Meta Review

This manuscript was automatically generated from greenelab/meta-review@e33d980 on March 14, 2018. The permalink for this manuscript version is https://greenelab.github.io/meta-review/v/e33d980b0f582e1dcd57be0a687d8a13f3348d2d/.

Authors

- **Daniel S. Himmelstein**
  0000-0002-3012-7446 · dhimmel · dhimmel
  Department of Systems Pharmacology and Translational Therapeutics, University of Pennsylvania · Funded by GBMF4552

- **Casey S. Greene**
  0000-0001-8713-9213 · cgreene · GreeneScientist
  Department of Systems Pharmacology and Translational Therapeutics, University of Pennsylvania · Funded by GBMF4552

- **Anthony Gitter**
  0000-0002-5324-9833 · agitter · anthonygitter
  Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison; Morgridge Institute for Research · Funded by NIH U54AI117924

Abstract

Scientific literature reviews are ideal for open, collaborative writing. Allowing any interested individual to contribute as an author can strengthen a review, providing broad and fair coverage of the subject matter. However, the traditional multi-author writing process breaks down at scale. We present techniques for overcoming the challenges of open manuscript writing. These include approaches for managing distributed authors and our new software for automating citation and manuscript building.

Introduction

Openness in research – which includes sharing code, data, and manuscripts – benefits the researchers who practice it [1], their scientific peers, and the public. `TODO: more references needed` However, though review articles aim to present the state of the art in a scientific field, they are written in private by a single research group or a small team of colleagues. In contrast, broadly opening the process to anyone engaged in the topic — such that planning, organizing, writing, and editing occur collaboratively in a public forum where anyone is welcome to participate — may maximize a review's value. Open drafting of reviews may be especially helpful for capturing state-of-the-art knowledge about rapidly advancing research topics at the intersection of existing disciplines where contributors bring diverse opinions and expertise.

In August 2016 we identified the role that deep learning was beginning to play in biomedical research as one such area, and we started an open online effort to survey deep learning's role in precision medicine and to predict its effect in the future. In May 2017, the project released a complete review titled "Opportunities and obstacles for deep learning in biology and medicine" [2]. While the article was under review, we continued to maintain the project and accepted new contributions. In February 2018, the article was accepted by the journal that it was initially submitted to. We'll discuss our experience with our open review including the pros and cons of open collaborative reviews, as well as the infrastructure we created, termed Manubot, to enable open manuscript writing online. We also discuss the extent to which such efforts can contribute to a living and frequently updated literature. `TODO: discuss stage 2 maintainers`

We initiated our open online review article by creating a GitHub repository (https://github.com/greenelab/deep-review) to coordinate and manage contributions. GitHub is a platform designed for collaborative software development that's adaptable for collaborative writing. From the start, we made the GitHub repository public, applying a Creative Commons Attribution License to the manuscript. Next, we encouraged anyone interested to contribute by proposing changes or additions. Although we requested that some authors participate for their specific expertise, most discovered the manuscript organically through conferences or social media, deciding without solicitation to contribute. In total, the project that we termed the "Deep Review" attracted 36 authors from 20 different institutions (and counting) who were not determined in advance.

Writing review articles in a public forum allows review authors to engage with the original researchers to clarify their methods and results and present them accurately, as exemplified here. `TODO: need archival issue link` In addition, discussing manuscripts in the open provides one form of pre- and post-publication peer review `TODO: define this or provide a reference?`, incentivizing the reviews with potential manuscript authorship. However, inviting wide authorship brings many technical and social challenges such as how to fairly distribute credit, coordinate the scientific content, and collaboratively manage extensive reference lists.

To coordinate this effort, we developed a manuscript writing process using the Markdown language, GitHub, and our new Manubot tool for automating manuscript generation.

Contribution workflow

There are many existing collaborative writing platforms ranging from rich text editors, which support Microsoft Word documents or similar formats, to LaTeX-based systems for technical writing [3] such as Overleaf and Authorea. These platforms ideally offer version control, multiple permission levels, or other functionality to support multi-author document editing. Although they work well for editing, they lack sufficient features for managing a collaborative manuscript and attributing precise credit, which are important for open writing.

