CiteSeerX: 20 Years of Service to Scholarly Big Data

Jian Wu
Old Dominion University
Norfolk, VA
jwu@cs.odu.edu

Kunho Kim
Pennsylvania State University
University Park, PA
kunho@cse.psu.edu

C. Lee Giles
Pennsylvania State University
University Park, PA
giles@ist.psu.edu

ABSTRACT

We overview CiteSeerX, the pioneer digital library search engine, that has been serving academic communities for more than 20 years (first released in 1998), from three perspectives. The **system** perspective summarizes its architecture evolution in three phases over the past 20 years. The **data** perspective describes how CiteSeerX has created searchable scholarly big datasets and made them freely available for multiple purposes. In order to be scalable and effective, AI technologies are employed in all essential modules. To effectively train these models, a sufficient amount of data has been labeled, which can then be reused for training future models. Finally, we discuss the **future** of CiteSeerX. Our ongoing work is to make CiteSeerX more sustainable. To this end, we are working to ingest all open access scholarly papers, estimated to be 30-40 million. Part of the plan is to discover dataset mentions and metadata in scholarly articles and make them more accessible via search interfaces. Users will have more opportunities to explore and trace datasets that can be reused and discover other datasets for new research projects. We summarize what was learned to make a similar system more sustainable and useful.

CCS CONCEPTS

• **Information systems** → **Digital libraries and archives**; *Information integration*; • **Computing methodologies** → *Information extraction.*

KEYWORDS

CiteSeerX, digital libraries, search engines, scholarly big data

ACM Reference Format:
Jian Wu, Kunho Kim, and C. Lee Giles. 2019. CiteSeerX: 20 Years of Service to Scholarly Big Data. In *Artificial Intelligence for Data Discovery and Reuse 2019 (AIDR '19), May 13–15, 2019, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3359115.3359119

1 INTRODUCTION

The number of scientific publications has been increasing exponentially after the mid 1900's [14]. This poses a great challenge in managing a large number of documents and providing timely access to a growing number of researchers. Mass digitization partially solved the problem by storing document collections in digital repositories. The advent of modern information retrieval methods significantly expedited the process of relevant search. However, documents are still saved individually by many users. In 1997, three computer scientists at the NEC Research Institute (now NEC Labs), New Jersey, United States – Steven Lawrence, Kurt Bollacker, and C. Lee Giles, conceived an idea to create a network of computer science research papers through citations, which was to be implemented by a search engine, the prototype *CiteSeer*. Their intuitive idea, automated citation indexing [8], changed the way researchers searched for papers. Users were readily able to navigate from one paper to another by tracking citation relationships.

CiteSeer first served the academic community in 1998 (mostly computer science). It is usually recognized as the first digital library search engine (DLSE)[1]. In 2008, CiteSeer was renamed CiteSeerX, where "X" stands for a series of enhancements as well as architecture and infrastructure redesigns. As a production system based in an academic setting, CiteSeerX has been steadily growing. A relatively small team overcame many scientific and technical challenges with the goal of making the system more accurate, accessible, and scalable. The current team has designed and implemented algorithms for several outstanding problems such as citation parsing [5], table extraction, e.g., [16], author name disambiguation, e.g., [21], document classification, e.g., [3], and data cleansing, e.g., [18].

Although there are similar DLSEs available nowadays. CiteSeerX maintains a unique position. (1) It uses a focused web crawler to actively crawl the public Web. This is different from ACM DL, and IEEE Xplore, where the metadata is entered by authors and provided by these publishers. Google Scholar obtains its data from both publishers and the Web and redirects users to webpages containing documents that are not necessarily open access (OA). Microsoft Academic's data is released via the Academic Knowledge API behind a pay wall. Unlike Academia.edu, ResearchGate, and AMiner, CiteSeerX does not solicit paper uploads from individual authors. CiteSeerX is an OA digital library and users have access to full-text of all documents searchable on its website. All papers are associated with public URLs. (2) CiteSeerX provides all automatically extracted metadata and citation context via an OAI (Open Archive Initiative) interface. The data can also be downloaded from a publicly available drive under a Creative Commons (CC) license, a service not available from Google Scholar and Semantic Scholar. (3) Nearly all CiteSeerX papers are indexed by Google Scholar. (4) CiteSeerX provides an open source software framework called SeerSuite, which has been deployed at other sites such as the Qatar University Library.

