Expanding the Citation Graph for Data Citations*

Peter Buneman[1], Dennis Dosso[2], Matteo Lissandrini[3] and Gianmaria Silvello[2]

[1]*University of Edinburgh, UK*
[2]*University of Padua, Italy*
[3]*Aalborg University, Denmark*

Abstract

The Citation Graph (CG) is a computational artifact widely used to represent the domain of published literature. There is an increasing demand to treat the publication of data in the same way that we treat conventional publications. It should be possible to cite data for the same reasons that is is necessary to cite other publications. In this paper we see some of the limitations of the citation graph, and we discuss how some implementation-agnostic extensions may solve them, thus also allowing the introduction of data and the management of data citations within the CG.

Keywords

Data Citation, Bibliometrics, Citation Graph

1. Introduction

Citations are essential to all forms of research: they are necessary, among other things, to identify the cited material, to retrieve it, and to give credit to its creators. The Citation Graph (CG) is a model that describes how citations link research entities, typically papers, journals, and books [2, 3]. It supports activities such as authorship tracking, discovery of new publications, and the computation of bibliometrics. Several implementations of the CG are available such as Google Scholar, Microsoft Academic Graph (MAG)[1], Scopus[2], and Web of Science (WoS)[3].

Much research now relies on curated databases, which have largely replaced traditional reference works and now play a crucial role in science [4]. There is now a strong demand to give databases the same scholarly status of traditional scientific works [5, 6], and it is a common opinion that citations to data should be given the same scholarly status as traditional citations and that they should contribute to bibliometric indicators [7, 8]. However, currently, no citation-based system properly takes into account scientific databases and data citations.

SEBD 2022: The 30th Italian Symposium on Advanced Database Systems, June 19-22, 2022, Tirrenia (PI), Italy

∗*The full paper was published in the Quantitative Science Studies journal [1].*

opb@inf.ed.ac.uk (P. Buneman); dennis.dosso@unipd.it (D. Dosso); matteo@cs.aau.dk (M. Lissandrini); gianmaria.silvello@unipd.it (G. Silvello)

http://www.dei.unipd.it/~dosso/ (D. Dosso); https://disi.unitn.it/~lissandrini/ (M. Lissandrini); http://www.dei.unipd.it/~silvello/ (G. Silvello)

0000-0001-7307-4607 (D. Dosso); 0000-0001-7922-5998 (M. Lissandrini); 0000-0003-4970-4554 (G. Silvello)

CEUR Workshop Proceedings (CEUR-WS.org)

[1] https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/

[2] https://www.scopus.com/home.uri

[3] https://clarivate.com/webofsciencegroup/solutions/web-of-science/

We claim that the current model of the citation graph cannot properly accommodate data citations. Two significant features are missing or poorly represented in CGs: (i) the lack of the representation of *context* of a citation, and (ii) the *versioning* of the publications (e.g., papers, databases, data subsets, for which new updated versions or corrections are published through time). These shortcomings limit the management of data objects, their citations, and also the representation of traditional publications and their connections.

To overcome these limitations and to introduce data citations in the CG, we propose and discuss two extensions of this model. The first one, *reference annotation*, models the information that may accompany a citation, such as its context for traditional citation or the query used to obtain the data, for a data citation. Through the information contained in the reference annotations, it is possible to add data and count data citations. The second extension is a new relationship among publications called *subsumption*. A publication subsumes another one when it takes its place in receiving citations,for instance when a new version of the publication is released. Moreover, using subsumption, it is possible to model citations to an evolving database.

The paper is organized as follows: Section 2 presents in further detail some core concepts and the existing limits of the CG, Section 3 presents our proposed extensions to overcome these limits, and how they can be exploited to add data and data citations in the CG, finally, Section 3.4 presents our conclusions.

