IEEE Computer, Volume 32, Number 6, pp. 67-71, 1999. Copyright © IEEE[1]

Research Feature

Digital Libraries and Autonomous Citation Indexing

Steve Lawrence
C. Lee Giles
Kurt Bollacker
NEC Research Institute

The Web is revolutionizing the way researchers access scientific literature, however scientific literature on the Web is largely disorganized. Autonomous citation indexing can help organize the literature by automating the construction of citation indices. Autonomous citation indexing aims to improve the dissemination and retrieval of scientific literature, and provides improvements in cost, availability, comprehensiveness, efficiency, and timeliness.

The rapid increase in the volume of scientific literature has led to researchers constantly fighting information overload in their pursuit of knowledge. Staying up to date with recently published literature— and actually finding relevant sources— is becoming increasingly difficult, if not impossible. Experience varies widely, but the time when every essential journal was held in all major academic libraries has passed.[1]

The Web promises to make more scientific articles more easily available. An increasing number of authors, journals, institutions, and archives make research articles available for almost immediate access. However, scientific literature on the Web remains remarkably disorganized. Scientists can post relevant preprints on their Web sites, but finding articles quickly can be difficult because Web search engines have difficulty keeping up to date[2] and currently do not index the contents of postscript and PDF (portable document format) files.

INDEXING INFORMATION

A citation index[3] catalogues the citations that an article makes, linking the articles with the cited works. Citation indices were originally designed mainly for information retrieval and to allow navigating the literature in unique ways, such as backward in time (through the list of cited articles) or forward in time (to find more recent, related articles).

Citation indexing can improve scientific communication by

- revealing relationships between articles,
- drawing attention to important corrections or retractions of published work,
- identifying significant improvements or criticisms of earlier work, and
- helping limit the wasteful duplication of prior research.

Citation indices can also be used to analyze research trends, identify emerging areas of science, and find out where and how often a particular article is cited.

Currently available and proposed citation indices of scientific literature, however, depend heavily on human preparation or editing of information. For example, Robert D. Cameron proposed a universal bibliographic and citation database that would link every scholarly work ever written.[4] He described a system in which all published research would be available to and searchable by any scholar with Internet access. The database would include citation links and would be comprehensive and current. Cameron's proposed system would transfer the manual effort associated with citation indexing to the authors or institutions, who would be required to provide citation information in a specific format.

Such workload requirements are probably a major factor preventing the realization of Cameron's proposal. Autonomous citation indexing (ACI), on the other hand, sidesteps these requirements by completely automating the citation indexing process without requiring any extra effort from authors or institutions. Additionally, ACI improves on other technologies by extracting and making the context of citations easy to access.

AUTONOMOUS CITATION INDEXING

An ACI system can automatically create a citation index from literature in electronic format. Such a system can autonomously locate articles, extract citations, identify citations to the same article that occur in different formats, and identify the context of citations in the body of articles. The viability of ACI depends on the ability to perform these functions accurately. We built a prototype digital library called CiteSeer that successfully performs these tasks with sufficient accuracy.[5]

Operating completely autonomously, CiteSeer works by downloading papers from the Web and converting them to text. It then parses the papers to extract the citations and the context in which the citations are made in the body of the paper, storing this information in a database. CiteSeer includes full-text article and citation indexing, and allows the location of papers by keyword search or citation links. It can also locate papers related to a given article by using common citation information or word similarity. Given a particular paper, CiteSeer can also display the context of how subsequent publications cite that paper.

Locating documents

An ACI system can find articles by searching the Web, monitoring mailing lists or newsgroups, or by linking directly to publishers. Once familiar with ACI systems, researchers will be able to notify the systems of new papers directly, allowing these papers to be indexed almost immediately. Journals typically charge for access to online papers, so one way to index these papers would be to make agreements with the publishers themselves. An ACI system is likely to benefit publishers by directing users to the journal's Web site.

Currently, CiteSeer uses Web search engines (like AltaVista, HotBot, and Excite) and heuristics to locate good starting points for crawling the Web. For example, CiteSeer can search for pages that contain the words "publications," "papers," and "postscript."

CiteSeer downloads Postscript or PDF files, which are then converted into text using PreScript from the New Zealand Digital Library project (http://www.nzdl.org/). CiteSeer checks to verify that the document is a research document by testing for the existence of a reference or bibliography section. In addition, CiteSeer detects and reorders postscript files that print pages in the reverse order.

