Refcat: The Internet Archive Scholar Citation Graph

Martin Czygan

Internet Archive
San Francisco, California, USA
martin@archive.org

Bryan Newbold

Internet Archive
San Francisco, California, USA
bnewbold@archive.org

Abstract

As part of its scholarly data efforts, the Internet Archive (IA) releases a first version of a citation graph dataset, named *refcat*, derived from scholarly publications and additional data sources. It is composed of data gathered by the fatcat cataloging project[1] (the catalog that underpins IA Scholar), related web-scale crawls targeting primary and secondary scholarly outputs, as well as metadata from the Open Library[2] project and Wikipedia[3]. This first version of the graph consists of over 1.3B citations. We release this dataset under a CC0 Public Domain Dedication, accessible through Internet Archive[4]. The source code used for the derivation process, including exact and fuzzy citation matching, is released under an MIT license[5]. The goal of this report is to describe briefly the current contents and the derivation of the dataset.

Index terms— Citation Graph, Web Archiving

1 Introduction

The Internet Archive released a first version of a citation graph dataset derived from a corpus of about 2.5B raw references[6] gathered from 63,296,308 metadata records (which are collected from various sources or based on data obtained by PDF extraction and annotation tools such as GROBID [Lopez, 2009]). Additionally, we consider integration with metadata from Open Library and Wikipedia. We expect this dataset to be iterated upon, with changes both in content and processing.

According to [Jinha, 2010] over 50M scholarly articles have been published (from 1726) up to 2009, with the rate of publications on the rise [Landhuis, 2016]. In 2014, a study based on academic search engines estimated that at least 114M English-language scholarly documents are accessible on the web [Khabsa and Giles, 2014].

Modern citation indexes can be traced back to the early computing age, when projects like the Science Citation Index (1955) [Garfield, 2007] were first devised, living on in commercial knowledge bases today. Open alternatives were started such as the Open Citations Corpus (OCC) in 2010 - the first version of which contained 6,325,178 individual references [Shotton, 2013]. Other notable projects include CiteSeer [Giles et al., 1998], CiteSeerX [Wu et al., 2019] and CitEc[7]. The last decade has seen the emergence of more openly available, large scale citation projects like Microsoft Academic [Sinha et al., 2015] and the Initiative for Open Citations[8] [Shotton, 2018]. In 2021, over one billion citations are publicly available, marking a "tipping point" for this category of data [Hutchins, 2021].

While a paper will often cite other papers, more citable entities exist such as books or web links and within links a variety of targets, such as web pages, reference entries, protocols or datasets. References can be extracted manually or through more automated methods, by accessing relevant metadata or structured data extraction from full text documents. Automated methods offer the benefits of scalability. The completeness of bibliographic metadata in references ranges from documents with one or more persistent identifiers to raw, potentially unclean strings partially describing a scholarly artifact.

[1] https://fatcat.wiki
[2] https://openlibrary.org
[3] https://wikipedia.org
[4] https://archive.org/details/refcat_2021-07-28
[5] https://gitlab.com/internetarchive/refcat
[6] Number of raw references: 2,507,793,772
[7] https://citec.repec.org
[8] https://i4oc.org

2 Related Work

Two typical problems in citation graph development are related to data acquisition and citation matching. Data acquisition itself can take different forms: bibliographic metadata can contain explicit reference data as provided by publishers and aggregators; this data can be relatively consistent when looked at per source, but may vary in style and comprehensiveness when looked at as a whole. Another way of acquiring bibliographic metadata is to analyze a source document, such as a PDF (or its text), directly. Tools in this category are often based on conditional random fields [Lafferty et al., 2001] and have been implemented in projects such as ParsCit [Councill et al., 2008], Cermine [Tkaczyk et al., 2014], EXCITE [Hosseini et al., 2019] or GROBID [Lopez, 2009].

The problem of citation matching is relatively simple when common, persistent identifiers are present in the data. Complications mount, when there is *Identity Uncertainty*, that is "objects are not labeled with unique identifiers or when those identifiers may not be perceived perfectly" [Pasula et al., 2003]. CiteSeer has been an early project concerned with citation matching [Giles et al., 1998]. A taxonomy of potential issues common in the matching process has been compiled by [Olensky et al., 2016]. Additional care is required, when the citation matching process is done at scale [Fedoryszak et al., 2013]. The problem of heterogenity has been discussed in the context of datasets by [Mathiak and Boland, 2015].

