Information Extraction for Scholarly Document Big Data

Jian Wu, C. Lee Giles
[†]Information Sciences and Technology, Pennsylvania State University, University Park, PA, 16802 USA

ABSTRACT
CiteSeerX is a digital library search engine that provides free access to over six million scholarly documents crawled from the public web. Their metadata is automatically extracted and tagged. We present key extraction technologies used in CiteSeerX, including document classification and de-duplication, document clustering, header/citation extraction, author disambiguation, and table/algorithm extraction. We also describe developing challenges and future work.

Categories and Subject Descriptors
H.4 [**Information Systems Applications**]: Miscellaneous; I.2.1 [**Artificial Intelligence**]: Applications and Expert Systems

1. INTRODUCTION
CiteSeerX's predecessor, CiteSeer, was developed at the NEC Research Institute, Princeton, NJ in 1997 and was considered by many to be the first scholarly digital library that provided autonomous citation indexing [15]. At Penn State since 2003, CiteSeer was renamed as CiteSeerX in 2008 with a new architecture and features and continued to be heavily used. A future goal is to utilize the metadata for various types of semantic search.

CiteSeerX is in many ways unique compared with other scholarly digital libraries and search engines since all documents are harvested from the public Web. Because of this, users have full-text access to all papers searchable in CiteSeerX. Also, CiteSeerX performs automatic extraction and indexing on paper entities such as tables and figures, which is rarely seen in other scholarly search engines. The metadata and a text extraction service [39] are made available for research. A focused web crawler actively harvests publicly available PDFs from the Web, which are then filtered for only scholarly documents. Metadata such as headers and citations are extracted and then ingested into the production databases. The production system is currently hosted in a private cloud [43]. We now discuss the key technologies used for CiteSeerX information extraction [44].

2. EXTRACTION TECHNOLOGIES

Document Classification. Text content extracted from the PDF of crawled documents is filtered to determine whether the document is scholarly or not. A more sophisticated machine learning approach [3] utilizes structural features to classify documents, including *File specific features*, *Text specific features*, *Section specific features*, and *Containment features*. These new classifiers significantly outperform our previous baselines in terms of precision, recall, and accuracy by at least 10%.

Header Extraction. Header extraction is performed using a support vector machine parser, SVMHeaderParse [17] based on *svm-light* [19] . It classifies textual contents into multiple classes, each of which corresponds to a header metadata field, e.g., title, authors. The entire process contains three phases: feature extraction, line classification, and metadata extraction. The overall accuracy of this extractor is 92.9%, which is better than the accuracy (90%) reported by [32]. Recent evidence implies that GROBID [28] would be a good replacement [25].

Citation Extraction. CiteSeerX uses ParsCit [13] for citation extraction, which is a conditional random field (CRF; Lafferty et al. 2001) model that labels the token sequences in reference strings. ParsCit first attempts to find the reference section before parsing reference strings and then searches for where the individual reference starts and ends using either reference markers or heuristic methods. It also extracts citation context. Evaluations show that the performance of ParsCit is comparable to the original CRF based system in Peng & McCallum (2004), and outperforms FLUX-CiM [12].

De-duplication. Near-duplicates (NDs) refer to documents with similar content but minor differences. NDs are very common in crawl-based digital libraries. In CiteSeerX, NDs are detected using a key-mapping algorithm, applied after the metadata extraction module but before papers are ingested. When a document is imported, a set of keys are generated by concatenating *normalized* author last names and *normalized* titles. The key-mapping algorithm [38] is comparable to the state-of-the-art *simhash* approach [4].

Author Disambiguation. CiteSeerX provides a special author search interface. Author search is also the foundation of several other services, such as collaborator search [5] and expert search [6]. Processing a name-based query can be complex since different authors may share the same name and the same author may have several name variations. To disambiguate authors, we block names into small blocks with the assumption that an author can only have different name variations within the same block. CiteSeerX groups two names if they share the same last name and the first initials. In many cases, other information related to authors is used including their collaborators and topics of their published papers. Our algorithm applies DBSCAN (Density-Based Spacial Clustering of Application with Noise) to resolve most of inconsistent classification results violating a transitivity principle [18]. The Random Forest training of the distance function [36] scales well and has decent performance [22].

Table Extraction. CiteSeerX uses the table metadata extractor developed by Liu et al. (2007), which is comprised of three major parts: a text information stripper, a table box detector, and a table metadata extractor. The text information stripper extracts out the textual information from the original PDF files *word by word* by analyzing the output of a general text extractor. These words are then reconstructed with their position information and written into a *document content file*, which specifies the position, line width and fonts of each line. Based on the *document content file*, the tables are identified using a *box-cutting* method, which attempts to divide all literal components in a page into "boxes". Finally, the algorithm finds tables and their metadata in these boxes [27].