2. Core concepts and limits of the CG

2.1. Core concepts

Citable Unit A *Citable Unit* (CU) is a published entity, such as a paper, a chapter, or portion of data or software, which presents all the necessary qualities to be considered as a "citable work". The characterization we adopt requires the CU to be: uniquely and unambiguously *identifiable* and *citable*; *available* in perpetuity and in *unchanged form*; *accessible*; *self-contained* and *complete* [9]. Although some of these requirements are subjective, and not straightforward in databases, they still provide a workable starting point. One of the most problematic aspect for a CU belonging to a database is for it to be unchanged, since databases evolve, and creating a CU for each version may be counterproductive. We also note that a CU may contain other citable units, such as the chapters forming a book. There is thus a "part-of" relationship between CUs that we discuss later.

Reference Usually scientific papers present a list of *references* at their end. Traditionally, a reference is a *pointer* to another publication in the literature, comprising also a brief *description* of it. It is a short text composed by fields representing metadata such as the title of the publication, its authors, the publication year, its venue, and other metadata. This information enables us to identify and find the entity. We note that contents of a reference are determined by the cited entity. That is, to within stylistic variation, the contents of a reference will be the same in any paper that cites the entity. In the citation graph, the presence of a reference determines a directed edge between nodes representing the citing CU and the cited CU.

Reference pointer, reference annotation, and citation Generally speaking, there is no universal agreement on the distinction between the concept of *reference* and *citation*, and these two terms are often used interchangeably [10, 11, 12]. In the body of a paper we may find, for example, a textual artifact such as "Austen, J. (2004). pp 101-104". We call the first part of this artifact, "Austen, J. (2004)", *reference pointer.* It is used to denote a single bibliographic reference in the reference section when mentioned in the body of the paper. Such pointers may be accompanied by some additional information. In our example, this additional information is provided by the text "pp 101-104", that specifies the exact location in the paper the citation is referring to. The same reference pointer therefore may appear many times in the same paper, each time with different additional information. We call this additional information *reference annotation*; it is not part of the reference itself and will depend on how and where the reference is used. This information can be thought as a form of *context.*

Finally, we can define a *citation* to be the combination of a reference pointer with the (optional) reference annotation.

Part-of A paper may appear in a collection of papers or in the proceedings of a conference; where both the paper and the collection are CUs. Databases and datasets often have a hierarchical structure; for example many datasets have a simple directory structure. In addition to accommodating reference pointers the citation graph may need to represent a *part-of* relationship citable units in the graph.

2.2. Limits of the Citation Graph

The basic model of CG consists of a directed graph $\mathcal{G} = \{V, E\}$, where V is the set of papers and $E \subseteq V \times V$ is the set of citations, where an edge $\langle p_1, p_2 \rangle$ signals the presence of a citation from paper p_1 to paper p_2. The following limitations of this model can already be seen in the more traditional systems that model the citations among paper, and are a hindrance also for the introduction of data citation.

Lack of context The nodes of the citation graph often contain the information of the entity they represent, but not of what we called context of the citations in this entity. The only information provided by the edge $\langle p_1, p_2 \rangle$, in fact, is that p_1 cites p_2. While we may need to know, for instance, which parts of p_2 is p_1 referencing to.

Versions Ideally, the entities in the citation graph should be clearly distinguishable between each other. However, this is not always the case and it happens that some entities in V may be quite similar to each other. This is due to many reasons, such as, but not limited to: one "abstract" version first published in some conference proceedings, and later published again in its "full version" in some journal; a paper first published in some online archive and later as a full-fledged version released as peer-reviewed publication in a conference.

For these reasons, it is not always clear which of the versions of a paper should be considered as *the* version that needs to receive the citations. Generally speaking, it is in the authors' interest to have these documents *conflated* into one, to accumulate the citations going to the different versions scattered among various locations into one representative entity. With Google Scholar,