Projects and datasets centered around citations or containing citation data as a core component are COCI, the "OpenCitations Index of Crossref open DOI-to-DOI citations", which was first released 2018-07-29[9] and has been regularly updated [Peroni and Shotton, 2020]. The WikiCite[10] project, "a Wikimedia initiative to develop open citations and linked bibliographic data to serve free knowledge" continuously adds citations to its database[11]. Microsoft Academic Graph [Sinha et al., 2015] is comprised of a number of entities[12] with *PaperReferences* being one relation among many others.

3 Dataset

We released the first version of the *refcat* dataset in a format used internally for storage and to serve queries (and which we call *biblioref* or *bref* for short). The dataset includes metadata from fatcat (the catalog underpinning IA Scholar), the Open Library project and inbound links from the English Wikipedia. The dataset is integrated into the fatcat.wiki website and allows users to explore inbound and outbound references[13].

Set		Count
COCIv11 (C)		1,186,958,897
refcat-doi (R)		1,303,424,212
$C \cap R$	overlap	1,046,438,515
$C \setminus R$	COCIv11 only	140,520,382
$R \setminus C$	*refcat-doi* only	256,985,697

Table 1: Comparison between Open Citations COCI corpus (v11, 2021-09-04) and *refcat-doi*, a subset of *refcat* where entities have a known DOI. At least 150,727,673 (58.7%) of the 256,985,697 references in *refcat-doi* only record links within a specific dataset provider; here GBIF with DOI prefix: 10.15468.

Edge type	Count
doi-doi	1,303,424,212
target-open-library	20,307,064
source-wikipedia	1,386,941

Table 2: Counts of classic DOI to DOI references as well as outbound references matched against Open Library as well as inbound references from the English Wikipedia.

The format records source and target identifiers, a few metadata attributes (such as year or release stage, i.e. preprint, version of record, etc) as well as information about the match status and provenance.

The dataset currently contains 1,323,423,672 citations across 76,327,662 entities (55,123,635 unique source and 60,244,206 unique target work identifiers; for 1,303,424,212 - or 98.49% of all citations - we do have a DOI for both source and target). The majority of matches - 1,250,523,321 - is established through identifier based matching (DOI, PMIC, PMCID, ARXIV, ISBN). 72,900,351 citations are established through fuzzy matching techniques, where references did not contain identifiers[14]. Citations from the Open Citations' COCI corpus[15] and *refcat* overlap to the most part, as can be seen in Table 1.

We started to include non-traditional citations into the graph, such as links to books included in Open Library

[9] https://opencitations.net/download

[10] https://meta.wikimedia.org/wiki/WikiCite

[11] http://wikicite.org/statistics.html

[12] https://docs.microsoft.com/en-us/academic-services/graph/reference-data-schema

[13] https://guide.fatcat.wiki/reference_graph.html

[14] This not necessary mean that the records in question do not have an identifier; however if an identifier existed, it was not part of the raw reference

[15] Reference dataset COCI v11, released 2021-09-04, http://opencitations.net/index/coci

and links from the English Wikipedia to scholarly works. For links between Open Library we employ both identifier based and fuzzy matching; for Wikipedia references we used a published dataset [Singh et al., 2020] and we are contributing to upstream projects related to wikipedia citation extraction, such as *wikiciteparser*[16] to generate updates from recent Wikipedia dumps[17]. Table 2 lists the counts for these links. Additionally, we are examining web links appearing in references: after an initial cleaning procedure we currently find 25,405,592 web links[18] in the reference corpus, of which 4,828,283 (19%) have been preserved as of August 2021 with an HTTP 200 status code in the Wayback Machine[19] of the Internet Archive. As an upper bound - if we additionally include all redirection (HTTP 3XX) and server error status codes (HTTP 5XX) - we find a total of 14,306,019 (56.3%) links preserved.

We ran a live URL check[20] over a sample of 364415 links which appear in the reference corpus *and* have a HTTP 200 status code archival copy in the Wayback Machine. Of the 364415 links we find 305476 (83.8%) responding with an HTTP 200 OK, whereas the rest of the links yield a variety of HTTP status codes, like 404, 403, 500 and others - resulting in about 16% of the links in the reference corpus preserved at the Internet Archive being currently inaccessible on the web[21] - making targeted web crawling and preservation of scholarly references a key activity for maintaining citation integrity.