[1]For reference, Google Scholar was launched in 2004; Windows Live Academic Search, later renamed Microsoft Academic Search, now called Microsoft Academic was launched in 2006; Semantic Scholar was launched in 2015.

AIDR '19, May 13–15, 2019, Pittsburgh, PA, USA

ACM ISBN 978-1-4503-7184-1/19/05…$15.00
https://doi.org/10.1145/3359115.3359119

Figure 1: CiteSeer(X) landing pages over selected years.

2 SYSTEM EVOLUTION

The evolution of the CiteSeer System can be divided into three phases: the single machine phase (1997–2003), the multiple server phase (2003–2013), and the private cloud phase (2013–present). The frontpage designs have changed selected years, as seen in Figure 1.

2.1 Phase I: Single Machine (1997–2003)

The original CiteSeer at NEC Research Institute was developed and deployed on a single server. The web service was based on Apache HTTP. Because there was not many open source software packages that fit their goals, the developers wrote almost all software by themselves in Perl for the web crawler, the indexer, and the search API. The search engine also used the name *ResearchIndex* at one point. The crawlers were seeded from manually curated homepage URLs of computer scientists. The search engine indexed about 220,000 documents with 2.5 million citations.

2.2 Phase II: Physical Cluster (2003–2013)

In 2003, an NSF SGER grant[2] allowed CiteSeer to be moved to the College of Information Sciences and Technology (IST) at the Pennsylvania State University (PSU). To overcome the capacity limit of a single machine, the search engine evolved to a multi-server system. Before 2007, there were 8 servers including 2 load balancers, 2 web servers, 3 repository servers, and 1 staging server (for development and web crawling). In 2005, the NSF CRI grant[3] proposed to develop the next generation of CiteSeer and scaled up the system to 14 servers. CiteSeer was renamed to CiteSeerX in 2008. *Lucene* was introduced as the main indexer. In 2011, Apache *Solr* was adopted as the main indexer, with about 2 million academic documents.

The backbone software was rewritten using Java [6], which used a digital library framework SeerSuite [20]. The web application uses a model-view-controller architecture implemented with the Spring framework. The frontend uses a mix of Java server pages and JavaScript to generate user interfaces. The web application is composed of servlets that interact with the index and database for keyword search and uses Data Access Objects to interact with databases and the repository. The entire data is partitioned across 3 major databases: user information, document metadata, and citation graphs. The metadata extraction method was built in Perl, working in batch mode. The ingestion system, which feeds the database and repository was integrated into the web application. The data were acquired using an incremental web crawler developed using Django. The crawler discovered 700k+ parent URLs linking to OA PDFs by 2013.

2.3 Phase III: Private Cloud (2013–present)

A thorough analysis indicated a private cloud was the most economic and efficient way to overcome the bottlenecks of system maintenance and scalability [27]. In 2013, supported by an NSF grant[4] CiteSeerX was successfully migrated into a private cloud. The infrastructure consists of 3 layers. The *storage layer* includes 2 servers for storing virtual machines (VMs); the *processing layer* includes 5 high-end servers running VMs for web service, database, etc.; the software on VMs runs on the *application layer*. At least 20 VMs are created and running in the private cloud. The web crawler, due to its high demand on bandwidth and disk access, is hosted on a physical server. The software was basically inherited from Phase II, but the system was enhanced with features such as author and table searches, built on author name disambiguation, e.g., [21], table extraction, e.g., [16], etc.

2.4 Usage and Community Benefit

By 2017, CiteSeerX had ingested the metadata and full text of more than 10 million OA academic documents on the Web and it is increasing. According to Google Analytics and local access logs, CiteSeerX has almost 3M individual users in Year 2017 and has 500,000 documents downloaded daily with on average 3 million hits per day. The OAI is accessed approximately 5000 times monthly [23]. A Google search of "CiteSeerX OR CiteSeer" returns about 10M results[5]. CiteSeerX has a world-wide user population. The top 5 countries in 2017 were China (33%), United States (27%), India (11%), United Kingdom (7%), and Germany (7%). Access log analysis in 2015 indicated that approximately 5000 and 7000 accesses per day are from Historical Black Colleges and Universities and Hispanic Serving Institutes, respectively.

