Improving Reproducibility in Computational Social Science with the Methods Hub

Arnim Bleier, Johannes Kiesel, Christina Viehmann, Felix Victor Münch, Chung-hong Chan, Raniere Costa da Silva, Ahrabhi Kathirgamalingam, Po-Chun Chang, Muhammad Taimoor Khan, Stephan Linzbach, Fakhri Momeni, Ran Yu, Stefan Dietze, Claudia Wagner

[1]GESIS – Leibniz Institute for the Social Sciences
Unter Sachsenhausen 6–8
50667 Cologne, Germany
methodshub@gesis.org

Abstract

Computational methods have become central to contemporary social science research, ranging from classical statistical modeling and simulation to large-scale computational social science enabled by organic data. While these developments rely on increasingly complex and compute-intensive analysis pipelines, reproducibility remains a major challenge. The Methods Hub is a community-driven service developed at GESIS to address this gap by enabling reproducible, executable, and well-documented computational methods and tutorials tailored to the needs of social scientists. It combines curated content, persistent identifiers, and seamless integration with interactive execution environments, lowering technical barriers while promoting best practices in open and reproducible science. In our ICWSM 2026 demonstration, we will show how the Methods Hub (methodshub.gesis.org) helps make contributors' work more reusable, executable, and accessible to a wider audience.

Introduction

Research software is a key driver of scientific progress, yet it is often difficult to find, reuse, or reproduce computational methods once projects end (Rieder, Peeters, and Borra 2024). Moreover, research software is often lost when funding for a research project ends and the developed software is not prepared for reuse. Documentation is frequently incomplete, inconsistent, or too technical for researchers without strong programming backgrounds. Existing repositories and hubs provide valuable infrastructure but are often not optimized for usability, discoverability, or reproducibility in the social sciences. The Methods Hub responds to these challenges by offering a structured, curated portal for computational methods and tutorials. Its design builds on GESIS's long-standing experience in supporting social science research infrastructures and emphasizes accessibility, citability, and reproducibility. Rather than replacing existing platforms, the Methods Hub integrates with established archival and code-hosting services, ensuring sustainability and interoperability. Compared to generic code hosting and archiving platforms, the Methods Hub adds domain-specific curation, standardized documentation templates, and one-click execution links that make methods easier to discover, evaluate, and reuse in social-science workflows.

Core Concepts and Architecture

The Methods Hub provides 1-click reproducibility through interactive environments, a portal for discovery and access, and a curated collection of methods and tutorials. Together, they provide the technical and organizational foundation for reproducible computational analysis pipelines.

Execution and 1-Click Reproducibility

Executability is the foundation of computational reproducibility, which has been identified as a central but still insufficiently addressed requirement in computational social science research (Schoch et al. 2024). The Methods Hub builds on containerization technologies that have transformed modern software engineering through DevOps and MLOps practices. By extending these ideas to the social sciences, the Hub ensures that computational methods can be executed reliably across different systems and over time.

Technically, the Hub relies on Docker containers built from standardized dependency specifications. Using tools such as jupyter-repo2docker (Forde et al. 2018), containers can be automatically created from repositories hosted on platforms commonly used by researchers, including GitHub, GitLab, Zenodo, Figshare, and Dataverse. This approach ensures close alignment with existing publication and archiving workflows in the social sciences.

These containers can be deployed in multiple execution environments, depending on user needs:

- **MyBinder**[1] provides short-term, unauthenticated access for exploratory use and teaching scenarios that do not require extensive computational resources (Ragan-Kelley et al. 2018).
- **Jupyter4NFDI**[2] is a JupyterHub instance with repo2docker import capabilities supporting authenticated, persistent work sessions and access to large computational resources, including HPC infrastructures.