4 System Design

4.1 Constraints

The constraints for the system design are informed by the volume and the variety of the data. The capability to run the whole graph derivation on a single machine[22] was a minor goal as well. In total, the raw inputs amount to a few terabytes of textual content, mostly newline delimited JSON. More importantly, while the number of data fields is low, certain documents are very partial with hundreds of different combinations of available field values found in the raw reference data. This is most likely caused by aggregators passing on reference data coming from hundreds of sources, each of which not necessarily agreeing on a common granularity for citation data and from artifacts of machine learning based structured data extraction tools.

Each combination of fields may require a slightly different processing path. For example, references with an Arxiv identifier can be processed differently from references with only a title.

4.2 Data Sources

Reference data comes from two main sources: explicit bibliographic metadata and PDF extraction. The bibliographic metadata is taken from fatcat, which itself harvests and imports web accessible sources such as Crossref, Pubmed, Arxiv, Datacite, DOAJ, dblp and others into its catalog (as the source permits, data is processed continuously or in batches). Reference data from PDF documents has been extracted with GROBID[23], with the TEI-XML results being cached locally in a key-value store accessible with an S3 API[24]. Archived PDF documents result from dedicated web-scale crawls of scholarly domains conducted with multiple open-source crawler technologies created by the Internet Archive and a variety of seed lists targeting journal homepages, repositories, dataset providers, aggregators, web archives and other venues. A processing pipeline merges catalog data from the primary database and cached data from the key-value store and generates the set of about 2.5B references records, which currently serve as an input for the citation graph derivation pipeline.

4.3 Methodology

Overall, a map-reduce style [Dean and Ghemawat, 2010] approach is followed[25], which allows for some uniformity in the processing. We extract *(key, document)* tuples (as TSV) from the raw JSON data and sort by key. We then group documents with the same key and apply a function on each group in order to generate our target schema or perform additional operations such as deduplication or fusion of matched and unmatched references for indexing.

The key derivation can be exact (via an identifier like DOI, PMID, etc) or based on a value normalization, like "slugifying" a title string. For identifier based matches we can generate the target schema directly. For fuzzy matching candidates, we pass possible match pairs through a verification procedure, which is implemented for *release entity*[26] pairs. This procedure is a domain dependent rule based

[16] https://github.com/dissemin/wikiciteparser

[17] Wikipedia dumps are available on a monthly basis from https://dumps.wikimedia.org/.

[18] The cleaning process is necessary because OCR artifacts and other metadata issues exist in the data. Unfortunately, even after cleaning not all links will be in the form as originally intended by the authors.

[19] https://archive.org/web/

[20] All links accessed on 2021-10-04 and 2021-10-05.

[21] We used the https://github.com/miku/clinker command line link checking tool.

[22] We used a shared virtual server with 24 cores and 48G of main memory. The most memory-intensive part of the processing currently are the buffers set aside for *GNU sort*.

[23] GROBID v0.5.5

[24] Currently, https://github.com/chrislusf/seaweedfs is used

[25] While the operations are similar, the processing is not distributed but runs on a single machine. For space efficiency, zstd [Collet and Kucherawy, 2018] is used to compress raw data and derivations.

[26] https://guide.fatcat.wiki/entity_release.html.

verification, able to identify different versions of a publication, preprint-published pairs and documents, which are are similar by various metrics calculated over title and author fields. The fuzzy matching approach is applied on all reference documents without any identifier (a title is currently required).

We currently implement performance sensitive parts in the Go programming language[27], with various processing stages (e.g. conversion, map, reduce, ...) represented by separate command line tools. A thin task orchestration layer using the luigi framework[28] allows for experimentation in the pipeline and for single command derivations, as data dependencies are encoded with the help of the orchestrator. Within the tasks, we also utilize classic platform tools such as GNU *sort* [McIlroy, 1971].

During a last processing step, we fuse reference matches and unmatched items into a single, indexable file. This step includes deduplication of different matching methods (e.g. prefer exact matches over fuzzy matches). This file is indexed into a search index and serves both matched and unmatched references for the web application, allowing for further collection of feedback on match quality and possible improvements.

With a few schema conversions, fuzzy matching has been be applied to Wikipedia articles and Open Library (edition) records as well. The aspect of precision and recall are represented by the two stages: we are generous in the match candidate generation phase in order to improve recall, but we are strict during verification, in order to control precision. Quality assurance for verification is implemented through a growing list of test cases of real examples from the catalog and their expected or desired match status[29].

5 Limitations and Future Work

As with other datasets in this field we expect this dataset to be iterated upon.