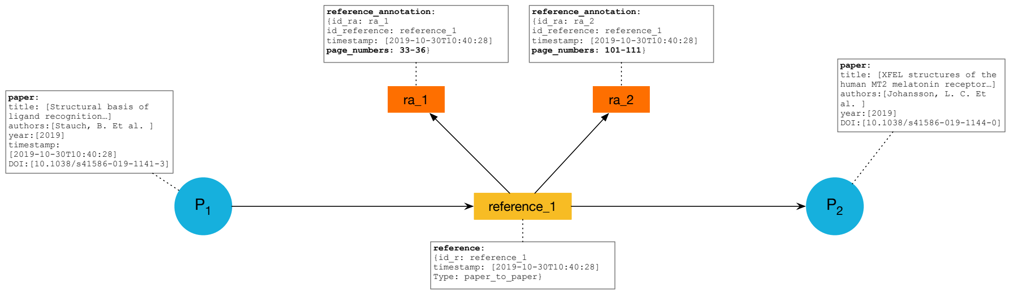
reference_annotation:
{id_ra: ra_1
id_reference: reference_1
timestamp: [2019-10-30T10:40:28]
page_numbers: 33-36}
reference_annotation:
{id_ra: ra_2
id_reference: reference_1
timestamp: [2019-10-30T10:40:28]
page_numbers: 101-111}
paper:
title: [Structural basis of ligand recognition…]
authors:[Stauch, B. Et al.]
year:[2019]
timestamp:
[2019-10-30T10:40:28]
DOI:[10.1038/s41586-019-1141-3]
ra_1
ra_2
paper:
title: [XFEL structures of the human MT2 melatonin receptor…]
authors:[Johansson, L. C. Et al.]
year:[2019]
DOI:[10.1038/s41586-019-1144-0]
P1
reference_1
P2
reference:
{id_r: reference_1
timestamp: [2019-10-30T10:40:28]
Type: paper_to_paper}

Figure 1: Exemplification of the use of references and reference annotations.

for example, it appears that all the versions being found on the Internet are clustered together in one unique “main” version, that becomes the recipient of all references coming from other entities.

Citations to data One of the primary roles of data citation is to give credit and attribution to the work of data creators and curators [5]. If integrated into the citation graph, data citations can be represented and analyzed as if they were conventional citations, with data CUs and corresponding authors receiving citations and thus credit for their work. The existing services present limitations that make data citation de facto unfeasible or limited [13]. Systems like Google Data Search[4] allows us to search for indexed data set, but they do not keep track of the citations to data, or other types of statistics, such as clicks or downloads. WoS models data citations, but only at the whole database level, as does Zenodo[5], therefore, when multiple authors contribute to the same curated database, it is impossible to distinguish to which of the authors the credit should go.

3. How to extend the graph and deal with data citations

The two extensions to the citation graph proposed in this paper already exist in limited forms in some implementations. For example, the MAG already presents the possibility to include a citation context, but it lacks the ability to accurately cite data and manage their versions. The extensions proposed in this section are independent of any specific implementation of the citation graph and, for the most part, they can directly be incorporated in the existing implementations of the CG rather than requiring a completely new implementation of the supporting database.

3.1. Reference Annotation

To address the problem of lack of context, it is necessary to annotate the CG edges that represent the references. Most data models currently implemented do not support data on edges[6], so for

[4] https://datasetsearch.research.google.com/
[5] https://zenodo.org/
[6] Property graphs are an exception to this observation.