3 AI AND REUSABLE DATA

3.1 AI in CiteSeerX

CiteSeerX incorporates AI technologies in many mission critical tasks. Figure 2 illustrates the procedures for mining and extracting scholarly big data in CiteSeerX, starting from raw PDFs to various data and AI-based software products. The PDFs are first classified into academic and non-academic documents. Machine learning based classifiers in place of the rule-based classifier boosts the F_1 by at least 10% [3]. For information extraction (IE), SVMHeaderParse is replaced [9] with GROBID [17] because of performance [15]. For non-textual information extraction, we use pdffigures2 [4] to extract figures and tables. Algorithms were also developed to extract algorithms [22] and chemical entities [12]. Math expression extraction is still under active research, e.g., [30]. We perform document deduplication (conflation), keyphrase extraction [2], and author

[2] SGER: A Digital Library Archive for Computer Scientists.
[3] CRI: Collaborative: Next Generation CiteSeer.
[4] Collaborative Research: CI-ADDO-EN: Semantic CiteseerX
[5] The exact number may vary somewhat depending on when to query.

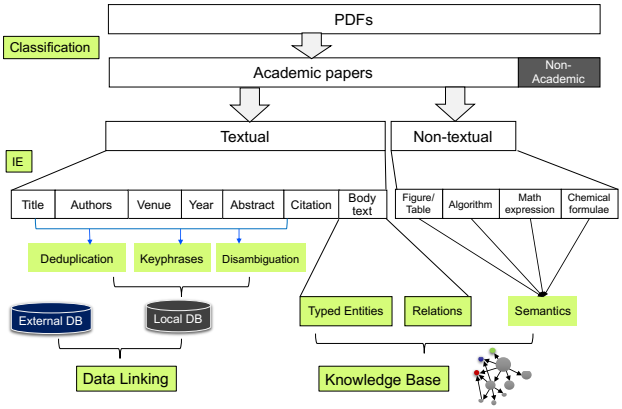

Figure 2: Overview of the mining of scholarly big data in CiteSeerX. Green boxes are tasks that use AI-based software.

name disambiguation [21]. We also link local databases with external databases [26]. Finally, a knowledge base can be constructed using typed entities [24] and relations extracted from body text, plus semantics extracted from non-textual information [1].

It should be noted that there have been a myriad of AI-based algorithms proposed for tasks above, but many cannot be adopted mainly due to 3 reasons: (1) published work had results were not repeatable; (2) the algorithm could not be scaled to big data; (3) the work was based on a toy model with unrealistic assumptions of input data quality. As such we strive to develop portable and scalable AI-based software and adopt new algorithms and implementations that can improve data quality, quantity, and user experience. Algorithms that require significant transfer overhead or not designed for big data are usually inappropriate.

3.2 Reusable Data

CiteSeerX offers two types of reusable data – automatically extracted data (AED) and manually labeled data (MLD). Types of AED and their sizes are tabulated in Table 1. *papers* to be indexed are obtained from web crawling. *Metadata* and *authors* are generated by IE (Figure 2). The *disambiguated authors* are obtained using random forests and DBSCAN clustering [7]. *Citations* and *citation context* are extracted using ParsCit [5]. The *citation graph* is generated heuristically [28]. Data in Table 1 are stored in MySQL and dumped into a `.sql` file. It takes about 550GB after imported into the database. The unique bibliographic records indexed by *Solr* is 360GB. The repository containing all types of files (PDF, XML, TXT, etc.) takes 15TB.

Table 2 presents examples of MLDs. These datasets can be reused for training and evaluating new models. Two examples below demonstrate how these datasets can be used.

