Journal of Informetrics 11 (2017) 176–197

Contents lists available at ScienceDirect

Journal of Informetrics

journal homepage: www.elsevier.com/locate/joi

Regular article

Introducing *metaknowledge*: Software for computational research in information science, network analysis, and science of science[☆]

John McLevey[a,*], Reid McIlroy-Young[b]

[a] University of Waterloo, Canada
[b] University of Chicago, United States

ARTICLE INFO

Article history:
Received 20 July 2016
Received in revised form
14 December 2016
Accepted 14 December 2016
Available online 29 December 2016

Keywords:
Informetrics
Scientometrics
Bibliometrics
Networks
Computational
Big data
Software
RPYS
Gender
Topic models
Burst analysis
Python

ABSTRACT

metaknowledge is a full-featured Python package for computational research in information science, network analysis, and science of science. It is optimized to scale efficiently for analyzing very large datasets, and is designed to integrate well with reproducible and open research workflows. It currently accepts raw data from the Web of Science, Scopus, PubMed, ProQuest Dissertations and Theses, and select funding agencies. It processes these raw data inputs and outputs a variety of datasets for quantitative analysis, including time series methods, Standard and Multi Reference Publication Year Spectroscopy, computational text analysis (e.g. topic modeling, burst analysis), and network analysis (including multi-mode, multi-level, and longitudinal networks). This article motivates the use of *metaknowledge* and explains its design and core functionality.

1. Introduction

Researchers in information science, network analysis, and science of science currently have access to an unprecedented volume of data. Researchers are increasingly working with datasets that include millions of observations (e.g. Börner, 2010, 2015; Boyack, Klavans, & Börner, 2005; Evans & Foster, 2011; Foster, Rzhetsky, & Evans, 2015; Rzhetsky, Foster, Foster, & Evans, 2015; Shi, Foster, & Evans, 2015; Sinatra, Deville, Szell, Wang, & Barabási, 2015; Skupin, Biberstine, & Börner, 2013; Sugimoto, Lariviere, Ni, Gingras, & Cronin, 2013; Uzzi, Mukherjee, Stringer, & Jones, 2013; Wang, Song, & Barabási, 2013).

[☆] Thanks to Jillian Anderson, Steven McColl, Alexander Graham, Amelia Howard, and Pierson Browne for comments on an earlier version of this manuscript. Jillian Anderson developed the Javascript library *mkD3* that enables interactive visualizations of *metaknowledge* datasets. She is a *metaknowledge* developer on releases >3.1. *metaknowledge* is funded in part by a Social Science and Humanities Research Council of Canada grant and an Early Researcher Award from the Ministry of Research and Innovation in Ontario, both awarded to Dr. John McLevey.

[*] Corresponding author.
E-mail address: john.mclevey@uwaterloo.ca (J. McLevey).

http://dx.doi.org/10.1016/j.joi.2016.12.005

In 2015, there were more than 3.8 million records indexed in ProQuest Dissertations and Theses, more than 23 million in PubMed, more than 60 million in Scopus, and the number of cited references indexed in the Web of Science surpassed 1 billion. The Scholarly Database – hosted by researchers at Indiana University – currently contains over 25 million records (LaRowe, Ambre, Burgoon, Ke, & Börner, 2009). The text and network datasets that can be extracted from these databases are often enormous. As de Solla Price (1963) predicted, we are in a period of abundant data, and more is being produced all the time.

In addition to being "bigger" than they used to be, bibliometric datasets are becoming more complex as researchers link them with data from online repositories, social media, blogs, surveys, and administrative data from institutions, granting agencies, and governments (e.g. Cronin & Sugimoto, 2014; Haustein, Peters, Sugimoto, Thelwall, & Larivière, 2014; Kronegger, Mali, Ferligoj, & Doreian, 2012; Sugimoto et al., 2013). Making the most of this abundant data requires access to sufficient infrastructure and software that scales efficiently, reduces opportunities for human error, and is compatible with open and reproducible workflows. Using these tools appropriately requires computing skills that have not traditionally been necessary for conducting sophisticated research on the structure, evolution, and content of science.