We adopted standard software development strategies in order to enable any contributor to edit any part of the manuscript but enforce discussion and review of all proposed changes. The GitHub platform provided support for organizing and editing the manuscript. We used GitHub *issues* for organization, opening a new issue for each paper under consideration. Within the issue, contributors summarized the research, discussed it (sometimes with the original authors), and assessed its relevance to the review. Issues also served as an open to-do list and a forum for debating the main message, themes, and topics of the review.

GitHub and the underlying git version control system [4,5] also structured the writing process. The official version of the manuscript is *forked* by individual contributors. A contributor then adds and revises files, grouping these changes into *commits*. When the changes are ready to be reviewed, the series of commits are submitted as a *pull request* through GitHub, which notifies other authors of the pending changes. GitHub's review interface allows anyone to comment on the changes, globally or at specific lines, asking questions or requesting modifications as depicted in [6]. Conversations during review can reference other pull requests, issues, or authors, linking the relevant people and content. Reviewing batches of revisions that focus on a single theme is more efficient than independently discussing isolated comments and edits, and it helps maintain consistent content and tone across different authors and reviewers. Once all requested modifications are made, the manuscript maintainers, a subset of authors with elevated GitHub permissions, formally approve the pull request and merge the changes into the official version. `TODO: need a figure with a flowchart showing this process` The process of writing and revising material can be orchestrated through GitHub with a web browser.

We found that this workflow was an effective compromise between fully unrestricted editing and a more heavily-structured approach that limited the authors or the sections they could edit. In addition, authors are associated with their commits, which makes it easy for contributors to receive credit for their work and helps prevent ghostwriting [7]. Figure 1 and the GitHub contributors page summarize all edits and commits from each author, providing aggregated information that is not available on other collaborative writing platforms. Because our writing process, like others backed

by the open git version control system (including Overleaf and Authorea), tracks the complete commit history, it also enables detailed retrospective contribution analysis.

```
TODO: confirm Overleaf and Authorea provide this type of git integration versus something more coar
```