[0]The GESIS Methods Hub Team: https://www.gesis.org/en/services/processing-and-analyzing-data/analyzing-digital-behavioral-data/gesis-methods-hub

[1]https://mybinder.org, (GESIS participates in the BinderHub Federation: https://mybinder.readthedocs.io/en/latest/about/federation.html)

[2]https://hub.nfdi-jupyter.de

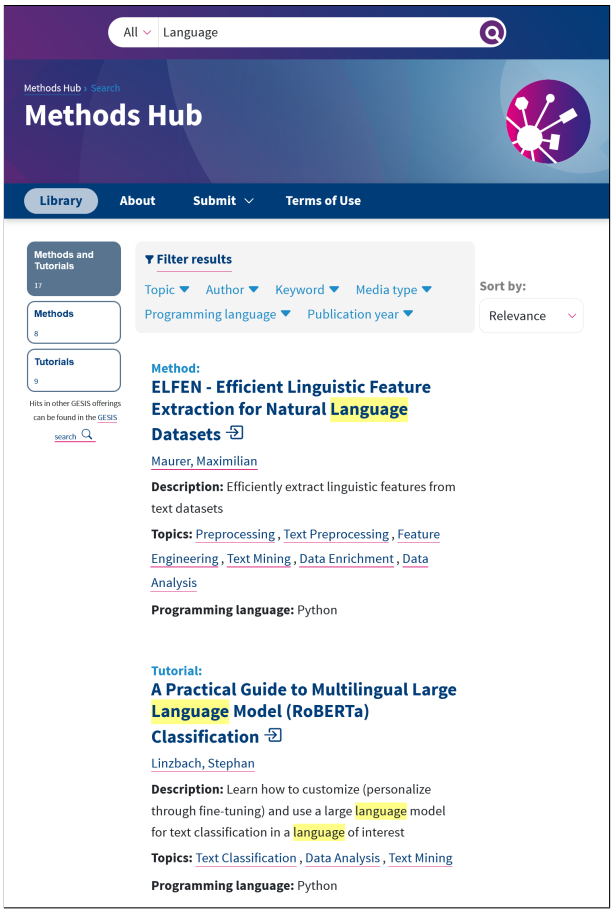

Figure 1: Excerpt from the Methods Hub search interface for the search query "Language" with the two best results and options for filtering the results and changing the ranking.

- **Additional environments** such as Google Colaboratory can be supported when required, for example to enable users from outside academia.

By abstracting away local installation and configuration issues, the Methods Hub allows researchers to focus on methodological questions rather than technical setup.

A Portal for Discovery and Access

Execution capabilities are surfaced through a dedicated frontend that acts as a central place for discovering and interacting with methods. The Methods Hub portal connects published computational analyses with executable environments through a searchable, curated gallery tailored to the social sciences. At the heart of the portal is the Methods Hub library, which integrates with GESIS Search. This integration ensures that methods and tutorials are discoverable alongside other research outputs such as publications, datasets, and tools. The search interface offers domain-specific facets, tailored result snippets, and direct action links for execution in interactive environments.

Curated Methods and Tutorials

The Methods Hub offers both computational methods and tutorials covering the full lifecycle of data-driven research, including data collection, preprocessing, analysis, visualization, validation, and workflow management. Content is presented from a non-technical, use-oriented perspective, allowing researchers to assess relevance before engaging with implementation details.

To ensure consistency and usability, the Hub employs an editorial process supported by checklists and templates. For methods, a standardized "Methods Hub–friendly README" guides authors through sections such as description, use cases, input and output data, hardware requirements, environment setup, and technical details. Similar templates exist for tutorials. These structures are designed to lower entry barriers for less technically oriented researchers while still supporting advanced use cases. Furthermore, this approach balances quality assurance with openness and interdisciplinarity, promoting FAIR access without enforcing rigid disciplinary boundaries.

Each method and tutorial is versioned, preserved via Git, and assigned a DOI. This ensures citability, long-term availability, and precise referencing of specific versions—key requirements for reproducible research.

Interactive Environments and User Experience

A defining feature of the Methods Hub is the tight integration of interactive execution environments. Users can explore rendered outputs directly on the Methods Hub website or launch methods and tutorials in external environments with a single click. This enables hands-on experimentation without local installation and supports learning-by-doing approaches.

The Methods Hub team actively supports contributors in adapting their content for compatibility with these environments, acknowledging that different platforms impose different technical constraints. By doing so, the Hub goes beyond being a static registry and becomes an active facilitator of reusable and executable research methods.

The Methods Hub Demo at ICWSM 2026

The Methods Hub demo at ICWSM 2026 will let attendees engage with the platform in three hands-on ways. First, visitors can browse and search the Methods Hub to discover relevant methods and tutorials and inspect their metadata and documentation (Figure 1). Second, visitors can launch selected items in interactive execution environments with a single click (Figure 2) to explore complete, executable workflows without local installation. Third, we will walk through the end-to-end submission workflow, including repository preparation, the "Methods Hub–friendly README" template, and the editorial checks used to improve usability and reproducibility.