For the first example, we attempt to cleanse the metadata produced by IE [18]. Data quality is a ubiquitous problem for automatic extraction pipelines. The errors in metadata can propagate and lead to unreliable results in downstream analysis. One approach is to cleanse the dataset (called target dataset) by matching it against a clean reference dataset, and then use reference data to overwrite target data. To train such a model, we developed the *paper entity matching* dataset (Table 2) containing 688 matching pairs between CiteSeerX and reference databases (DBLP, IEEE, etc.), with an equal amount of negative matching pairs. By matching headers alone, the model achieves $F_1 \sim 92\%$. By matching headers and citations, the model achieves $F_1 > 99\%$. The dataset *CiteSeerX-2018* (Table 1) comes from matching the entire CiteSeerX database with DBLP, and Medline [25].

Another example is disambiguating author mentions in academic papers. Name disambiguation is a common and important issue (unfortunately often ignored) for nearly all problems involving author names. The goal is to build a model that cluster the same surface name corresponding to different individuals (e.g., *Michael Jordan*, a computer scientist or a basketball player?) and different surface names referring to the same individual (e.g., *CL Giles*, *Lee Giles*, and *C. Lee Giles* all are the same person). The *author name disambiguation* dataset was constructed in 2004 [10] and has been reused in several papers for the same task [11, 13, 19, 21, 29].

Table 1: Automatically extracted data 2018.

Data type	Size	Description
papers	10M	Full text with metadata
authors	32M	Author mentions
disambiguated authors	2M	Profile and linked papers
citations	240M	Citation mentions
citation context	203M	Text around in-text citations
citation graph	71M vertices 183M edges	citation relations of unique bibliographic records

4 LESSONS LEARNED

As a system designed to serve the academic and research community, CiteSeerX is one of the few systems that still exists after 20 years. To keep the system up to date, the CiteSeerX team undertakes both scientific research and system design and development. As such there are lessons learned that may benefit related systems of similar size and functionality. (1) Maintenance is extremely important. Several other very good systems did not last long because

Table 2: CiteSeerX Manually labeled datasets.

Dataset	Size	Description
Document type classification	3000	PDF documents labeled as papers, theses, slides, books, resumes, etc.
Author name disambiguation	8500	Author mentions from 600 individuals.
Paper entity matching	1376	Matching pairs between CiteSeerX and external databases.
CiteSeerX-2018	4.5M	Cleansed paper metadata

of poor maintenance, such as coding, lack of documentation, system frailty, and IT support. (2) Research and the system need to be strongly coupled. Research provides cutting-edge tech support for the system while the system provides *real* data and test beds for research. (3) The system must provide a reliable and unique service to maintain a considerable user population. For CiteSeerX, this means complete open source software (where possible) and data including all documents.

5 FUTURE PROSPECTS

The future of CiteSeerX relies on both research and system development. Metadata extraction has made great progress in the past decade, but much of the text is still relatively unexplored. For example, although there are OA data repositories (e.g., figshare) and search engines (e.g., Google Data Search), a myriad of datasets mentioned in academic papers that are not discovered or used. We propose research on AI-based algorithms that extract *datasets* and their associated metadata from the full text of academic papers in multiple domains. Two of the biggest challenges are the lack of domain knowledge and the large amount of training data. In CiteSeerX, papers in different scientific domains are all together. It would be useful to first classify them by subject categories. In a preliminary study, we used feature-based machine learning models and a multilayer perceptron to classify papers into 6 categories and achieved a micro-$F_1 \approx 0.83$ [25]. We are also experimenting with deep neural networks in order to expand the current method to 104 scientific categories. To build the labeled corpus, we will first build author profiles and then request annotations from authors and readers.

From a sustainability perspective, we are investigating a four sided model that will sustain CiteSeerX for the next 10 years. This includes (1) increasing coverage and freshness of the collection, (2) improving metadata quality with state-of-the-art extractors and data cleansing modules, (3) employing ElasticSearch as an indexer *and* a metadata storage in place of a database; and (4) enriching semantic data extracted from full text.

ACKNOWLEDGMENTS

We gratefully acknowledge partial support from National Science Foundation and thank the reviewers for constructive comments.