There are currently many excellent software options for constructing and analyzing bibliometric datasets, small or large. There is specialized software for historical bibliometrics (e.g. Garfield's (2009) HistCite, Van Eck and Waltmen's (2014) CiteNetExplorer, Thor, Marx, Leydesdorff, and Bornmann's (2016) CRExplorer, and Comins and Leydesdorff's (2016b) RPYS i/o) and for mapping the topic and network structures of science (e.g. Van Eck and Waltmen's (2010) VOSViewer, Chen's (2006) CiteSpace, and WoS2Pajek for Pajek (De Nooy, Mrvar, & Batagelj, 2011)). Katy Börner and her collaborators developed Sci^2 and the Network Workbench (NWB) as modular "plug and play" programs, intended to be collaboratively developed by scientometric researchers as the field evolves (Börner, 2011). All of these programs have their own parsers for converting raw data files into something useful for bibliometric and scientometric research. Most tend to focus on very specific research ends (e.g. creating topic maps) and attempt to cover an entire research workflow from parsing raw data to producing graphs intended for publication. They are all primarily graphical user interfaces (GUIs) with drop down menus that require repetitive user input.[1]

GUI systems dramatically lower the barriers to conducting bibliometric and scientometric research, but many of the most exciting and promising developments in the field require computing workflows that are better suited to scripted data analysis, for example in R, Python, or Stata. Almost all research workflows include many small sequential tasks, some of which have to be repeated many times. A GUI program can require hours of tedious and error prone user input every time the workflow is executed. This is a waste of researcher time and effort. It could be automated and made reproducible with data cleaning and analysis scripts. While we fully support efforts to empower as many researchers as possible to leverage access to data and computing power to advance research in information science, network analysis, and science of science, there is a trade-off. GUI software plays a central role in research, but we also require software that is optimized for scalability, speed, reproducibility, easily linking open data, and open workflows.[2]

This article introduces *metaknowledge*, a Python package for computational research in information science, network analysis, and science of science.[3] The package name is adopted from Evans and Foster's (2011) brief article in *Science*. In short, it accepts raw data inputs from the Web of Science, PubMed, Scopus, Proquest Dissertation and Theses, and administrative data from some funding agencies. It outputs tidy datasets for a wide range of quantitative analyses, including but not limited to longitudinal analysis, Standard and Multi Reference Publication Year Spectroscopy (RPYS), computational text analysis (e.g. topic modeling, burst analysis), and network analysis (including multi-mode, multi-level, and longitudinal networks). Although *metaknowledge* is aimed at researchers with some programming knowledge, who are working with large and complex bibliometric datasets and / or who are committed to open and reproducible research, it fits into any research workflow in bibliometrics and scientometrics. In the sections below, we discuss the design and core functionality of *metaknowledge*, explain how to get started, and demonstrate some of its most useful functions.

2. Design and general overview

metaknowledge was designed with open and reproducible research workflows in mind. First, it is open source (General Public License 2). All source code is easily available online, enabling other researchers to make modifications that are useful in their own work, such as by adding custom parsers to process administrative data from institutions in their own country. Second, as a Python package, *metaknowledge* is scriptable, meaning researchers write small amounts of code to process and analyze their data. These scripts can be re-run anytime, and all revisions can be tracked using version control systems such as *git* and hosted on online platforms such as Github, GitLab, or the Open Science Framework. Analyses can be automated using clearly documented dependencies between files, for example by using Makefiles

[1] One exception is Gagolewski's (2011) CITAN package for R, which is primarily focused on impact assessment, e.g. computing *h* index and *g* index.

[2] The availability of sophisticated libraries in R (e.g. statnet suite and igraph) and packages in Python (e.g. networkx), for example, has been enormously productive for social networks researchers despite the fact that GUIs like UCINet, Pajek, Visone, and Gephi are widely used.

[3] We chose to make *metaknowledge* a Python package because Python excels at cleaning and manipulating strings and is well-suited for intensive research computing.