Algorithm Extraction. We developed three methods for detecting pseudo-codes in scholarly documents based on textual content extracted from PDF documents. The rule-based method detects the presence of pseudo-code captions using a set of regular expressions. This method yields high detection precision (87%), but low recall (45%), because a large proportion of pseudo-codes (roughly 26%) do not have associated captions. A machine-learning method directly detects the presence of pseudo-code content assuming that pseudo-codes are written in a sparse, programming-like manner, which can be visually spotted as sparse regions in documents and can capture most non-captioned pseudo-codes. The ones that cannot be captured are either written in a descriptive manner or are presented as figures. A hybrid method combines both and achieves a precision of 87% and a recall of 67% [37].

3. USAGE AND PAYOFF

Since 2008, the document collection of CiteSeerX has been steadily growing, now at six million. Currently, CiteSeerX servers are hit more than 2 million times a day and 3–10 PDF files are downloaded per second [34]. Besides the web search, CiteSeerX also provides an OAI protocol for metadata harvesting in order to facilitate content dissemination [40]. Dumps of our database are also available on Amazon S3[1]. CiteSeerX data is updated regularly. The crawler downloads 50,000 to 100,000 PDF files per day, and up to 50,000 new papers are ingested every day. The CiteSeerX data is heavily used in research projects, e.g., [14, 29, 31, 2, 1]. CiteSeerX has released the open source digital library search engine framework, *SeerSuite* [35], which can be used for building personalized digital library search engines.

[1]Accessible upon request.

4. NEW EXTRACTION FRAMEWORK

In general, a scholarly document consists of several of these entities, if not all: a header, a text body, a bibliography, figures, tables, math and algorithms (even chemical formulae [33]). Recently, we developed a multi-entity knowledge extraction framework for scholarly documents in PDF format called PDFMEF [41][2]. It is implemented with a framework that encapsulates open-source extraction tools. Currently, it leverages PDFBox and TET for full text extraction, the scholarly document filter introduced in [3] for document classification, GROBID for header extraction, ParsCit for citation extraction, PDFFigures [11] for figure and table extraction, and algorithm extraction algorithm introduced in [37]. While it can be run out-of-box, the extraction tool in each module is customizable. The framework is designed to be scalable and is capable of running in parallel using a multi-processing technique in Python.

5. CHALLENGES AND FUTURE WORK

Two big challenges in CiteSeerX are data acquisition and information quality. Previously, the majority of CiteSeerX papers were from the computer sciences. Recently, a large number of papers have been collected from mathematics, physics, and medical science by incorporating papers from open-access repositories such as PubMed (subset), and crawling URLs released by Microsoft Academic Search. Our experiments indicate that the crawl efficiency increases by at least 20% using a whitelist policy [42]. One extension of the crawl module is to integrate a crawl scheduler that generates whitelists on a daily basis, which used the webpage updating rate as a selection criteria based on estimated crawl history [7, 9]. To increase coverage and freshness, the crawling process should be parallelized [8]. In the near future we hope to harvest the estimated 25 million freely available scholarly documents [21] on the web.

To increase metadata quality, we recently developed PDFMEF, which can be used to rebuild the entire metadata database. We are also using multiple manually created reference data sets, such as from DBLP, some publishers, etc., to sanitize and correct mistakenly extracted metadata. We have also developed a multi-document-type classifier, which classifies crawled documents into finer categories, such as slides, papers, and theses. This classifier will be used to build a large data corpora for information extraction research. We intend to link multiple documents types to scholarly documents and make them accessible from a federated view. New features that could be incorporated into CiteSeerX are algorithm search [37], figure search [10], and acknowledgment search [16, 23].

6. CONCLUSION

CiteSeerX is an open access digital library search engine which has incorporated multiple information extraction technologies and plans to grow much larger and improve its

[2]https://github.com/SeerLabs/new-csx-extractor

metadata quality. In addition, other metadata such as chemical formulae [20] can be extracted, linked, and used for other types of search including semantic search.

7. ACKNOWLEDGMENTS

We gratefully acknowledge partial support from the National Science Foundation.

8. REFERENCES

[1] S. Bhatia, C. Caragea, H.-H. Chen, J. Wu, P. Treeratpituk, Z. Wu, M. Khabsa, P. Mitra, and C. L. Giles. Specialized research datasets in the citeseer^x digital library. *D-Lib Magazine*, 18(7/8), 2012.