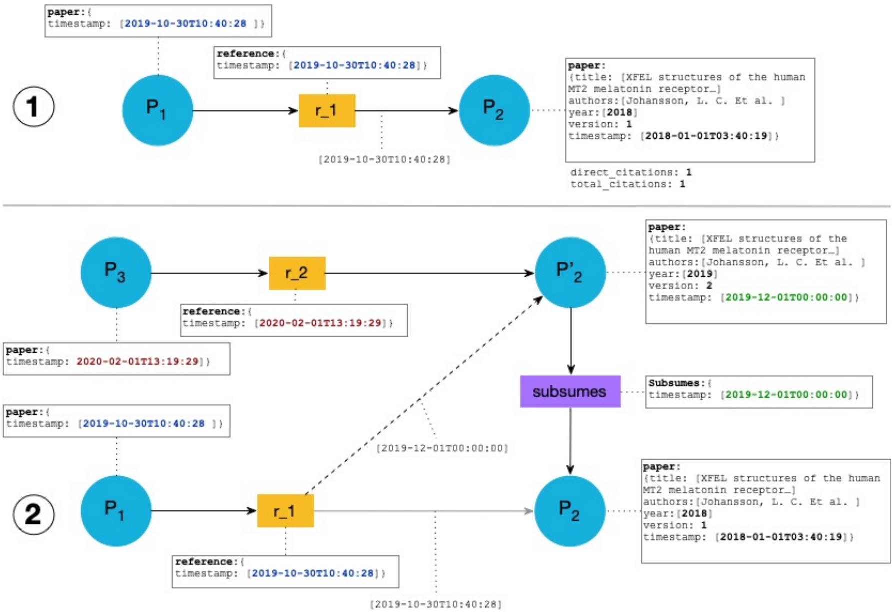
paper:{
timestamp: [2019-10-30T10:40:28]}
reference:{
timestamp: [2019-10-30T10:40:28]}
1
P1
r_1
P2
[2019-10-30T10:40:28]
paper:
{title: [XFEL structures of the human
MT2 melatonin receptor…]
authors:[Johansson, L. C. Et al.]
year:[2018]
version: 1
timestamp: [2018-01-01T03:40:19]}
direct_citations: 1
total_citations: 1
P3
r_2
P'2
paper:
{title: [XFEL structures of the
human MT2 melatonin receptor…]
authors:[Johansson, L. C. Et al.]
year:[2019]
version: 2
timestamp: [2019-12-01T00:00:00]}
reference:{
timestamp: [2020-02-01T13:19:29]}
paper:{
timestamp: 2020-02-01T13:19:29]}
subsumes
Subsumes:{
timestamp: [2019-12-01T00:00:00]}
paper:{
timestamp: [2019-10-30T10:40:28]}
[2019-12-01T00:00:00]
2
P1
r_1
P2
paper:
{title: [XFEL structures of the human
MT2 melatonin receptor…]
authors:[Johansson, L. C. Et al.]
year:[2018]
version: 1
timestamp: [2018-01-01T03:40:19]}
reference:{
timestamp: [2019-10-30T10:40:28]}
[2019-10-30T10:40:28]

Figure 2: The subsumption relation between two CUs.

consistency with these models, for our first extension we propose two new node types, rather than new kinds of edges: the *reference* and *reference annotation* nodes.

Consider Figure 1, where the paper P_1 is citing P_2. We can imagine a reference in the "References" section of P_1 as something similar to "Johansson, L. C. et al. (2019). XFEL structures of the human MT 2 melatonin receptor reveal the basis of subtype selectivity. Nature, 569 (7755), 289–292. doi: 10.1038/s41586-019-1144-0.". The existence of this reference is reflected by the presence of an edge between the two nodes. We add the reference node `reference_1` between P_1 and P_2. This new type of node contains information such as the edge type (reference), the timestamp of when the citation was registered by the system, and the reference type (in this case, from a paper to another paper). The actual information contained in a reference node can be modeled according to whatever model we decide to follow.

Let us now suppose that P_1 cites P_2 twice, that is, in P_1's body there are two reference pointers to P_2, each with its context. To model this, we add two neighbor reference annotation nodes to `reference_1`, namely `ra_1` and `ra_2`. These two contain the information describing the context of the two citations found in P_1, which may consist in references to particular pages, tables or images, comments on the nature of the citations, or some other type of metadata. The reference annotation node therefore acts as a container of all the information of the context. As we shall see, this new node is extremely helpful in introducing data citations to the CG.

3.2. Subsumption

To deal with the different versions of a CU we propose the introduction in the citation graph of a new relation, called *subsumption*.

Consider Figure 2, divided in two subsequent moments in time, t' and t''. At time t', the paper P_1 is citing paper P_2, as seen with the reference node `r_1`. Let us now consider a new version of the same paper, P_2', which is published and inserted in the citation graph at time t''. Thus P_2' subsumes P_2 indicates that the former is a new version of the latter and it is, from now on, *the* paper to consult and reference. For consistency with our approach with the reference and reference annotation nodes, we are modeling the subsumption property as a new type of node.

