Microsourcing: A literature review (illustration)

Gerit Wagner Julian Prester Roman Lukyanenko Guy Paré

> **Note**
>
> - Create colrev repository in parallel
> - Add this manuscript vignette at the end
> - Push to C5-DM-vignette
> - Publish as Quarto website
> - Update links in this document
> - Include vignette (screenshot?) in the paper
> - Share with JP

Plan

- The review is conducted using a shared GitHub repository, which was be synchronized locally by the team
- Record metadata is curated as follows:
  - Data retrieved in the search is stored in the data/raw directory; the Git history of this path shows that the files were preserved in their original form, i.e., treated as raw data
  - Records were imported into the data/records.bib as the primary data structure; the Git history of this file documents how each record evolved through the process (e.g., manual or computational change of metadata, merging of records, prescreening decisions)
- For primary data (record metadata), the Bibtex format was chosen and consistent formatting was ensured by CoLRev. BibTex is machine readable and the changes can easily be interpreted when inspecting the git history.

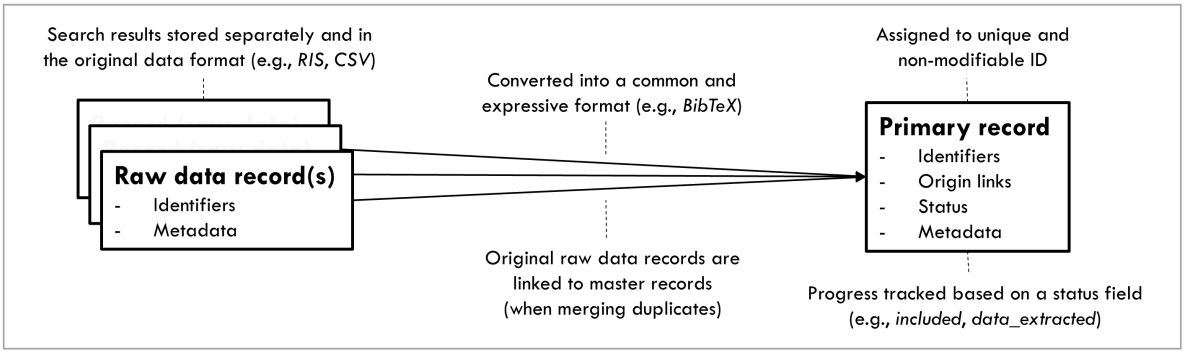


Figure 1: Data structures

Search

- The search strategy is stored together with the raw data files here, in line with the standard of Haddaway et al. (2022).
- Open-access API-searches (licensing issues permit publication of raw data exported from databases like WOS or EBSCO)
- We reused samples from prior reviews (Wagner, Prester, and Paré 2021; Fiers 2023): hrere

> **Note**
>
> - Search-query was used to validate syntactic correctness and ...
> - scope (????)

Dedupe

- Preparation was done using CoLRev and extensions (see prep commit).
- Deduplication was done using BibDedupe. Deduplication changes are in dedupe commit.

> **Note**
>
> Dedupe changes were validated using the max-diff strategy (`colrev validate XXXX`).
> Preparation changes were validated using the max-diff strategy (`colrev validate XXXX`).

Prescreen

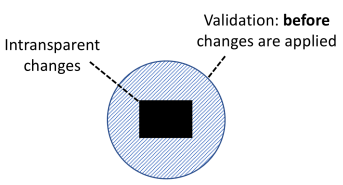
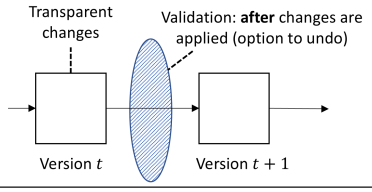
Data management system	Relational database	Collaborative versioning system
Illustration		
Prevalence	High (e.g., ASReview, Buhos, Rayyan, HubMeta, CRUISE-screening)	Low
Nature of changes	Not transparent and hard to undo	Transparent and reversible
Typical approach adopted by researchers	**Protective boundary approach** Authors of literature reviews operate with *black-box data* and categorically exclude computational technqiues (e.g., ML, LLMs), crowd workers, research assistants, and junior scholars due to unknown performance.	**Test-validate-undo approach** Authors operate with *white-box data* and use state-of-the-art computational techniques, more diverse and larger teams because their contributions can be tested, validated, and undone.