REFERENCES

[1] Rabah A. Al-Zaidy and C. Lee Giles. 2017. A Machine Learning Approach for Semantic Structuring of Scientific Charts in Scholarly Documents *(AAAI'17)*. 4644–4649. http://aaai.org/ocs/index.php/IAAI/IAAI17/paper/view/14275

[2] Cornelia Caragea, Florin Adrian Bulgarov, Andreea Godea, and Sujatha Das Gollapalli. 2014. Citation-Enhanced Keyphrase Extraction from Research Papers: A Supervised Approach *(EMNLP'14)*. 1435–1446. http://aclweb.org/anthology/D/D14/D14-1150.pdf

[3] Cornelia Caragea, Jian Wu, Sujatha Das Gollapalli, and C. Lee Giles. 2016. Document Type Classification in Online Digital Libraries *(AAAI'16)*. 3997–4002. http://www.aaai.org/ocs/index.php/IAAI/IAAI16/paper/view/12343

[4] Christopher Clark and Santosh Kumar Divvala. 2016. PDFFigures 2.0: Mining Figures from Research Papers *(JCDL'16)*. 143–152. https://doi.org/10.1145/2910896.2910904

[5] Isaac Councill, C Lee Giles, and Min-Yen Kan. 2008. ParsCit: an Open-source CRF Reference String Parsing Package *(LREC'08)*.

[6] I. G. Councill, C. L. Giles, E. Di Iorio, M. Gori, M. Maggini, and A. Pucci. [n.d.]. Towards Next Generation Citeseer: A Flexible Architecture for Digital Library Deployment *(ECDL'06)*. 111–122. https://doi.org/10.1007/11863878_10

[7] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 2016. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise *(KDD'96)*. 226–231. http://www.aaai.org/Library/KDD/1996/kdd96-037.php

[8] C. Lee Giles, Kurt D. Bollacker, and Steve Lawrence. 1998. CiteSeer: An Automatic Citation Indexing System *(JCDL'98)*. 89–98. https://doi.org/10.1145/276675.276685

[9] Hui Han, C. Lee Giles, Eren Manavoglu, Hongyuan Zha, Zhenyue Zhang, and Edward A. Fox. 2003. Automatic document metadata extraction using support vector machines *(JCDL'03)*. 37–48. http://dl.acm.org/citation.cfm?id=827140.827146

[10] Hui Han, C. Lee Giles, Hongyuan Zha, Cheng Li, and Kostas Tsioutsiouliklis. [n.d.]. Two supervised learning approaches for name disambiguation in author citations *(JCDL'04)*. 296–305. https://doi.org/10.1145/996350.996419

[11] Jian Huang, Seyda Ertekin, and C. Lee Giles. 2006. Efficient Name Disambiguation for Large-Scale Databases *(PKDD'06)*. 536–544. https://doi.org/10.1007/11871637_53

[12] Madian Khabsa and C. Lee Giles. 2015. Chemical entity extraction using CRF and an ensemble of extractors. *Journal of Cheminformatics* 7, 1 (2015), S12. https://doi.org/10.1186/1758-2946-7-S1-S12

[13] Madian Khabsa, Pucktada Treeratpituk, and C. Lee Giles. 2015. Online Person Name Disambiguation with Constraints *(JCDL'15)*. 37–46. https://doi.org/10.1145/2756406.2756915

[14] PederOlesen Larsen and Markus von Ins. 2010. The rate of growth in scientific publication and the decline in coverage provided by Science Citation Index. *Scientometrics* 84, 3 (2010), 575–603. https://doi.org/10.1007/s11192-010-0202-z

[15] Mario Lipinski, Kevin Yao, Corinna Breitinger, Joeran Beel, and Bela Gipp. 2013. Evaluation of Header Metadata Extraction Approaches and Tools for Scientific PDF Documents *(JCDL'13)*. 385–386. https://doi.org/10.1145/2467696.2467753

[16] Ying Liu, Prasenjit Mitra, and C. Lee Giles. 2008. Identifying table boundaries in digital documents via sparse line detection *(CIKM'08)*. 1311–1320. https://doi.org/10.1145/1458082.1458255

[17] Patrice Lopez. 2009. GROBID: Combining Automatic Bibliographic Data Recognition and Term Extraction for Scholarship Publications *(ECDL'09)*. 473–474. http://dl.acm.org/citation.cfm?id=1812799.1812875

[18] Athar Sefid, Jian Wu, Allen C. Ge, Jing Zhao, Lu Liu, Cornelia Caragea, Prasenjit Mitra, and C. Lee Giles. 2019. Cleansing Noisy and Heterogeneous Metadata for Record Linking Across Scholarly Big Datasets *(IAAI'19)*.