From time t'' onward, each new paper such as P_3 now cites P_2'. Moreover, through the subsumption, the citations that were before assigned to P_2 can now be "moved" to its latest version. However, we do not want to destroy the original link from P_1 to P_2: to do so would be to "rewrite history" and remove information from the graph. Thus, we add a new edge from the reference node `r_1` to P_2'. This edge describes the fact that the old citation is now "re-assigned" to the new node P_2'. A timestamp on the edges indicates the moment in time in which they were added to the graph. The most recent edge can thus be considered the "valid" one and used, for instance, by algorithms that compute bibliometric measures such as h-index or impact factor.

We note that while one paper may subsume more than one other paper (e.g., a book node with its chapters), it does not make sense for one paper to be subsumed by more than one other CU. Thus, the subsumption relation is many-to-one and acyclic, and it creates a forest within the CG. The roots of the trees of this forest are the ones receiving all the credit, and we call them Primary Citable Units (PCU).

It may not always be desirable or correct for citations to be "inherited" through the subsumption relation, for example when a new version presents a different authors' list. In this case different types of algorithms can be considered to decide how to transfer the citations depending on the nature of the CU.

3.3. Data in the Citation Graph

To deal with data citations in the CG, we use the term "database" in its most general sense: it may be a relational database, an ontology, some form of graph database, or a collection of files [14]. The whole database may be considered a CU, however *parts* of a database may also be CUs. In fact, different parts of a database may present different type of information, that the user may want to cite explicitly. Also, different parts of a database can be curated by different sets of authors (this is particularly true for curated databases such as GtoPdb [15]), making an accurate citation crucial for the correct attribution of credit.

Here, by "part" of a database we mean a *view* [14]. A view is a query which we again generalize to being anything from a relational query for a relational database, a directory path or an URI for a collection of files, or some query in one of the several available query languages. We assume that it is the task of the Data Base Administrator (DBA) to define these views and, consequently, the corresponding CUs.

Let us start, for simplicity, by considering a *static database*, i.e., a database that does not evolve over time. By defining the CUs as views, we immediately obtain a part-of relationship. Given two views V_1 and V_2, we say that V_1 is *part of* V_2 if it can be answered from the result of V_2, i.e., if there is a query Q such that, for each database instance D, $V_1(D) = Q(V_2(D))$.

In this context we observe that the query that defines the view being cited is a fundamental part of the data citation itself. As such, it can be inserted in the reference annotation node of the corresponding citation as metadata. Many approaches can be defined to decide how to *introduce* a new CU corresponding to a view in the CG. We discuss two of them.

One first solution can be to keep the whole database as the only CU and recipient of citations. Every time a paper wants to cite data in the database, it cites that CU, while the reference annotation contains the query used to identify the cited data. With this solution the number of citations to a database may become very high, but, on the other hand, it may happen that the rightful authors and curators of the parts of the database being actually cited do not receive any credit for their work. The queries contained in the reference annotation can be used, though, to accurately compute the correct bibliometric measures, when required.

A second strategy sees a new CU being created every time a paper cites a data subset extracted through a *new* query. With this solution, new views are created every time it is necessary. This, on one hand, may produce an explosion in the number of nodes in the CG, many of which may receive only one citation. On the other hand, it is possible to cite the exact set of data extracted by the query, and to give credit to the rightful authors. If these CUs are connected to the main database through the part-of relationship, the whole database may still inherit these citations, depending on the strategy defined by the DBAs.

Most databases are not static and, unlike documents, they are *expected* to evolve over time. If a new version of a database is released periodically, one option would be to treat each version as a new CU. However, a database may present in the CG a hierarchy of CUs connected among them through the part-of property. While the database may change, some of the single CUs of its hierarchy may not. Moreover, even when one CU changes, it may not change in its entirety, thus we may want to treat it as a new *version*, rather than an entirely unrelated new CU.