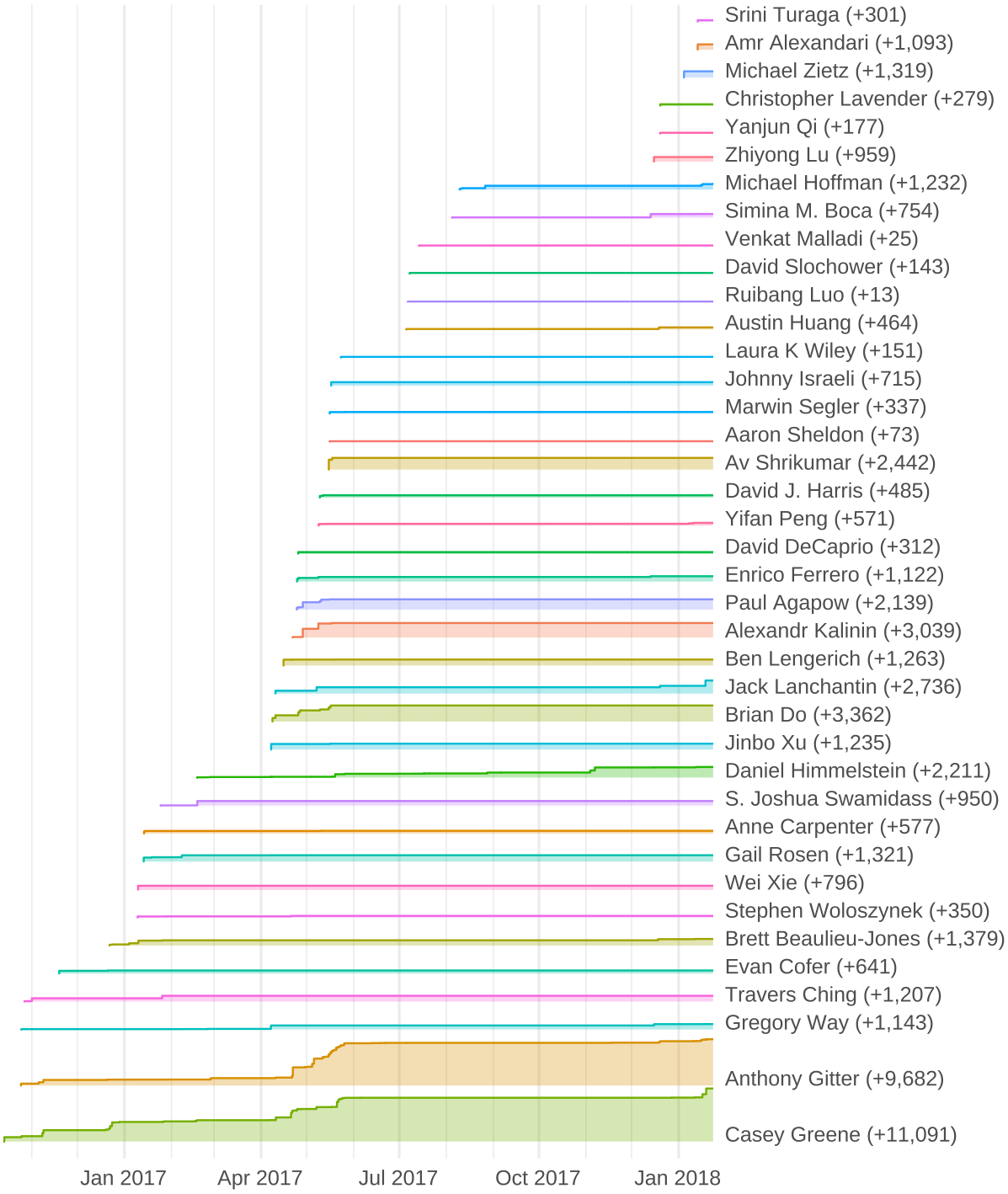

Figure 1: **Deep Review contributions by author over time.** The total words added to the Deep Review by each author is plotted over time (final values in parentheses). These statistics were extracted from git commit diffs of the manuscript's markdown source. This figure reveals the composition of written contributions to the manuscript at every point in its history.

Manubot

We developed Manubot, a system for writing scholarly manuscripts via GitHub. With Manubot, manuscripts are written as plain-text markdown files, which is well suited for version control using git. The markdown standard itself provides limited yet crucial formatting syntax, including the ability to embed images and format text via bold, italics, hyperlinks, headers, inline code, codeblocks, blockquotes, and numbered or bulleted lists. In addition, Manubot relies on extensions from Pandoc markdown to enable citations, tables, captions, and equations specified using the popular TeX math syntax.

Manubot includes an additional layer of citation processing, currently unique to the system. All citations point to a standard identifier, for which Manubot automatically retrieves bibliographic metadata. Currently, citations to DOIs (Digital Object Identifiers), PubMed or PubMed Central identifiers, arXiv identifiers, and URLs (web addresses) are supported. Metadata is retrieved using DOI Content Negotiation, NCBI's E-utilities and Citation Exporter, the arXiv API, and Greycite [8]. Metadata is exported to CSL JSON Items, an open standard that's widely supported by reference managers [9,10]. In cases where automatic retrieval of metadata fails or produces incorrect references — which is most common for URL citations — users can manually provide the correct CSL JSON.

The Manubot formats bibliographies according to a Citation Style Language (CSL) specification. As a result, users can choose from thousands of existing CSL styles or use Manubot's default style. Styles define how references are constructed from bibliographic metadata, controlling layout details such as the max number of authors to list per reference. Thousands of journals have predefined styles. As a result, adopting the specific bibliographic format required by a journal usually just requires specifying the style's source URL in the Manubot configuration.

Manubot uses Pandoc to convert manuscripts from markdown to HTML, PDF, and optionally DOCX outputs. Pandoc supports conversion between additional formats — such as LaTeX, AsciiDoc, EPUB, and JATS — offering Manubot users broad interoperability. Journal Article Tag Suite (JATS) is a standard XML format for scholarly articles that is used by publishers, archives, and text miners [11–13]. Going forward, we hope to integrate Manubot with the larger JATS ecosystem. For now however, the primary Manubot output is HTML intended to be viewed in a web browser.