[2] C. Caragea, J. Wu, A. Ciobanu, K. Williams, J. Fernandez-Ramirez, H.-H. Chen, Z. Wu, and C. L. Giles. Citeseerx: A scholarly big dataset. ECIR '14, pages 311–322, 2014.

[3] C. Caragea, J. Wu, K. Williams, S. D. Gollapalli, M. Khabsa, and C. L. Giles. Automatic identification of research articles from crawled documents. WSDM 2014 Workshop on Web-scale Classification: Classifying Big Data from the Web, 2014.

[4] M. Charikar. Similarity estimation techniques from rounding algorithms. STOC '02, pages 380–388, 2002.

[5] H.-H. Chen, L. Gou, X. Zhang, and C. L. Giles. CollabSeer: a search engine for collaboration discovery. JCDL '11, pages 231–240, 2011.

[6] H.-H. Chen, P. Treeratpituk, P. Mitra, and C. L. Giles. CSSeer: an expert recommendation system based on CiteSeerX. JCDL '14, pages 381–382, 2013.

[7] J. Cho and H. Garcia-Molina. The evolution of the web and implications for an incremental crawler. In *Proceedings of the 26th International Conference on Very Large Data Bases*, VLDB '00, pages 200–209, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[8] J. Cho and H. Garcia-Molina. Parallel crawlers. In *Proceedings of the 11th International Conference on World Wide Web*, WWW '02, pages 124–135, New York, NY, USA, 2002. ACM.

[9] J. Cho and H. Garcia-Molina. Estimating frequency of change. *ACM Trans. Internet Technol.*, 3(3):256–290, Aug. 2003.

[10] S. R. Choudhury, S. Tuarob, P. Mitra, L. Rokach, A. Kirk, S. Szep, D. Pellegrino, S. Jones, and C. L. Giles. A figure search engine architecture for a chemistry digital library. JCDL '13, pages 369–370, 2013.

[11] C. Clark and S. Divvala. Looking beyond text: Extracting figures, tables, and captions from computer science paper. AAAI 2015 Workshop on Scholarly Big Data, 2015.

[12] E. Cortez, A. S. da Silva, M. A. Gonçalves, F. Mesquita, and E. S. de Moura. Flux-cim: Flexible unsupervised extraction of citation metadata. JCDL '07, pages 215–224, 2007.

[13] I. Councill, C. L. Giles, and M.-Y. Kan. Parscit: an open-source crf reference string parsing package. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may 2008. European Language Resources Association (ELRA).

[14] D. G. Feitelson and U. Yovel. Predictive ranking of computer scientists using Citeseer data. *Journal of Documentation*, 60:44–61, 2004.

[15] C. L. Giles, K. Bollacker, and S. Lawrence. CiteSeer: An automatic citation indexing system. JCDL '98, pages 89–98, 1998.

[16] C. L. Giles and I. Councill. Who gets acknowledged: Measuring scientific contributions through automatic acknowledgement indexing. *PNAS*, 101(51):17599–17604, 2004.

[17] H. Han, C. L. Giles, E. Manavoglu, H. Zha, Z. Zhang, and E. A. Fox. Automatic document metadata extraction using support vector machines. JCDL '03, pages 37–48, 2003.

[18] J. Huang, S. Ertekin, and C. L. Giles. Efficient name disambiguation for large-scale databases. PKDD '06, pages 536–544. 2006.

[19] T. Joachims. Making large-scale svm learning practical. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods*, chapter Making Large-scale SVM Learning Practical, pages 169–184. MIT Press, Cambridge, MA, USA, 1999.

[20] M. Khabsa and C. Giles. Chemical entity extraction using crf and an ensemble of extractors. *Journal of Cheminformatics*, 7(Suppl 1):S12, 2015.

[21] M. Khabsa and C. L. Giles. The number of scholarly documents on the public web. *PLoS ONE*, 9(5):e93949, May 2014.

[22] M. Khabsa, P. Treeratpituk, and C. Giles. Large scale author name disambiguation in digital libraries. In *Big Data (Big Data), 2014 IEEE International Conference on*, pages 41–42, Oct 2014.

[23] M. Khabsa, P. Treeratpituk, and C. L. Giles. AckSeer: a repository and search engine for automatically extracted acknowledgments from digital libraries. JCDL '12, pages 185–194, 2012.

[24] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. ICML '01, pages 282–289, 2001.