[19] Yang Song, Jian Huang, Isaac G. Councill, Jia Li, and C. Lee Giles. 2007. Generative models for name disambiguation *(WWW'07)*. 1163–1164. https://doi.org/10.1145/1242572.1242746

[20] Pradeep B. Teregowda, Isaac G. Councill, R. Juan Pablo Fernández, Madian Khabsa, Shuyi Zheng, and C. Lee Giles. 2010. SeerSuite: Developing a Scalable and Reliable Application Framework for Building Digital Libraries by Crawling the Web *(WebApps'10)*. 14–14. http://dl.acm.org/citation.cfm?id=1863166.1863180

[21] Pucktada Treeratpituk and C. Lee Giles. 2009. Disambiguating authors in academic publications using random forests *(JCDL'09)*. 39–48. https://doi.org/10.1145/1555400.1555408

[22] Suppawong Tuarob, Sumit Bhatia, Prasenjit Mitra, and C. Lee Giles. 2013. Automatic Detection of Pseudocodes in Scholarly Documents Using Machine Learning *(ICDAR'13)*. 738–742. https://doi.org/10.1109/ICDAR.2013.151

[23] Kyle Williams, Jian Wu, Sagnik Ray Choudhury, Madian Khabsa, and C. Lee Giles. [n.d.]. Scholarly big data information extraction and integration in the CiteSeerX digital library *(ICDEW'14)*. 68–73.

[24] Jian Wu, Sagnik Ray Choudhury, Agnese Chiatti, Chen Liang, and C. Lee Giles. 2017. HESDK: A Hybrid Approach to Extracting Scientific Domain Knowledge Entities *(JCDL'17)*. 241–244.

[25] Jian Wu, Bharath Kandimalla, Shaurya Rohatgi, Athar Sefid, Jianyu Mao, and C. Lee Giles. 2018. CiteSeerX-2018: A Cleansed Multidisciplinary Scholarly Big Dataset *(BigData'18)*. 5465–5467. https://doi.org/10.1109/BigData.2018.8622114

[26] Jian Wu, Athar Sefid, Allen C. Ge, and C. Lee Giles. 2017. A Supervised Learning Approach To Entity Matching Between Scholarly Big Datasets *(K-CAP'17)*. Article 42, 4 pages. https://doi.org/10.1145/3148011.3154470

[27] Jian Wu, Pradeep B. Teregowda, Kyle Williams, Madian Khabsa, Douglas Jordan, Eric Treece, Zhaohui Wu, and C. Lee Giles. 2014. Migrating a Digital Library to a Private Cloud *(IC2E'14)*. https://doi.org/10.1109/IC2E.2014.77

[28] Jian Wu, Kyle Williams, Hung-Hsuan Chen, Madian Khabsa, Cornelia Caragea, Alexander Ororbia, Douglas Jordan, and C. Lee Giles. 2014. CiteSeerX: AI in a Digital Library Search Engine *(IAAI'14)*. 2930–2937. http://www.aaai.org/ocs/index.php/IAAI/IAAI14/paper/view/8607

[29] Baichuan Zhang and Mohammad Al Hasan. 2017. Name Disambiguation in Anonymized Graphs using Network Embedding *(CIKM'17)*. 1239–1248. https://doi.org/10.1145/3132847.3132873

[30] Wei Zhong and Richard Zanibbi. 2019. Structural Similarity Search for Formulas Using Leaf-Root Paths in Operator Subtrees *(ECIR'19)*. 116–129. https://doi.org/10.1007/978-3-030-15712-8_8