Processing and parsing documents

Once CiteSeer has a document in usable form, it must locate the section containing the reference list, either by identifying the section header or the citation list itself. It then extracts individual citations, delineating individual citations by citation identifiers, vertical spacing, or indentation.

CiteSeer parses each citation using heuristics to extract fields such as title, author, year of publication, page numbers, and the citation identifier. CiteSeer uses citation identifiers like "[6]," "[Giles97]," or "Marr 1982" to locate the citations in the document body, after which CiteSeer can extract the context of the citations. By using regular expressions, CiteSeer can handle variations in the citation identifier, such as when a citation lists all authors or only the first author.

We constructed the heuristics used to parse the citations using an *invariants first* method. This means that subfields of a citation that have relatively uniform syntax, position, and composition given all previous parsing, are parsed next. For example, citation identifiers always appear at the beginning of citations, and retain the same format across all of an article's citations. Once CiteSeer identifies a citation's more regular features, it uses trends in syntactic relationships between subfields to predict where a desired subfield exists, if at all. For example, author information almost always precedes title information. CiteSeer also uses databases of author names, journal names, and so forth to help identify citation subfields.

Citations to a given article may have widely varying formats. For example, Figure 1 shows a sample extracted from machine learning publications on the Web. Much of the significance of ACI and CiteSeer derives from the ability to recognize that all of these citations refer to the same article. With this capability, such systems can generate lists of citations across multiple articles andstatistics on citation frequency.

Aha, D. W. (1991), Instance-based learning algorithms, Machine Learning 6(1), 37-66.

D. W. Aha, D. Kibler and M. K. Albert, Instance-Based Learning Algorithms. Machine Learning 6 37-66, Kluwer Academic Publishers, 1991.

Aha, D. W., Kibler, D. & Albert, M. K. (1990). Instance-based learning algorithms. Draft submission to Machine Learning.

Figure 1. Sample of citations to the same paper showing typical variations in format.

As suggested by the citations in Figure 1, the problem is not completely trivial. All fields, including the title, author names, and even the year of publication routinely contain errors. Autonomously determining the subfields of a citation is not always easy. For example, commas are often used to separate fields, but they are also used to separate lists of authors and are frequently embedded in titles. Periods are used to separate fields but are also used to denote abbreviations. Sometimes there is no punctuation at all between fields.

Methods. We have considered four broad classes of methods for identifying and grouping citations to identical articles:

- String distance or edit distance measurements, which consider distance as the difference between strings of symbols. The Levenshtein distance is a well-known edit distance where the difference between two strings is simply the number of insertions, deletions, or substitutions required to transform one string into another. A more recent and sophisticated example is LikeIt, an intelligent string comparison algorithm introduced by Peter Yianilos.[6]
- Word frequency or word occurrence measurements, which are based on the statistics of words that are common to each string. Word frequency measurements such as term frequency × inverse document frequency (TFIDF) are common in information retrieval.
- Knowledge about subfields or the structure of the data can also be used. In citations, subfields such as author name, title, year of publication, and so forth can be used.
- Probabilistic models, which use known bibliographic information to identify subfields from the words contained in and/or the structure of citations. These subfields could be used with any of the previous methods.

We investigated algorithms from each of these classes and performed quantitative tests. We extracted several sets of citations from online papers, manually grouped identical citations, tuned the algorithms on a training set, and compared the correct groupings with the automated groupings.

CiteSeer currently uses an algorithm based on normalization of the citations, sorting according to length and matching words and phrases within subfields. On tests covering 1,158 citations, about five percent of the automated groupings this algorithm produced contained an error. This does not mean that CiteSeer incorrectly grouped five percent of citations; just one incorrect citation in a group marks the entire group as incorrect.

Improving the algorithm. While CiteSeer's current algorithm is sufficient for practical use, it could be improved in many ways. For example, the use of machine learning techniques and probabilistic estimation based on training sets of known bibliographic data may boost performance. Large quantities of bibliographic information are freely available on the Web (like the collection of bibliographies at http://liinwww.ira.uka.de/bibliography/index.html). This information provides labeled training data that learning techniques can use to associate the words contained in and/or the structure of citations with the corresponding subfields.