Figure 2: Data structures

> **Note**
>
> - For prescreening, we tested the new llm-prescreener in ref. Comparison with prescreening decisions of GW showed low reliability with the llm-prescreener (command + kappa). Results were therefore reverted (ref) and a fully manual prescreen was implemented.
>
> Note: this could also be done in a separate branch, or the changes could be undone using a hard git reset.
>
> - Screen: fulltext documents were shared in a protected drive (link to Dropbox)

Data extraction

- For data extraction, four scenarios were considered:
  - A purely narrative review (the synthesis is written in this document)
  - A bibliometric analysis (the citation network is here)
  - An emergent mapping study (the notes are here and illustrated here)

– A structured extraction of evidence (a preliminary coding scheme is here and the pilot coding here)

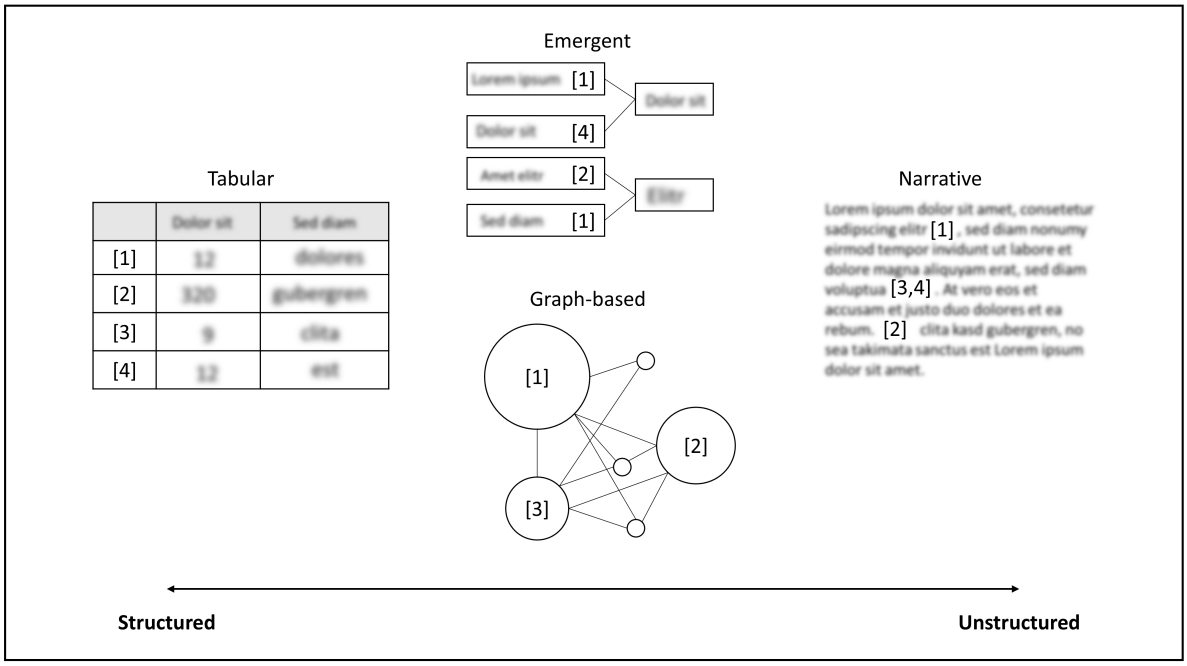


Figure 3: Data structures

Synthesis

- The narrative synthesis is in this document in Markdown format, allowing for larger teams to work on the same document (similar to *cite-covid-review*)
- To make the review reusable, we added the XY license (indexing in SYNERGY, SearchRXiv is planned once the review progresses beyond the *illustration* stages)
- We reused prior reviews as follows:
  - The search strategy of XY were considered in the design of the database searches
  - The sample of XY was imported as part of the search.
- The current status of the project is automatically updated with every change and reflected in the PRISMA chart (Page et al. 2021):