To solve these new problems one option is to let the DBA decide. Every time a new version of the database is released, the administrators go through the different CUs that compose the part-of hierarchy and decide which CUs need a new versions. These new versions will be connected to the old one via subsumption if the DBAs deem that the new CU can be considered a new version of the old one, rather than a completely different entity (e.g., when there is some structural change, or the number of curators changed).

3.4. Conclusions

The current basic model of the citation graph is unsuited to the representation of published data and data citations. We propose an implementation-agnostic model that includes reference annotations and a new subsumption property. The annotations contain the context of a citation, such as page numbers or the query issued to obtain the data. This property indicates that one citable unit version “takes the place” of another and inherits its citations if needed. Using these extensions we discussed how to represent data and data citation in the graph, facing problems such the correct attribution of a data citation to authors and the evolution of citable units over time.

Acknowledgments

The work was partially supported by the ExaMode project, as part of the European Union H2020 program under Grant Agreement no. 825292. Matteo Lissandrini is supported by the

European Union H2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 838216. Peter Buneman was partly supported by the Huawei Edinburgh Research Laboratory.

References

[1] P. Buneman, D. Dosso, M. Lissandrini, G. Silvello, Data citation and the citation graph, Quantitative Science Studies 2 (2022) 1399–1422. doi:10.1162/qss_a_00166.

[2] A.-W. K. Harzing, R. Van der Wal, Google scholar as a new source for citation analysis, Ethics in science and environmental politics 8 (2008) 61–73.

[3] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, Z. Su, ArnetMiner: extraction and mining of academic social networks, in: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2008, pp. 990–998.

[4] L. Candela, D. Castelli, P. Manghi, A. Tani, Data Journals: A Survey, Journal of the Association for Information Science and Technology 66 (2015) 1747–1762. URL: http://dx.doi.org/10.1002/asi.23358. doi:10.1002/asi.23358.

[5] Out of Cite, Out of Mind: The Current State of Practice, Polocy, and Technology for the Citation of Data, volume 12, CODATA-ICSTI Task Group on Data Citation Standards and Practices, 2013.

[6] FORCE-11, Data Citation Synthesis Group: Joint Declaration of Data Citation Principles, FORCE11, San Diego, CA, USA, 2014.

[7] C. W. Belter, Measuring the Value of Research Data: A Citation Analysis of Oceanographic Data Sets, PLoS ONE 9 (2014) e92590.

[8] I. Peters, P. Kraker, E. Lex, C. Gumpenberger, J. Gorraiz, Research data explored: An extended analysis of citations and altmetrics, Scientometrics 107 (2016) 723–744.

[9] C. Wilke, What constitutes a citable scientific work?, 2015. https://serialmentor.com/blog/2015/1/2/what-constitutes-a-citable-scientific-work.

[10] M. Altman, M. Crosas, The evolution of data citation: from principles to implementation, IAssist quarterly 37 (2014) 62–62.

[11] M. Daquino, S. Peroni, D. Shotton, G. Colavizza, B. Ghavimi, A. Lauscher, P. Mayr, M. Romanello, P. Zumstein, The OpenCitations data model, in: International Semantic Web Conference, Springer, 2020, pp. 447–463.

[12] F. Osareh, Bibliometrics, citation analysis and co-citation analysis: A review of literature i, Libri 46 (1996) 149–158.

[13] P. Buneman, G. Christie, J. A. Davies, R. Dimitrellou, S. D. Harding, A. J. Pawson, J. L. Sharman, Y. Wu, Why data citation isn't working, and what to do about it, Database (2020). doi:10.1093/databa/baaa022.

[14] P. Buneman, S. Davidson, J. Frew, Why data citation is a computational problem, Communications of the ACM 59 (2016) 50–57.

[15] C. Southan, J. L. Sharman, H. E. Benson, E. Faccenda, A. J. Pawson, S. P. Alexander, O. P. Buneman, A. P. Davenport, J. C. McGrath, J. A. Peters, et al., The IUPHAR/BPS Guide to PHARMACOLOGY in 2016: towards curated quantitative interactions between 1300 protein targets and 6000 ligands, Nucleic acids research 44 (2015) D1054–D1068.