We initially chose not to use models trained on specific words because the sole use of such models would bias the errors made by the system,and because performance depends critically on the coverage and recency of available training data. For example, errors are more likely to occur for new authors, journals, and areas not included in the training data. Preliminary investigations suggest that probabilistic information from specific words and learning techniques can provide very good performance, and future research could consider adding these techniques to the methods outlined earlier. Another method for improving citation-matching performance would be to allow certain users to correct errors.

An ACI system should also identify the bibliographic details of the indexed papers. CiteSeer uses font and spacing information to identify the title and author of documents being indexed. Identifying the indexed documents allows analyzing the graph formed by citation links, for example CiteSeer computes *hubs* (articles that cite many highly cited articles) and *authorities* (highly cited articles). Ranking by hubs is useful to identify survey, tutorial, or review style articles.

Querying and browsing

CiteSeer's keyword search can return a list of citations matching the query or a list of indexed articles. The articles can then be browsed by following the links between the articles made by citations. Figure 2 shows a sample response for the query "Quinlan" in a CiteSeer library of computer science literature.

CiteSeer's window displays the number of citations to each article in the left-hand column. The "hosts" column indicates the number of unique hosts (Web servers) from which the articles containing the citations originated. The "self" column indicates the citations to the given paper that CiteSeer predicts are self-citations. At the end of the response is a graph showing the number of citations versus the year of publication for each cited article. CiteSeer does not include the number of self-citations in the main number of citations or the graph.

CiteSeer indexes the full text of citations and articles, providing full Boolean search with phrase and proximity support (proximity support allows searching for words separated by a specified maximum distance). When searching for citations, the default mode of operation is to retrieve all citations matching the given query, group the citations to identical papers, and order the results by the number of citations to each paper. CiteSeer does not currently perform any special processing to account for different ways of referencing proper names. However, the Boolean and proximity support can be used to cover variant forms of author names. If an author's last name is unique within a given database, it is sufficient to search just for the last name.

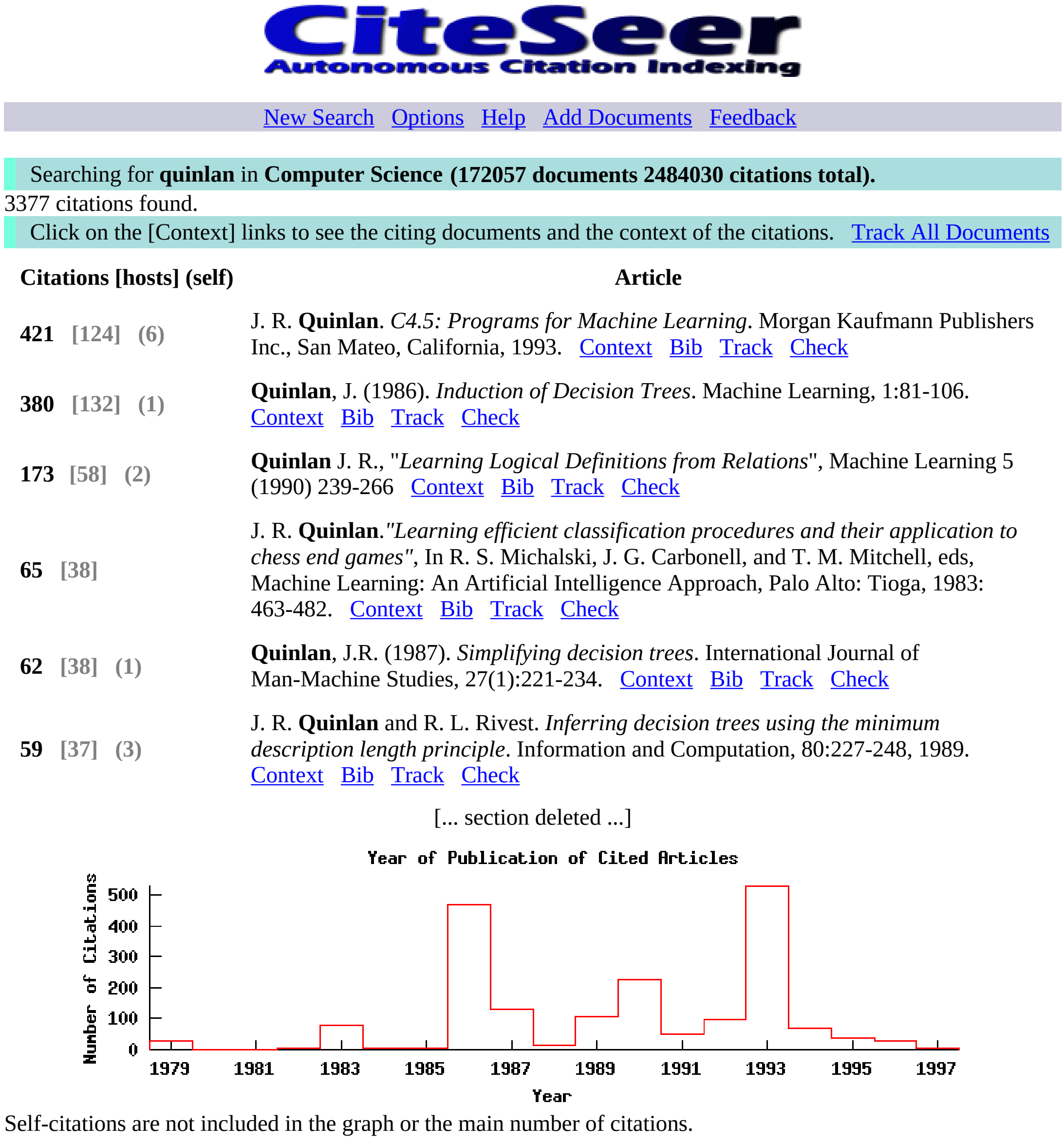
CiteSeer also does not use any "stop" words (such as common words like "the," which indexing typically excludes), so it is possible to search for phrases containing initials. When searching the full text of indexed articles, CiteSeer returns the header for matching documents along with the context of the articles where the keywords occur. Users can order documents according to the number of citations to them, their citations of important articles, or by date. CiteSeer can display details of particular documents, including the abstract, full text, list of citations, and an active bibliography of related documents.