- The fatcat catalog updates its metadata continuously[30] and web crawls are conducted regularly. Current processing pipelines cover raw reference snapshot creation and derivation of the graph structure, which allows to rerun the processing pipeline based on updated data as it becomes available.
- Metadata extraction from PDFs depends on supervised machine learning models, which in turn depend on available training datasets. With additional crawls and metadata available we hope to improve models used for metadata extraction, improving yield and reducing data extraction artifacts in the process.
- As of this version, a number of raw reference docs remain unmatched, which means that neither exact nor fuzzy matching has detected a link to a known entity. Metadata might be missing. However, parts of the data will contain a reference to a catalogued entity, but in a specific, dense and harder to recover form.
- The reference dataset contains millions of URLs and their integration into the graph has been implemented as a prototype. A full implementation requires a few data cleanup and normalization steps.

[27] https://golang.org/

[28] https://github.com/spotify/luigi [Bernhardsson and Freider, 2018], which has been used in various scientific pipeline application, like [Schulz et al., 2016], [Erdmann et al., 2017], [Lampa et al., 2019], [Czygan, 2014] and others.

[29] The list can be found under: https://gitlab.com/internetarchive/refcat/-/blob/master/skate/testdata/verify.csv. It is helpful to keep this test suite independent of any specific programming language.

[30] A changelog can currently be followed here: https://fatcat.wiki/changelog.

6 Acknowledgements

This work is partially supported by grants from the *Andrew W. Mellon Foundation*, especially "Ensuring the Persistent Access of Open Access Journal Literature: Phase II" (1910-07256, Jefferson Bailey, Principal Investigator).

Appendix: Reference Relations

Figure 1 shows the schematic reference relations.

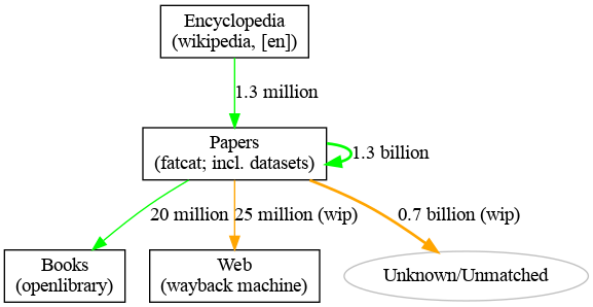

Figure 1: Schematics of the main reference entities; green: included in the corpus; orange: currently in development; gray: Planned, but not in development; red: long-term desiderata.

Appendix: Data Quality

A note on data quality: While we implement various data quality measures, real-world data, especially coming from many different sources will contain issues. Among other measures, we keep track of match reasons, especially for

fuzzy matching to be able to zoom in on systematic errors more easily (see Table 3).

Count	Provenance	Status	Reason
934932865	crossref	exact	doi
151366108	fatcat-datacite	exact	doi
65345275	fatcat-pubmed	exact	pmid
48778607	fuzzy	strong	jaccardauthors
42465250	grobid	exact	doi
29197902	fatcat-pubmed	exact	doi
19996327	fatcat-crossref	exact	doi
11996694	fuzzy	strong	slugtitleauthormatch
9157498	fuzzy	strong	tokenizedauthors
3547594	grobid	exact	arxiv
2310025	fuzzy	exact	titleauthormatch
1496515	grobid	exact	pmid
680722	crossref	strong	jaccardauthors
476331	fuzzy	strong	versioneddoi
449271	grobid	exact	isbn
230645	fatcat-crossref	strong	jaccardauthors
190578	grobid	strong	jaccardauthors
156657	crossref	exact	isbn
123681	fatcat-pubmed	strong	jaccardauthors
79328	crossref	exact	arxiv
57414	crossref	strong	tokenizedauthors
53480	fuzzy	strong	pmiddoipair
52453	fuzzy	strong	dataciterelatedid
47119	grobid	strong	slugtitleauthormatch
36774	fuzzy	strong	arxivversion

Table 3: Table of match counts (top 25), reference provenance, match status and match reason. Provenance currently can name the raw origin (e.g. *crossref*) or the method (e.g. *fuzzy*). The match reason identifier encodes a specific rule in the domain dependent verification process and is included for completeness - we do not include the details of each rule in this report.

References

E Bernhardsson and E Freider. Rouhani a. spotify/luigi-github, 2018.

Yann Collet and Murray Kucherawy. Zstandard compression and the application/zstd media type. *RFC 8478*, 2018.

Isaac G Councill, C Lee Giles, and Min-Yen Kan. Parscit: an open-source crf reference string parsing package. In *LREC*, volume 8, pages 661–667, 2008.

