa project has focused on the metadata extraction process, the metadata-enhanced corpus is an important data source for further research in relational learning and information retrieval.

Total paper entries	964,977
Papers with text	105,601
Total number of citations (X cites Y)	1.46 million
Total number of cited papers	675,372

Table 2. Statistics from the Rexa collection used in our experiments

The Rexa corpus continues to grow as new papers are crawled. For this research, we use a snapshot of the data from the middle of 2005, as detailed in

Table 2. At this point, approximately 100,000 papers had been processed. The processing was augmented by bibliographic information from DBLP [11]. This information provides a canonical version of references. This additional data gives the collection an interesting composition; it contains almost 1 million paper references, but only about 100,000 of these papers contain full text and a references list. However, almost 700,000 papers are cited by some paper in the collection, so the citation information in the collection is quite rich.

In both the manual and automatic evaluations, we treated an actual research paper as a query. In the manual evaluation, we manually check papers retrieved for relevance. In the automatic evaluation, we match the papers retrieved with the actual papers cited by the query paper.

5 Manual Evaluation

One possible problem with our automatic evaluation is that it is circular; our system is attempting to improve the citing ability of authors, but we evaluate with the papers that authors actually cite. The documents that we hope our system will find are those that the paper author would not have considered otherwise; but it is precisely these documents that our automatic evaluation strategy will consider not relevant.

A full manual evaluation of retrieval accuracy was not possible, but we did analyze a single retrieval to understand the types of papers our experimental system returns.

The paper we consider in this publication is "Extensible Kernels are leading OS Research Astray," by Druschel, et al. [2]. This paper is a position paper describing why the authors believe that extensible operating systems are an important research tool, but that better alternatives exist for end users. The nature of this paper's argument means that it will necessarily cite a diverse set of papers in systems research, and it does; it cites work on extensible kernels, network stacks, the BSD operating system, and caching systems.

Our system manages to find one of its citations in the top 10 papers retrieved, another 8 in the top 100, and 5 more in the top 500. It does not return 3 of the citations. The resulting retrieval has an average precision of 0.052. By contrast, using the text similarity feature alone finds none of the citations. These numbers are somewhat incomplete in that they ignore the possibility that the system found excellent citations that were simply not used by the original paper's authors. Therefore, we analyzed the top 50 papers retrieved by both our experimental system and the baseline system. Space constraints keep us from printing the titles of the papers retrieved, but we summarize our findings here.

Of the top 10 papers retrieved by our experimental model, four are almost certainly not relevant. The other six are possibly relevant. The top-ranked paper is about a modular network infrastructure that has some parallels to extensible operating system kernels. The fourth, seventh and ninth papers are explicitly about extensible operating systems. The sixth and tenth papers discuss process fault isolation, which is an important component of the author's paper.

The top 10 from the baseline system are vaguely on topic, but the quality appears to be lower than in the experimental model results. Surprisingly, there is no overlap between this top 10 list and the one retrieved by the experimental model. Upon evaluating these documents, we believe only three could be considered relevant to the query.

In the top 50 papers retrieved by the experimental model, we find other possibly good references. We see some common themes in these papers that are shared by the query. For example, seven papers discuss specific modifications to operating systems in order to support specific workloads, which has been a key driving force in extensible kernel research. Five papers discuss virtualization or ways that traditional kernel-level computations can be executed in some kind of protected mode, which is a concept the query paper advocates.

The baseline system retrieves some of the same documents as the experimental model does, but misses some of them as well. In all, we find 12 usable documents in the baseline system results versus 18 from the experimental system.

The findings in this section are merely qualitative, but they serve to support our automatic experimental methodology shown in the next section.

6 Automatic Evaluation

In order to have an objective evaluation of the system, we used an automatic evaluation. To do this, we considered a particular paper from the collection as a query and its citations as the relevant documents. In order to have the best possible generalization to full text collections, we chose the 1000 documents that had the highest percentage of citations with full text as well.