After making an initial keyword search, the user can browse the digital library using citation links. CiteSeer shows which papers are cited by a particular publication and which papers cite a particular publication, including the context of those citations. Figure 3 lists the papers that cite an article in Figure 2, along with the context of the citations (obtained by clicking on the appropriate context link shown in Figure 2). The context may contain a brief summary of the paper, another author's response to the article, limitations or criticism of the original work, or subsequent work that builds upon the original article. The context of citations can help a researcher determine whether to read the citing or cited articles.

CiteSeer can also find related articles by using several algorithms:

- word vectors, a TFIDF scheme used to locate articles with similar words;
- distance comparison of the article headers, used to find similar headers; and
- Common Citation × Inverse Document Frequency (CCIDF), which finds articles with similar citations.

CCIDF is analogous to the word-oriented TFIDF because it considers the common citations between any pair of documents weighted by the inverse frequency of citation. The weighting downplays the importance of common citations to highly cited methodological papers, for example.


CiteSeer
Autonomous Citation Indexing
New Search Options Help Add Documents Feedback
Year of Publication of Cited Articles
Number of Citations
0
100
200
300
400
500
1979
1981
1983
1985
1987
1989
1991
1993
1995
1997
Year

Searching for **quinlan** in **Computer Science (172057 documents 2484030 citations total).**

3377 citations found.

Click on the [Context] links to see the citing documents and the context of the citations. Track All Documents

Citations [hosts] (self)	Article
421 [124] (6)	J. R. **Quinlan**. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Mateo, California, 1993. Context Bib Track Check
380 [132] (1)	**Quinlan**, J. (1986). *Induction of Decision Trees*. Machine Learning, 1:81-106. Context Bib Track Check
173 [58] (2)	**Quinlan** J. R., "*Learning Logical Definitions from Relations*", Machine Learning 5 (1990) 239-266 Context Bib Track Check
65 [38]	J. R. **Quinlan**.*"Learning efficient classification procedures and their application to chess end games"*, In R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, eds, Machine Learning: An Artificial Intelligence Approach, Palo Alto: Tioga, 1983: 463-482. Context Bib Track Check
62 [38] (1)	**Quinlan**, J.R. (1987). *Simplifying decision trees*. International Journal of Man-Machine Studies, 27(1):221-234. Context Bib Track Check
59 [37] (3)	J. R. **Quinlan** and R. L. Rivest. *Inferring decision trees using the minimum description length principle*. Information and Computation, 80:227-248, 1989. Context Bib Track Check

[... section deleted ...]