Martin Czygan. Design and implementation of a library metadata management framework and its application in fuzzy data deduplication and data reconciliation with authority data. *Informatik 2014*, 2014.

Jeffrey Dean and Sanjay Ghemawat. Mapreduce: a flexible data processing tool. *Communications of the ACM*, 53 (1):72–77, 2010.

M Erdmann, B Fischer, R Fischer, and M Rieger. Design and execution of make-like, distributed analyses based on spotify's pipelining package luigi. In *Journal of Physics: Conference Series*, volume 898, page 072047. IOP Publishing, 2017.

Mateusz Fedoryszak, Dominika Tkaczyk, and Łukasz Bolikowski. Large scale citation matching using apache hadoop. In *International Conference on Theory and Practice of Digital Libraries*, pages 362–365. Springer, 2013.

Eugene Garfield. The evolution of the science citation index. *International microbiology*, 10(1):65, 2007.

C Lee Giles, Kurt D Bollacker, and Steve Lawrence. Citeseer: An automatic citation indexing system. In *Proceedings of the third ACM conference on Digital libraries*, pages 89–98, 1998.

Azam Hosseini, Behnam Ghavimi, Zeyd Boukhers, and Philipp Mayr. Excite–a toolchain to extract, match and publish open literature references. In *2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 432–433. IEEE, 2019.

B Ian Hutchins. A tipping point for open citation data. *Quantitative Science Studies*, pages 1–5, 2021.

Jinha. Article 50 million: an estimate of the number of scholarly articles in existence. 23, Jul 2010. doi:10.1087/20100308.

Khabsa and Giles. The number of scholarly documents on the public web. May 2014. doi:10.1371/journal.pone.0093949.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

Samuel Lampa, Martin Dahlö, Jonathan Alvarsson, and Ola Spjuth. Scipipe: A workflow library for agile development of complex and dynamic bioinformatics pipelines. *GigaScience*, 8(5):giz044, 2019.

Landhuis. Scientific literature: Information overload. 535 (7612), Jul 2016. doi:10.1038/nj7612-457a.

Patrice Lopez. Grobid: Combining automatic bibliographic data recognition and term extraction for scholarship publications. In *International conference on theory and practice of digital libraries*, pages 473–474. Springer, 2009.

Brigitte Mathiak and Katarina Boland. Challenges in matching dataset citation strings to datasets in social science. *D-Lib Magazine*, 21(1/2):23–28, 2015.

M Douglas McIlroy. A research unix reader: annotated excerpts from the programmer's manual. 1971.

Marlies Olensky, Marion Schmidt, and Nees Jan van Eck. Evaluation of the citation matching algorithms of cwts and i fq in comparison to the w eb of science. *Journal of the Association for Information Science and Technology*, 67(10):2550–2564, 2016.

Hanna Pasula, Bhaskara Marthi, Brian Milch, Stuart J Russell, and Ilya Shpitser. Identity uncertainty and citation matching. In *Advances in neural information processing systems*, pages 1425–1432, 2003.

Silvio Peroni and David Shotton. Opencitations, an infrastructure organization for open scholarship. *Quantitative Science Studies*, 1(1):428–444, 2020.

Wade L Schulz, Thomas JS Durant, Alexa J Siddon, and Richard Torres. Use of application containers and workflows for genomic data analysis. *Journal of pathology informatics*, 7, 2016.

David Shotton. Publishing: open citations. *Nature News*, 502(7471):295, 2013.

David Shotton. Funders should mandate open citations. *Nature*, 553(7686):129–130, 2018.

Harshdeep Singh, Robert West, and Giovanni Colavizza. Wikipedia Citations: A comprehensive dataset of citations with identifiers extracted from English Wikipedia, July 2020. URL `https://doi.org/10.5281/zenodo.3940692`.

Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June Hsu, and Kuansan Wang. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th international conference on world wide web*, pages 243–246, 2015.

Dominika Tkaczyk, Pawel Szostek, Piotr Jan Dendek, Mateusz Fedoryszak, and Lukasz Bolikowski. Cermine–automatic extraction of metadata and references from scientific literature. In *2014 11th IAPR International Workshop on Document Analysis Systems*, pages 217–221. IEEE, 2014.

Jian Wu, Kunho Kim, and C Lee Giles. Citeseerx: 20 years of service to scholarly big data. In *Proceedings of the Conference on Artificial Intelligence for Data Discovery and Reuse*, pages 1–4, 2019.