We evaluate a text similarity baseline, which returns the top 100 most similar documents to the query using the multinomial diffusion kernel mentioned earlier. Since other models may return more than 100 documents, we also perform a truncated evaluation for each model, where only the top 100 documents are considered. The numbers in the truncated column allow a fair comparison between the text similarity baseline and the other models.

The results we show here are the result of a 10-fold cross validation experiment. We trained 10 models, each using 900 query documents and a test set of 100 documents. Each document was used as a test document exactly once, and as a training document exactly nine times. The reported results show the mean average precision over all 10 experiments, and also the maximum and minimum value seen.

In order to assess the usefulness of particular features, we performed experiments that removed each feature from the model in isolation. We expect that if a feature is very useful, the retrieval effectiveness of the system will drop dramatically when a feature is removed; if it is not useful, we expect effectiveness to stay the same. Note that we did not re-train the model for these tests; we only set the weight of the removed feature to zero.

		Full			Truncated		
		Mean	Interval		Mean	Interval	
Baseline	Text Similarity	0.0079	0.0055	0.0103	0.0079	0.0055	0.0103
Experimental	All Features	0.1016	0.0781	0.1251	0.0940	0.0727	0.1153
	No Text	0.0675	0.0539	0.0811	0.0612	0.0469	0.0754
	No Author	0.0983	0.0747	0.1219	0.0917	0.0701	0.1132
	No Katz	0.0335	0.0256	0.0414	0.0257	0.0194	0.0320
	No Cite Count	0.1005	0.0771	0.1238	0.0931	0.0718	0.1144
	No Date	0.1052	0.0834	0.1269	0.0979	0.0784	0.1174
	No Title	0.1016	0.0781	0.1251	0.0940	0.0727	0.1153

Table 3. Results of 10-fold cross validation experiments on a 1000 query set. Results are reported using the mean average precision metric. Full results represent mean average precision over the entire retrieved set, while the truncated results reflect mean average precision computed over the first hundred retrieved documents. Confidence intervals are based on the t distribution over all 10-folds. All experimental models significantly outperform text similarity (Wilcoxon, $p = 0.01$). All experimental models with the Katz measure significantly outperform the "No Katz" method (Wilcoxon, $p = 0.01$)

6.1 Results

The results of our experiments are shown in Table 3.

Our experimental results show the effectiveness of our system in various modes against a text similarity baseline. The confidence intervals come from the t distribution, which makes a mild normal assumption about our data which may not be true. However, we also performed the distribution-free Wilcoxon signed rank test ($p < 0.01$), which makes no such assumptions. From this, we find that all experimental models significantly outperform the text similarity baseline. Also, we find that the "No Katz" experimental model is significantly outperformed by all other experimental models ($p < 0.01$). The truncated "No Text" is significantly outperformed by all models with both the Katz feature and Text ($p < 0.05$), although we can conclude nothing about the "No Text" non-truncated model.

Perhaps surprisingly, text similarity on its own appears to be a poor way to succeed in this evaluation. The citation metrics play a major role in quality document ranking.

A second surprise is how little many of the features we used matter in the final ranking of documents. The author, citation count, publication date and title text features add little to nothing to the effectiveness of the system. This is not to say that these features are not correlated with citation, but they appear to be dominated by the full text and Katz features. Notice that the "No Title" line of the table is identical to the "All Features" line; this is because the training process assigned the title a weight of zero in every model we trained.

The Katz measure shows itself to be crucial to the performance of our model. Without the Katz feature, model performance drops by over half. One way to interpret this result is that the Katz measure, among all features we use, is the one that is closest in capturing what actual scientists actually cite.

The "No Text" model relies entirely on author information to find related documents. This means that the authors of the query are used to find other documents written by those authors, and then those documents are ranked. No documents not written by the query's authors are considered. The performance here suggests that authors often cite papers twice in successive publications. If the author feature is removed, the effectiveness of this approach drops to zero (no documents are retrieved).