Manubot performs continuous publication: every update to a manuscript's source is automatically reflected in the online outputs. The approach uses continuous integration (CI) [14,15], specifically Travis CI at the moment, to monitor for changes. When changes occur, the CI service generates an updated manuscript. If this process is error free, the CI service timestamps the manuscript and uploads the output files to the GitHub repository. Since the HTML manuscript is hosted using

GitHub Pages, the CI service automatically deploys the new manuscript version when it pushes the updated outputs to GitHub.

For this article, the source GitHub repository is https://github.com/greenelab/meta-review. When this repository changes, Travis CI rebuilds the manuscript. If successful, the output is deployed back to GitHub (to dedicated `output` and `gh-pages` branches). As a result, https://greenelab.github.io/meta-review stays up to date with the latest HTML manuscript.

The idea of the "priority of discovery" is important to science, and Vale and Hyman discuss the importance of both disclosure and validation [16]. In their framework, disclosure occurs when a scientific output is released to the world. However, for a manuscript that is shared as it is written, being able to establish priority could be challenging. We implemented support for OpenTimestamps in Manubot to timestamp the HTML and PDF outputs on the Bitcoin blockchain. This procedure allows one to retrospectively prove that a manuscript version existed prior to its blockchain-verifiable timestamp [17–20]. For Manubot manuscripts, scientific precedence can now be indisputably established, and timestamps protect against attempts to rewrite a manuscript's history. Such timestamping practices help ensure accurate histories, potentially alleviating certain authorship or priority disputes. Since all bitcoin transactions are competing for limited space on the blockchain, the fees required to send a single transaction can be high. OpenTimestamps avoids this fee by encoding many timestamps into a single Bitcoin transaction [21], but there can be a lag of a few hours before the transaction is made. We judged this to be suitable for the purposes of scientific writing.

We designed Manubot to power the next generation of scholarly manuscript. Manubot transforms publication, making it permissionless, reproducible, free of charge, and largely open source. Manubot does rely on gratis services from two proprietary platforms: GitHub and Travis CI. Fortunately, lock-in to these services is minimal, and several substitutes already exist. One direction Manubot is working towards is end-to-end document reproducibility, where every figure or piece of data in a manuscript can be traced back to its origin [22]. Already, Manubot is well suited for preserving provenance. For example, figures can be specified using versioned URLs that refer to the code that created them. In addition, manuscripts can be templated, so that numerical values or tables get inserted directly from the repository that created them. An example repository demonstrates Manubot's features and serves as a template for users to write their own manuscript with Manubot.

Authorship

To determine authorship we followed the International Committee of Medical Journal Editors (ICMJE) guidelines and used GitHub to track contributions. ICMJE recommends authors substantially contribute to, draft, approve, and agree to be accountable for the manuscript. We acknowledged other contributors who did not meet all four criteria, including contributors who provided text but did not review and approve the complete manuscript. Although these criteria

provided a straightforward, equitable way to determine who would be an author, they did not produce a traditionally ordered author list. In biomedical journals, the convention is that the first and last authors made the most substantial contributions to the manuscript. This convention can be difficult to reconcile in a collaborative effort. Using git, we could quantify the number of commits each author made or the number of sentences an author wrote or edited, but these metrics discount intellectual contributions such as discussing primary literature and reviewing pull requests. However, there is no objective system to compare and weight the different types of contributions and produce an ordered author list.

To address this issue, we generalized the concept of “co-first” authorship, in which two or more authors are denoted as making equal contributions to a paper. We defined four types of contributions [2], from major to minor, and reviewed the GitHub discussions and commits to assign authors to these categories. A randomized algorithm then arbitrarily ordered authors within each contribution category, and we combined the category-specific author lists to produce a traditional ordering. The randomization procedure was shared with the authors in advance (pre-registered) and run in a deterministic manner. Given the same author contributions, it always produced the same ordered author list. We annotated the author list to indicate that author order was partly randomized and emphasize that the order did not indicate one author contributed more than another from the same category.