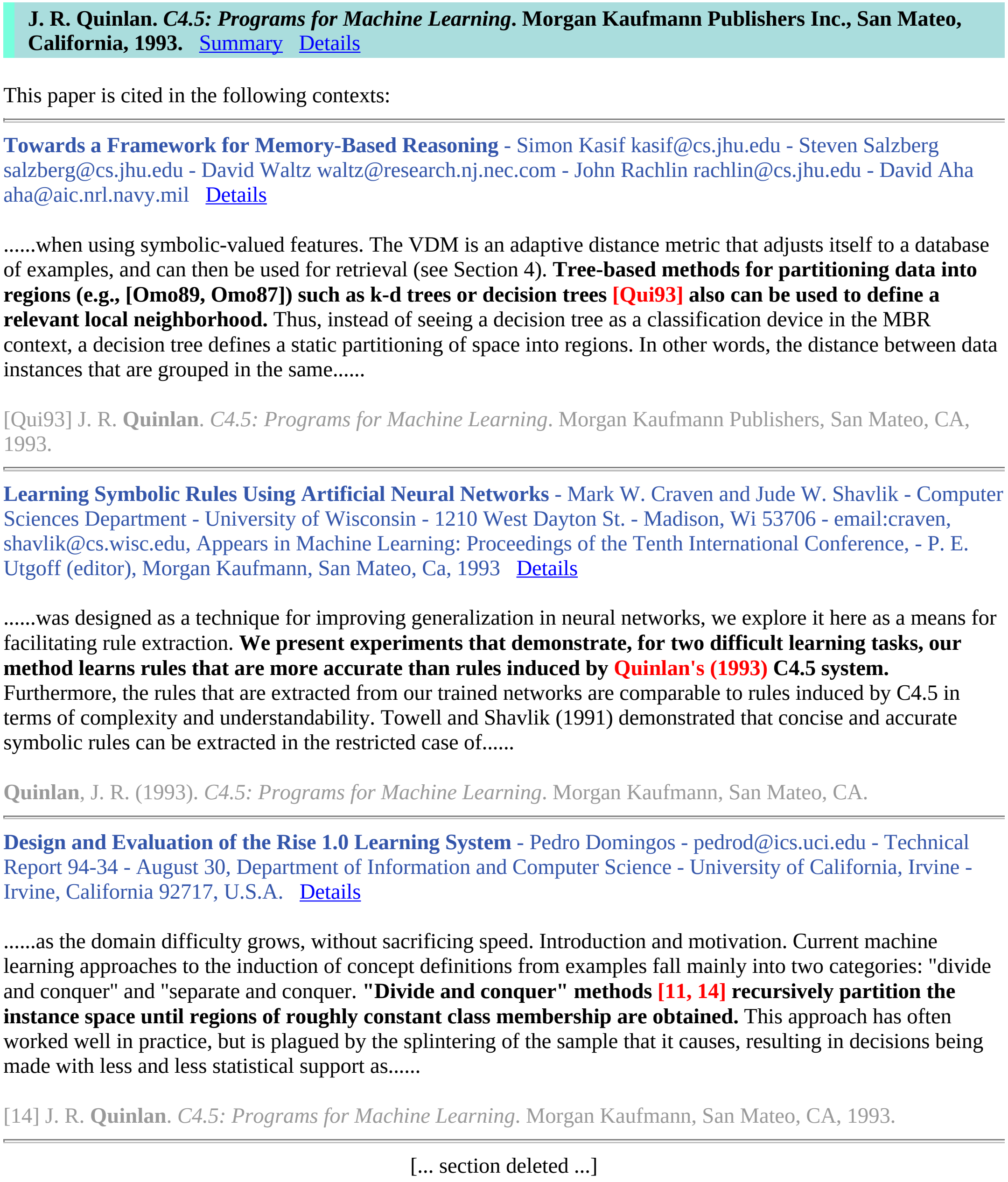
Self-citations are not included in the graph or the main number of citations.

Figure 2. CiteSeer returns this information from a keyword search for the author "Quinlan" in a small test digital library of computer science literature.

DISCUSSION

While CiteSeer is already in use, there are many ways to improve the dissemination and access of scientific information on the Web. For example, printed literature may be processed with optical character recognition and stored efficiently using technology such as DjVu image compression (http://djvu.research.att.com/).