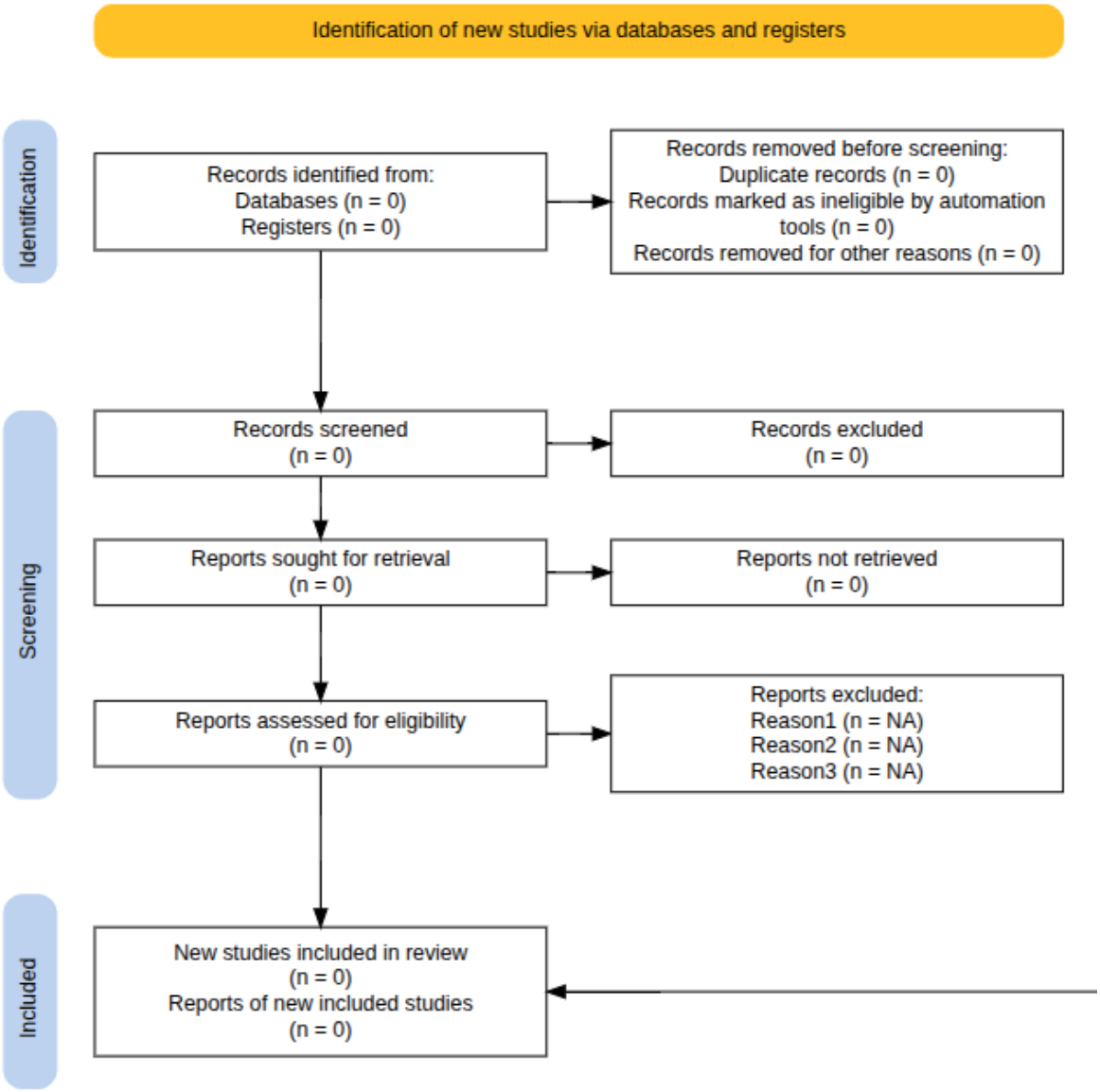


Figure 4: PRISMA Flow Chart (generated by python-prisma)

Repository

- GitHub Repository

Fiers, Fien. 2023. "Inequality and Discrimination in the Online Labor Market: A Scoping Review." *New Media & Society* 25 (12): 3714–34. https://doi.org/10.1177/14614448221128

379.

Haddaway, Neal R., Melissa L. Rethlefsen, Melinda Davies, Julie Glanville, Bethany McGowan, Kate Nyhan, and Sarah Young. 2022. "A Suggested Data Structure for Transparent and Repeatable Reporting of Bibliographic Searching." *Campbell Systematic Reviews* 18 (4): 1–12. https://doi.org/10.1002/CL2.1288.

Page, Matthew J., Joanne E. McKenzie, Patrick M. Bossuyt, Isabelle Boutron, Tammy C. Hoffmann, Cynthia D. Mulrow, Larissa Shamseer, et al. 2021. "The PRISMA 2020 Statement: An Updated Guideline for Reporting Systematic Reviews." *Systematic Reviews* 10 (1). https://doi.org/10.1186/S13643-021-01626-4.

Wagner, Gerit, Julian Prester, and Guy Paré. 2021. "Exploring the Boundaries and Processes of Digital Platforms for Knowledge Work: A Review of Information Systems Research." *The Journal of Strategic Information Systems* 30 (4): 101694. https://doi.org/10.1016/j.jsis.2021.101694.