`TODO: In Discussion, present alternative author ordering strategies and literature on contribution`

Discussion

Many others have embraced open science principles and piloted open approaches toward drug discovery [23,24], data management [25–27], and manuscript review [28].

`TODO: need help deciding what related topics to include here and which references to use, these are`

`TODO: more ideas in doi:10.7287/peerj.preprints.2711v2` Several of these open science efforts are GitHub-based like our collaborative writing process. The ReScience [29], the Journal of Open Source Software [30], and some other Open Journals rely on GitHub for peer review and hosting. GitHub is also increasingly used for resource curation [31], and collaborative scholarly reviews combine literature curation with discussion and interpretation.

`TODO: describe Manubot related work here?` [32] and https://github.com/ewanmellor/gh-publisher

There are potential limitations of our GitHub-based approach. Because our review manuscript pertained to a computational topic, most of the authors had computational backgrounds, including previous experience with version control workflows and GitHub. In other disciplines, collaborative writing via GitHub and Manubot could present a steeper barrier to entry and deter participants. In addition, git carefully tracks all revisions to the manuscript text but not the surrounding conversations that take place through GitHub issues and pull requests. These discussions must be archived to ensure that important decisions about the manuscript are preserved and authors receive credit for intellectual contributions that are not directly reflected in the manuscript’s text.

GitHub supports programmatic access to issues, pull requests, and reviews so tracking these conversations is feasible in the future.

In our open review paper, we established contributor guidelines that discussed norms in the areas of of text contribution, peer review, and authorship, which we identified in advance as potential areas of disagreement. Our contributor guidelines required verifiable participation: either directly attributable changes to the text or participation in the discussion on GitHub. We maintained our guidelines, even when a case arose where two authors had written text together but only one had directly attributable changes and participation. These guidelines did not discuss broader community norms that may have improved inclusiveness. It is also important to consider how the move to an open contribution model affects under-represented minority members of the scientific community [33]. Recent work has identified clear social norms and processes as helpful to maintaining a collaborative culture [34]. Conferences and open source projects have used codes of conduct to establish these norms [35,36]. We would encourage the maintainers of similar projects to consider broader codes of conduct for project participants that establish on social as well as academic norms.

Open writing presents new opportunities for scholarly communication.
`TODO: reference "paper of the future"? arXiv:1601.02927 doi:10.22541/au.149693987.70506124 doi:10.2`
[2], 12 new contributors have updated the manuscript (Figure 1) and existing authors continue to discuss new literature, creating a living document.
`TODO: update new author count before submitting` The Manubot system can also facilitate open research [37] in addition to review articles.
`TODO: get permission and add https://slochower.github.io/nonequilibrium-barrier/ https://zietzm.git`

Our process represents an early step toward open massively collaborative reviews, and there are certainly aspects that can be improved. We invite the scientific community to adapt and build upon our experience and open software.

Acknowledgements

`TODO: deep review authors for support in testing this process`

`TODO: manubot-rootstock contributors`

References

1. **How open science helps researchers succeed**
Erin C McKiernan, Philip E Bourne, C Titus Brown, Stuart Buck, Amye Kenall, Jennifer Lin, Damon McDougall, Brian A Nosek, Karthik Ram, Courtney K Soderberg, … Tal Yarkoni
eLife (2016-07-07) https://doi.org/10.7554/elife.16800

2. **Opportunities And Obstacles For Deep Learning In Biology And Medicine**
Travers Ching, Daniel S. Himmelstein, Brett K. Beaulieu-Jones, Alexandr A. Kalinin, Brian T. Do, Gregory P. Way, Enrico Ferrero, Paul-Michael Agapow, Michael Zietz, Michael M Hoffman, … Casey S. Greene
Cold Spring Harbor Laboratory (2017-05-28) https://doi.org/10.1101/142760

3. **Scientific writing: the online cooperative**
Jeffrey M. Perkel
Nature (2014-10-01) https://doi.org/10.1038/514127a

4. **A Quick Introduction to Version Control with Git and GitHub**
John D. Blischak, Emily R. Davenport, Greg Wilson
PLOS Computational Biology (2016-01-19) https://doi.org/10.1371/journal.pcbi.1004668