Digital libraries with ACI can provide many additional services, such as current awareness and community features. For example, papers or research topics may be linked to a discussion area where scientists may post formal or informal comments, reviews, responses, and new results. CiteSeer allows researchers to sign up to receive e-mail notification of new citations to papers of interest, or notification of new documents that match a personal profile.



J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Mateo, California, 1993. Summary Details

This paper is cited in the following contexts:

Towards a Framework for Memory-Based Reasoning - Simon Kasif kasif@cs.jhu.edu - Steven Salzberg salzberg@cs.jhu.edu - David Waltz waltz@research.nj.nec.com - John Rachlin rachlin@cs.jhu.edu - David Aha aha@aic.nrl.navy.mil Details

......when using symbolic-valued features. The VDM is an adaptive distance metric that adjusts itself to a database of examples, and can then be used for retrieval (see Section 4). **Tree-based methods for partitioning data into regions (e.g., [Omo89, Omo87]) such as k-d trees or decision trees [Qui93] also can be used to define a relevant local neighborhood.** Thus, instead of seeing a decision tree as a classification device in the MBR context, a decision tree defines a static partitioning of space into regions. In other words, the distance between data instances that are grouped in the same......

[Qui93] J. R. **Quinlan**. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

Learning Symbolic Rules Using Artificial Neural Networks - Mark W. Craven and Jude W. Shavlik - Computer Sciences Department - University of Wisconsin - 1210 West Dayton St. - Madison, Wi 53706 - email:craven, shavlik@cs.wisc.edu, Appears in Machine Learning: Proceedings of the Tenth International Conference, - P. E. Utgoff (editor), Morgan Kaufmann, San Mateo, Ca, 1993 Details

......was designed as a technique for improving generalization in neural networks, we explore it here as a means for facilitating rule extraction. **We present experiments that demonstrate, for two difficult learning tasks, our method learns rules that are more accurate than rules induced by Quinlan's (1993) C4.5 system.** Furthermore, the rules that are extracted from our trained networks are comparable to rules induced by C4.5 in terms of complexity and understandability. Towell and Shavlik (1991) demonstrated that concise and accurate symbolic rules can be extracted in the restricted case of......

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA.

Design and Evaluation of the Rise 1.0 Learning System - Pedro Domingos - pedrod@ics.uci.edu - Technical Report 94-34 - August 30, Department of Information and Computer Science - University of California, Irvine - Irvine, California 92717, U.S.A. Details

......as the domain difficulty grows, without sacrificing speed. Introduction and motivation. Current machine learning approaches to the induction of concept definitions from examples fall mainly into two categories: "divide and conquer" and "separate and conquer. **"Divide and conquer" methods [11, 14] recursively partition the instance space until regions of roughly constant class membership are obtained.** This approach has often worked well in practice, but is plagued by the splintering of the sample that it causes, resulting in decisions being made with less and less statistical support as......

[14] J. R. **Quinlan**. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.

[... section deleted ...]

Figure 3. For each article, CiteSeer shows the header, the context of the citation, and the specific form of the citation. CiteSeer automatically highlights the sentence containing the citation. The Details link allows users to view the full details of the articles (header, abstract, citations, source location, related documents, and so forth). The Summary link shows a summary of citing documents without citation context.

CiteSeer should complement commercial citation indices such as The Institute for Scientific Information's Science Citation Index (SCI). Although CiteSeer is sufficiently accurate to be very useful, SCI can provide greater accuracy, especially in areas where it indexes informal citations (like a reference to a work of art within the body of an article).

But citation indices like SCI are limited because they require manual effort. This limitation means that the database publishers must be selective in the literature that they index, because it is not practical for them to index all literature. SCI indexes predominantly journal articles. Such selective indexing is justified by the fact that a relatively small number of journals account for the bulk of significant scientific results.[7] However, this situation may at least partially arise from information overload: Researchers may only read a small set of journals and miss significant results published elsewhere. Widespread use of digital libraries with ACI should promote the visibility and dissemination of more literature.

There are definite disadvantages to limited journal selection. Journal selection typically follows a review process, which implies that articles making the journal worthy of indexing have already been published. Limiting indexing to journals excludes the information from conferences, monographs, technical reports, and preprints. In areas such as computer science, significant research is often presented at conferences.

