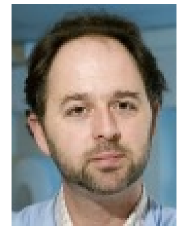
Fork, merge and crowd-sourcing data curation: tools for collective data processing and analysis.

blogs.lse.ac.uk/impactofsocialsciences/2014/05/05/fork-merge-crowdsourcing-data-curation/

5/5/2014



With the right formats, licensing and distribution mechanisms, people can easily collaborate over data, enhance the analysis and re-purpose for their own needs. **Cameron Neylon** *reflects on the tools available for these aims. The interfaces that make working with the data easy may create barriers to automation and computational processing down the line. Further mechanisms are needed, both social and technical, to make it easy to contribute variations, enhancements and new ideas back to the original resources.*

Over the past few weeks there has been a sudden increase in the amount of financial data on scholarly communications in the public domain. This was triggered in large part by the Wellcome Trust releasing data on the prices paid for Article Processing Charges by the institutions it funds. The release of this pretty messy dataset was followed by a substantial effort to clean that data up. This crowd-sourced data curation process has been described by Michelle Brook. Here I want to reflect on the tools that were available to us and how they made some aspects of this collective data curation easy, but also made some other aspects quite hard.

The data started its life as a csv file on Figshare. This is a very frequent starting point. I pulled that dataset and did some cleanup using OpenRefine, a tool I highly recommend as a starting point for any moderate to large dataset, particularly one that has been put together manually. I could use OpenRefine to quickly identify and correct variant publisher and journal name spellings, clean up some of the entries, and also find issues that looked like mistakes. It's a great tool for doing that initial cleanup, but its a tool for a single user, so once I'd done that work I pushed my cleaned up csv file to github so that others could work with it.

After pushing to github a number of people did exactly what I'd intended and *forked* the dataset. That is, they took a copy and added it to their own repository. In the case of code people will fork a repository, add to or improve the code, and then make a *pull request* that gives the original repository owner that there is new code that they might want to *merge* into their version of the codebase. The success of github has been built on making this process easy, even fun. For data the merge process can get a bit messy but the potential was there for others to do some work and for us to be able to combine it back together.

Image credit: Phillipe Put (Flickr, CC BY)

But github is really only used by people comfortable with command line tools – my thinking was that people would use computational tools to enhance the data. But Theo Andrews had the idea to bring in many more people to manually look at and add to the data. Here an online spreadsheet such as those provided by GoogleDocs that many people can work with is a powerful tool and it was through that adoption of the GDoc that somewhere over 50 people were able to add to the spreadsheet and annotate it to create a high value dataset that allowed the Wellcome Trust to do a much deeper analysis than had previously been the case. The dataset had been forked again, now to a new platform, and this tool enabled what you might call a "social merge" collecting the individual efforts of many people through an easy to use tool.

The interesting thing was that exactly the facilities that made the GDoc attractive for manual crowdsourcing efforts made it very difficult for those of us working with automated tools to contribute effectively. We could take the data and manipulate it, forking again, but if we then pushed that re-worked data back we ran the risk of overwriting what anyone else had done in the meantime. That live online multi-person interaction that works well for people, was actually a problem for computational processing. The interface that makes working with the data easy for people actually created a barrier to automation and a barrier to merging back what others of us were trying to do. [As an aside, yes we could in principle work through the GDocs API but that's just not the way most of us work doing this kind of data processing].

Crowdsourcing of data collection and curation tends to follow one of two paths. Collection of data is usually done into some form of structured data store, supported by a form that helps the contributor provide the right kind of structure. Tools like EpiCollect provide a means of rapidly building these kinds of projects. At the other end large scale data curation efforts, such as GalaxyZoo, tend to create purpose built interfaces to guide the users through the curation process, again creating structured data. Where there has been less tool building and less big successes are the space in the middle, where messy or incomplete data has been collected and a community wants to enhance it and clean it up. OpenRefine is a great tool, but isn't collaborative. GDocs is a great collaborative platform but creates barriers to using automated cleanup tools. Github and code repositories are great for supporting the fork, work, and merge back patterns but don't support direct human interaction with the data.

These issues are part of a broader pattern of issues with the Open Access, Data, and Educational Resources more generally. With the right formats, licensing and distribution mechanisms we've become very very good at supporting the fork part of the cycle. People can easily take that content and re-purpose it for their own local needs. What we're not so good at is providing the mechanisms, both social and technical, to make it easy to contribute those variations, enhancements and new ideas back to the original resources. This is both a harder technical problem and challenging from a social perspective. Giving stuff away, letting people use it is easy because it requires little additional work. Working with people to accept their contributions back in takes time and effort, both often in short supply.

The challenge may be even greater because the means for making one type of contribution easier may make others harder. That certainly felt like the case here. But if we are to reap the benefits of open approaches then we need to do more than just throw things over the fence. We need to find the ways to gather back and integrate all the value that downstream users can add.

This post originally appeared on Cameron Neylon's blog Science in the Open and is reposted under CC-0.

Note: This article gives the views of the authors, and not the position of the Impact of Social Science blog, nor of the London School of Economics. Please review our Comments Policy if you have any concerns on posting a comment below.

About the Author

Cameron Neylon *is a biophysicist and well known advocate of opening up the process of research. He is Advocacy Director for PLOS and speaks regularly on issues of Open Science including Open Access publication, Open Data, and Open Source as well as the wider technical and social issues of applying the opportunities the internet brings to the practice of science. He was named as a SPARC Innovator in July 2010 and is a proud recipient of the Blue Obelisk for contributions to open data. He writes regularly at his blog, Science in the Open.*