We will provide QR codes and short URLs so that visitors can try selected demo items on their own devices and optionally register interest in contributing. Team members will be available throughout the demo to answer questions,

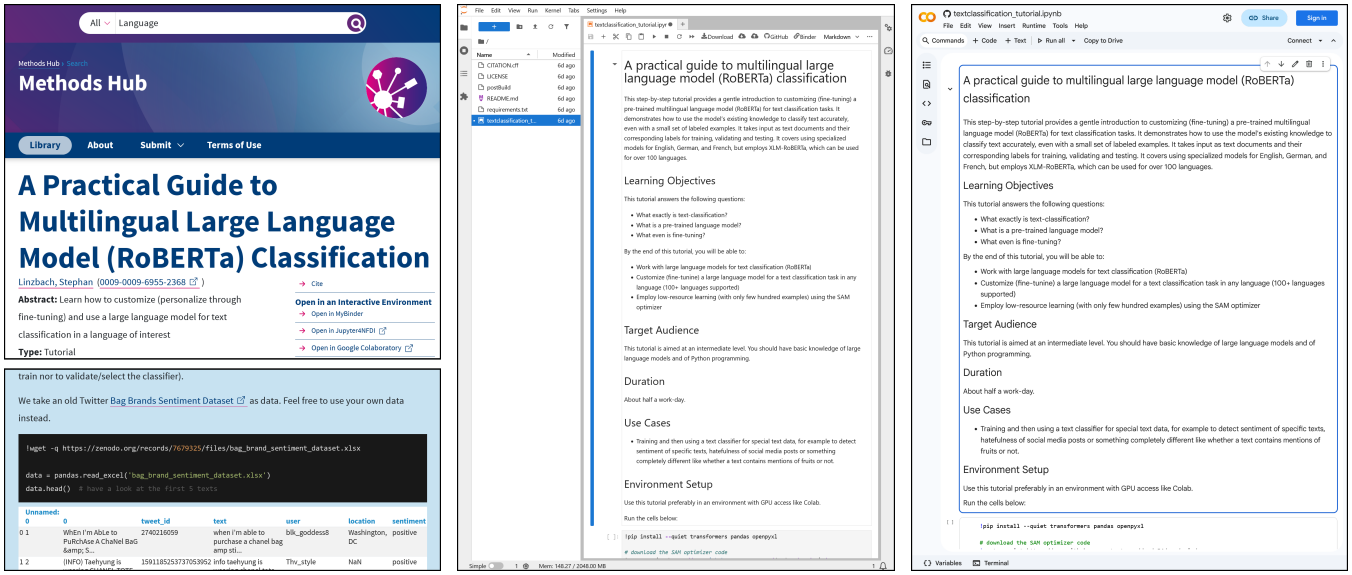

Figure 2: Several excerpts from the Methods Hub and linked interactive environments for a single tutorial are shown. The content page in the Methods Hub (top left) displays four action links on the right-hand side, including three links for opening the tutorial in interactive execution environments ("Open in *X*"–only available if the method or tutorial supports the respective environment). The content page also displays the results of a full tutorial execution (bottom left; the table shows the result of executing the code above), allowing users to quickly assess whether the method or tutorial meets their needs. The MyBinder and Jupyter4NFDI environments (the latter shown in the center) use the same interface. However, while the MyBinder environment does not require an account or registration, the Jupyter4NFDI environment requires a dedicated university account but offers better hardware. Using Google Colaboratory (shown on the right) requires a Google account and additionally provides access to GPU resources.

collect feedback, and discuss how attendees can adapt and submit their own methods or tutorials to the Methods Hub.

Conclusion

The Methods Hub provides an integrated solution for finding, reusing, and executing computational methods in the social sciences. By combining curated content, persistent identifiers, search infrastructure, and executable environments, it lowers technical barriers while promoting reproducibility and best practices in open science. Although developed as part of the GESIS service portfolio, its design principles and features make it attractive to a broad, interdisciplinary research community. The Methods Hub thus represents a concrete step toward making reproducible computational analysis the norm rather than the exception.