5. **Ten Simple Rules for Taking Advantage of Git and GitHub**
Yasset Perez-Riverol, Laurent Gatto, Rui Wang, Timo Sachsenberg, Julian Uszkoreit, Felipe da Veiga Leprevost, Christian Fufezan, Tobias Ternent, Stephen J. Eglen, Daniel S. Katz, … Juan Antonio Vizcaíno
PLOS Computational Biology (2016-07-14) https://doi.org/10.1371/journal.pcbi.1004947

6. **Opportunities And Obstacles For Deep Learning In Biology And Medicine**
Johnny Israeli
Medium (2017-05-31) https://medium.com/towards-data-science/opportunities-and-obstacles-for-deep-learning-in-biology-and-medicine-6ec914fe18c2

7. **What Should Be Done To Tackle Ghostwriting in the Medical Literature?**
Peter C Gøtzsche, Jerome P Kassirer, Karen L Woolley, Elizabeth Wager, Adam Jacobs, Art Gertel, Cindy Hamilton
PLoS Medicine (2009-02-03) https://doi.org/10.1371/journal.pmed.1000023

8. **Twenty-Five Shades of Greycite: Semantics for referencing and preservation**
Phillip Lord, Lindsay Marshall
arXiv (2013-04-26) https://arxiv.org/abs/1304.7151v1

9. **Reference Management**
Martin Fenner, Kaja Scheliga, Sönke Bartling
Opening Science (2013-12-17) https://doi.org/10.1007/978-3-319-00026-8_8

10. **Comparison of Select Reference Management Tools**
Yingting Zhang
Medical Reference Services Quarterly (2012-01) https://doi.org/10.1080/02763869.2012.641841

11. **JATS: Journal Article Tag Suite, version 1.1**
National Information Standards Organization
(2015) http://www.niso.org/standards/z39-96-2015/

12. **Journal Article Tag Suite 1.0: National Information Standards Organization standard of journal extensible markup language**
Sun Huh
Science Editing (2014-08-18) https://doi.org/10.6087/kcse.2014.1.99

13. **NISO Z39.96-201x, JATS: Journal Article Tag Suite**
Mark H. Needleman
Serials Review (2012-09) https://doi.org/10.1080/00987913.2012.10765464

14. **Collaborative software development made easy**
Andrew Silver
Nature (2017-10-04) https://doi.org/10.1038/550143a

15. **Reproducibility of computational workflows is automated using continuous analysis**
Brett K Beaulieu-Jones, Casey S Greene
Nature Biotechnology (2017-03-13) https://doi.org/10.1038/nbt.3780

16. **Priority of discovery in the life sciences**
Ronald D Vale, Anthony A Hyman
eLife (2016-06-16) https://doi.org/10.7554/elife.16931

17. **Decentralized Trusted Timestamping using the Crypto Currency Bitcoin**
Bela Gipp, Norman Meuschke, André Gernandt
arXiv (2015-02-13) https://arxiv.org/abs/1502.04015v1

18. **The Grey Literature — Proof of prespecified endpoints in medical research with the bitcoin blockchain**
Benjamin Gregory Carlisle
(2014-08-25) https://www.bgcarlisle.com/blog/2014/08/25/proof-of-prespecified-endpoints-in-medical-research-with-the-bitcoin-blockchain/

19. **The most interesting case of scientific irreproducibility?**
Daniel Himmelstein
Satoshi Village (2017-03-08) http://blog.dhimmel.com/irreproducible-timestamps/

20. **Bitcoin: A Peer-to-Peer Electronic Cash System**
Satoshi Nakamoto
(2017-09-20) http://git.dhimmel.com/bitcoin-whitepaper/

21. **OpenTimestamps: Scalable, Trustless, Distributed Timestamping with Bitcoin***Peter Todd*
(2018-02-10) https://petertodd.org/2016/opentimestamps-announcement

22. **eLife supports development of open technology stack for publishing reproducible manuscripts online**(2017-09-07) https://elifesciences.org/for-the-press/e6038800/elife-supports-development-of-open-technology-stack-for-publishing-reproducible-manuscripts-online