7 Conclusion

We find that ranking academic documents is a difficult problem. We find that using text similarity alone as a retrieval feature is not enough for high quality retrieval for this task, and that many features that might seem to be useful are not helpful in increasing retrieval performance. However, we find that using citation information is critical for ranking documents in a way that reflects how scientists actually cite.

Further progress in this area will require new evaluation strategies. In this paper, we have considered only binary relevance, as is used in traditional TREC tasks. In this task, we must contend with potentially hundreds of papers that could be relevant, while most conference papers cite only twenty. Assessing different levels of relevance may be necessary to address this distinction.

We hope that this work will spark new interest in the academic literature search problem. Unlike other tasks, the academic literature search task offers a real world task where we might expect a user to input an extremely long query, wait an hour for a result, or wade through hundreds of results in search of the perfect document. In addition, these documents contain rich citation information that can be leveraged to find the structure in the corpus. All of these attributes let us consider approaches to the academic literature retrieval task that would not be considered practical for a traditional ad hoc task.

8 Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval, in part by NSF grant #CNS-0454018 , and in part by The Central Intelligence Agency, the National Security Agency and National Science Foundation under NSF grant #IIS-0326249. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor.

References

1. F. Diaz. Regularizing ad hoc retrieval scores. In *CIKM '05*. ACM Press, 2005.
2. P. Druschel, V. Pai, and W. Zwaenepoel. Extensible kernels are leading OS research astray. In *WWOS-VI*, pages 38–42, 1997.
3. E. Garfield. Citation analysis as a tool in journal evaluation. *Science*, 178:471–479, 1972.
4. C. Giles, K. Bollacker, and S. Lawrence. Citeseer: An automatic citation indexing system. In *Digital Libraries '98*, pages 89–98, New York, NY, USA, 1998. ACM Press.
5. Google. Google scholar. `http://scholar.google.com`.
6. S. Hill and F. Provost. The myth of the double-blind review?: Author identification using only citations. *SIGKDD Explorations Newsletter*, 5(2):179–184, 2003.
7. T. Joachims. A support vector method for multivariate performance measures. In *ICML 2005*, 2005.
8. J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.
9. R. Krovetz. Viewing morphology as an inference process. In *SIGIR '93*, pages 191–202. ACM Press, 1993.
10. J. Lafferty and G. Lebanon. Diffusion kernels on statistical manifolds. *JMLR*, 6:129–163, 2005.
11. M. Ley. Dblp: Digital bibliography and library project. `http://dblp.uni-trier.de/`.
12. D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM 2003: Proceedings of the 12th International Conference on Information and Knowledge Management*, pages 556–559. ACM Press, 2003.
13. A. McCallum, A. Saunders, A. Culotta, G. Huang, C. Sutton, and P. Kanani. Rexa. `http://www.rexa.info`.
14. D. Metzler and W. B. Croft. A Markov Random Field model for term dependencies. In *SIGIR '05*, pages 472–479, 2005.
15. L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.
16. G. Salton and C. Buckley. On the use of spreading activation methods in automatic information. In *SIGIR '88*, pages 147–160, New York, NY, USA, 1988. ACM Press.
17. T. Strohman, D. Metzler, H. Turtle, and W. B. Croft. Indri: A language model-based search engine for complex queries. In *Proceedings of the International Conference on Intelligence Analysis*, 2005.
18. B. Wellner, A. McCallum, F. Peng, and M. Hay. An integrated, conditional model of information extraction and coreference with application to citation matching. In *AUAI '04*, pages 593–601, Arlington, Virginia, United States, 2004. AUAI Press.
19. H. D. White and K. W. McCain. Bibliometrics. *Annual Review of Information Science and Technology*, 24:119–165, 1989.
20. P. Wouters. *The Citation Culture*. PhD thesis, University of Amsterdam, March 1999.