Acknowledgements

The work presented in this paper is financially supported by NFDI4DS[3] (DFG project no. 460234259) and Jupyter4NFDI[4] as part of Base4NFDI (DFG project no. 521453681). AI Disclosure: A generative AI tool was used to assist with language editing and readability improvements. All changes were reviewed and verified by the authors.

[3]https://www.nfdi4datascience.de

[4]https://base4nfdi.de/projects/jupyter4nfdi

References

Forde, J.; Head, T.; Holdgraf, C.; Panda, Y.; Nalvarete, G.; Ragan-Kelley, B.; and Sundell, E. 2018. Reproducible research environments with repo2docker.

Ragan-Kelley, B.; Willing, C.; Akici, F.; Lippa, D.; Niederhut, D.; and Pacer, M. 2018. Binder 2.0-Reproducible, interactive, sharable environments for science at scale. In *Proceedings of the 17th python in science conference*, 113–120. F. Akici, D. Lippa, D. Niederhut, and M. Pacer, eds.

Rieder, B.; Peeters, S.; and Borra, E. 2024. From tool to tool-making: Reflections on authorship in social media research software. *Convergence*, 30(1): 216–235.

Schoch, D.; Chan, C.-h.; Wagner, C.; and Bleier, A. 2024. Computational reproducibility in computational social science. *EPJ Data Science*, 13(1): 75.

Paper Checklist

1. For most authors...
   (a) Would answering this research question advance science without violating social contracts, such as violating privacy norms, perpetuating unfair profiling, exacerbating the socio-economic divide, or implying disrespect to societies or cultures? Yes
   (b) Do your main claims in the abstract and introduction accurately reflect the paper's contributions and scope? Yes

(c) Do you clarify how the proposed methodological approach is appropriate for the claims made? Yes

(d) Do you clarify what are possible artifacts in the data used, given population-specific distributions? NA

(e) Did you describe the limitations of your work? Yes

(f) Did you discuss any potential negative societal impacts of your work? NA

(g) Did you discuss any potential misuse of your work? NA

(h) Did you describe steps taken to prevent or mitigate potential negative outcomes of the research, such as data and model documentation, data anonymization, responsible release, access control, and the reproducibility of findings? NA

(i) Have you read the ethics review guidelines and ensured that your paper conforms to them? Yes

2. Additionally, if your study involves hypotheses testing...

(a) Did you clearly state the assumptions underlying all theoretical results? NA

(b) Have you provided justifications for all theoretical results? NA

(c) Did you discuss competing hypotheses or theories that might challenge or complement your theoretical results? NA

(d) Have you considered alternative mechanisms or explanations that might account for the same outcomes observed in your study? NA

(e) Did you address potential biases or limitations in your theoretical framework? NA

(f) Have you related your theoretical results to the existing literature in social science? NA

(g) Did you discuss the implications of your theoretical results for policy, practice, or further research in the social science domain? NA

3. Additionally, if you are including theoretical proofs...

(a) Did you state the full set of assumptions of all theoretical results? NA

(b) Did you include complete proofs of all theoretical results? NA

4. Additionally, if you ran machine learning experiments...

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? NA

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? NA

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? NA

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? NA

(e) Do you justify how the proposed evaluation is sufficient and appropriate to the claims made? NA

(f) Do you discuss what is "the cost" of misclassification and fault (in)tolerance? NA

5. Additionally, if you are using existing assets (e.g., code, data, models) or curating/releasing new assets, **without compromising anonymity**...

(a) If your work uses existing assets, did you cite the creators? NA

(b) Did you mention the license of the assets? NA

(c) Did you include any new assets in the supplemental material or as a URL? NA

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? NA

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? No, no personally identifiable information is curated.

(f) If you are curating or releasing new datasets, did you discuss how you intend to make your datasets FAIR (see **?**)? NA

(g) If you are curating or releasing new datasets, did you create a Datasheet for the Dataset (see **?**)? NA

6. Additionally, if you used crowdsourcing or conducted research with human subjects, **without compromising anonymity**...

(a) Did you include the full text of instructions given to participants and screenshots? NA

(b) Did you describe any potential participant risks, with mentions of Institutional Review Board (IRB) approvals? NA

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? NA

(d) Did you discuss how data is stored, shared, and deidentified? NA