The broader coverage that ACI provides can clearly be helpful for literature search, allowing a scientists to find work that cites their own work or is relevant to their research. For work that reaches journal publication, broader coverage of preprints, technical reports, and conference proceedings can provide more timely access. Even work that does not reach journal publication may contain important and/or useful feedback or connections.

Citation statistics are widely used for evaluation. However, evaluation based on citation statistics can lead to erroneous conclusions. The underlying assumption that a large number of citations imply scholarly impact is not always true.[8] What is actually written about a cited document can be very important, but is typically not considered when evaluating citation statistics. Statistics on recent work may not even be available because of the delay imposed by the journal review and publishing process. By making the context of citations easily and quickly browsable, and by indexing technical reports, conference papers, and other literature often available earlier than journal articles, ACI can help to evaluate the importance of individual contributions more accurately and quickly.

The revolution that the Web has brought to information dissemination is not so much due to the availability of information—huge amounts of information has long been available in libraries and elsewhere—but rather the improved efficiency of accessing information. Digital libraries incorporating ACI can help organize scientific literature and may significantly improve the efficiency of dissemination and feedback. ACI may also help speed the transition to scholarly electronic publishing. A widely available linked network of scientific literature could encourage scientists to pursue publication avenues that make their work available online as quickly as possible. ❖

Obtaining CiteSeer

NEC Research has made the CiteSeer software available at no cost for non-commercial use. A demonstration version of CiteSeer indexing over 200,000 computer science articles and over 2 million citations can be found at http://csindex.com/. For current information, contact citeseer@research.nj.nec.com or visit http://www.neci.nec.com/~lawrence/citeseer.html. To subscribe to the CiteSeer listserve, send a message to majordomo@research.nj.nec.com with the text "subscribe citeseer-announce" in the body of the message.

Acknowledgments

We thank Haym Hirsh, Bob Krovetz, Michael Lesk, Michael Nelson, Craig Nevill-Manning, Harold Stone, David Waltz, and Peter Yianilos for useful comments and suggestions.

References

1. S. Hitchcock et al., "Citation Linking: Improving Access to Online Journals," *Proc. 2nd ACM Int'l Conf. on Digital Libraries*, ACM Press, New York, 1997, pp. 115-122.
2. S. Lawrence and C.L. Giles, "Searching the World Wide Web," *Science*, Volume 280, Number 5360, April 3 1998, pp. 98-100.
3. E. Garfield, *Citation Indexing: Its Theory and Application in Science, Technology, and Humanities*, John Wiley & Sons, New York, 1979.
4. R.D. Cameron, "A Universal Citation Database As a Catalyst for Reform in Scholarly Communication," *First Monday*, Apr. 1997, http://www.firstmonday.dk/issues/issue2_4/cameron/index.html.
5. C.L. Giles, K. Bollacker, and S. Lawrence, "CiteSeer: An Automatic Citation Indexing System," *Digital Libraries 98: Third ACM Conf. on Digital Libraries*, ACM Press, New York, 1998, pp. 89-98.
6. P. Yianilos, *The LikeIt Intelligent String Comparison Facility*, Tech. Report 97-093, NEC Research Institute, 1997, http://www.neci.nj.nec.com/homepages/pny/pap

ers/likeit/main.html.

7. J. Testa, "The ISI Database: The Journal Selection Process," http://www.isinet.com/whatshot/essays/-199701.html.
8. T.A. Brooks, "Evidence of Complex Citer Motivations," *J. Am. Soc. Information Science*, Volume 37, January 1986, pp. 34-36.

__Steve Lawrence__ is a research scientist at NEC Research Institute. His research interests include machine learning, artificial intelligence, neural networks, and information retrieval, dissemination, and access. He received a PhD in computer science from the University of Queensland, Australia.

__C. Lee Giles__ is a senior research scientist in computer science at NEC Research Institute. He is also affiliate faculty at the Institute for Advanced Computer Studies at the University of Maryland. His research interests include Web computing, agent/artificial-intelligence technology and neural and machine learning. He received a PhD in optical sciences from the University of Arizona.

__Kurt Bollacker__ is a scientist at NEC Research Institute. His research interests include machine learning, personal assistant agents and autonomous database creation. He received a PhD in computer engineering from the University of Texas.

Contact the authors at NEC Research Institute, 4 Independence Way, Princeton, NJ, 08540; {lawrence,giles,kurt}@research.nj.nec.com.