[25] M. Lipinski, K. Yao, C. Breitinger, J. Beel, and B. Gipp. Evaluation of header metadata extraction approaches and tools for scientific pdf documents. In *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '13, pages 385–386, New York, NY, USA, 2013. ACM.

[26] Y. Liu, K. Bai, P. Mitra, and C. L. Giles. Tableseer: Automatic table metadata extraction and searching in digital libraries. In *Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '07, pages 91–100, New York, NY, USA, 2007. ACM.

[27] Y. Liu, P. Mitra, C. L. Giles, and K. Bai. Automatic extraction of table metadata from digital documents. In *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '06, pages 339–340, New York, NY, USA, 2006. ACM.

[28] P. Lopez. Grobid: Combining automatic bibliographic data recognition and term extraction for scholarship publications. In *Proceedings of the 13th European Conference on Research and Advanced Technology for Digital Libraries*, ECDL'09, pages 473–474, Berlin, Heidelberg, 2009. Springer-Verlag.

[29] J. Madadhain, D. Fisher, P. Smyth, S. White, and Y. Boey. Analysis and visualization of network data using jung. *Journal of Statistical Software*, 10:1–35, 2005.

[30] F. Peng and A. McCallum. Information extraction from research papers using conditional random fields. *Inf. Process. Manage.*, 42(4):963–979, July 2006.

[31] M. Pham, R. Klamma, and M. Jarke. Development of computer science disciplines: a social network analysis approach. *Social Network Analysis and Mining*, 1(4):321–340, 2011.

[32] K. Seymore, A. Mccallum, and R. Rosenfeld. Learning Hidden Markov Model Structure for Information Extraction. AAAI '99 Workshop on Machine Learning for Information Extraction, 1999.

[33] B. Sun, P. Mitra, C. Lee Giles, and K. T. Mueller. Identifying, indexing, and ranking chemical formulae and chemical names in digital documents. *ACM Trans. Inf. Syst.*, 29(2):12:1–12:38, Apr. 2011.

[34] P. Teregowda, B. Urgaonkar, and C. L. Giles. Cloud computing: A digital libraries perspective. CLOUD '10, pages 115–122, 2010.

[35] P. B. Teregowda, I. G. Councill, R. J. P. Fernández, M. Khabsa, S. Zheng, and C. L. Giles. Seersuite: developing a scalable and reliable application framework for building digital libraries by crawling the web. WebApps'10, pages 14–14, 2010.

[36] P. Treeratpituk and C. L. Giles. Disambiguating authors in academic publications using random forests. JCDL '09, pages 39–48, 2009.

[37] S. Tuarob, S. Bhatia, P. Mitra, and C. L. Giles. Automatic detection of pseudocodes in scholarly documents using machine learning. ICDAR, pages 738–742, 2013.

[38] K. Williams and C. L. Giles. Near duplicate detection in an academic digital library. DocEng '13, pages 91–94, 2013.

[39] K. Williams, L. Li, M. Khabsa, J. Wu, P. Shih, and C. L. Giles. A web service for scholarly big data information extraction. ICWS '14, 2014.

[40] K. Williams, J. Wu, S. R. Choudhury, M. Khabsa, and C. L. Giles. Scholarly Big Data Information Extraction and Integration in the CiteSeerX Digital Library. IIWeb '14, 2014.

[41] J. Wu, J. Killian, H. Yang, K. Williams, S. R. Choudhury, S. Tuarob, C. Caragea, and C. L. Giles. Pdfmef: A multi-entity knowledge extraction framework for scholarly documents and semantic search. In *Proceedings of the 8th International Conference on Knowledge Capture*, K-CAP '15. ACM, 2015 accepted.

[42] J. Wu, P. Teregowda, J. P. F. Ramírez, P. Mitra, S. Zheng, and C. L. Giles. The evolution of a crawling strategy for an academic document search engine: whitelists and blacklists. In *Proceedings of the 3rd Annual ACM Web Science Conference*, WebSci '12, pages 340–343, New York, NY, USA, 2012. ACM.

[43] J. Wu, P. Teregowda, K. Williams, M. Khabsa, D. Jordan, E. Treece, Z. Wu, and C. L. Giles. Migrating a digital library into a private cloud. IC2E '14, 2014.

[44] J. Wu, K. Williams, H.-H. Chen, M. Khabsa, C. Caragea, A. Ororbia, D. Jordan, and C. L. Giles. Citeseerx: Ai in a digital library search engine. In *The Twenty-Sixth Annual Conference on Innovative Applications of Artificial Intelligence*, IAAI '14, 2014.