23. **Systematic integration of biomedical knowledge prioritizes drugs for repurposing**
Daniel Scott Himmelstein, Antoine Lizee, Christine Hessler, Leo Brueggeman, Sabrina L Chen, Dexter Hadley, Ari Green, Pouya Khankhanian, Sergio E Baranzini
eLife (2017-09-22) https://doi.org/10.7554/elife.26726

24. **An open source pharma roadmap**
Manica Balasegaram, Peter Kolb, John McKew, Jaykumar Menon, Piero Olliaro, Tomasz Sablinski, Zakir Thomas, Matthew H. Todd, Els Torreele, John Wilbanks
PLOS Medicine (2017-04-18) https://doi.org/10.1371/journal.pmed.1002276

25. **The Open Knowledge Foundation: Open Data Means Better Science**
Jennifer C. Molloy
PLoS Biology (2011-12-06) https://doi.org/10.1371/journal.pbio.1001195

26. **The FAIR Guiding Principles for scientific data management and stewardship**
Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne, … Barend Mons
Scientific Data (2016-03-15) https://doi.org/10.1038/sdata.2016.18

27. **First, design for data sharing**
John Wilbanks, Stephen H Friend
Nature Biotechnology (2016-04) https://doi.org/10.1038/nbt.3516

28. **A multi-disciplinary perspective on emergent and future innovations in peer review**
Jonathan P. Tennant, Jonathan M. Dugan, Daniel Graziotin, Damien C. Jacques, François Waldner, Daniel Mietchen, Yehia Elkhatib, Lauren B. Collister, Christina K. Pikas, Tom Crick, … Julien

Colomb
F1000Research (2017-07-20) https://doi.org/10.12688/f1000research.12037.1

29. **Sustainable computational science: the ReScience initiative**
Nicolas P. Rougier, Konrad Hinsen, Frédéric Alexandre, Thomas Arildsen, Lorena Barba, Fabien C. Y. Benureau, C. Titus Brown, Pierre de Buyl, Ozan Caglayan, Andrew P. Davison, … Tiziano Zito
arXiv (2017-07-14) https://arxiv.org/abs/1707.04393v2

30. **Journal of Open Source Software (JOSS): design and first-year review**
Arfon M Smith, Kyle E Niemeyer, Daniel S Katz, Lorena A Barba, George Githinji, Melissa Gymrek, Kathryn D Huff, Christopher R Madan, Abigail Cabunoc Mayes, Kevin M Moerman, … Jacob T Vanderplas
arXiv (2017-07-07) https://arxiv.org/abs/1707.02264v2

31. **The appropriation of GitHub for curation**
Yu Wu, Na Wang, Jessica Kropczynski, John M. Carroll
PeerJ Computer Science (2017-10-09) https://doi.org/10.7717/peerj-cs.134

32. **Formatting Open Science: agilely creating multiple document formats for academic manuscripts with Pandoc Scholar**
Albert Krewinkel, Robert Winkler
PeerJ Computer Science (2017-05-08) https://doi.org/10.7717/peerj-cs.112

33. **Open science and open science**
Laurent Gatto
(2017-06-05) https://lgatto.github.io/open-and-open/

34. **Innovating Collaborative Content Creation: The Role of Altruism and Wiki Technology**
Christian Wagner, Pattarawan Prasarnphanich
2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07) (2007) https://doi.org/10.1109/hicss.2007.277

35. **Code of conduct evaluations***Geek Feminism Wiki* (2017-06-13) http://geekfeminism.wikia.com/wiki/Code_of_conduct_evaluations?oldid

36. **Contributor Covenant: A Code of Conduct for Open Source Projects**
Coraline Ada Ehmke
(2014) https://www.contributor-covenant.org/

37. **Sci-Hub provides access to nearly all scholarly literature**
Daniel S Himmelstein, Ariel R Romero, Stephen R McLaughlin, Bastian Greshake Tzovaras, Casey S Greene
PeerJ (2017-07-20) https://doi.org/10.7287/peerj.preprints.3100v1