



UvA-DARE (Digital Academic Repository)

From tool to tool-making: Reflections on authorship in social media research software

Rieder, B.; Peeters, S.; Borra, E.

DOI

[10.1177/13548565221127094](https://doi.org/10.1177/13548565221127094)

Publication date

2024

Document Version

Final published version

Published in

Convergence : The International Journal of Research into New Media Technologies

License

CC BY-NC

[Link to publication](#)

Citation for published version (APA):

Rieder, B., Peeters, S., & Borra, E. (2024). From tool to tool-making: Reflections on authorship in social media research software. *Convergence : The International Journal of Research into New Media Technologies*, 30(1), 216–235.
<https://doi.org/10.1177/13548565221127094>

General rights

It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations

If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: <https://uba.uva.nl/en/contact>, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (<https://dare.uva.nl>)

From tool to tool-making: Reflections on authorship in social media research software

Bernhard Rieder 

University of Amsterdam, Netherlands

Stijn Peeters 

University of Amsterdam, Netherlands

Erik Borra 

University of Amsterdam, Netherlands

Convergence: The International
Journal of Research into
New Media Technologies
2024, Vol. 30(1) 216–235
© The Author(s) 2022



Article reuse guidelines:

sagepub.com/journals-permissions

DOI: [10.1177/13548565221127094](https://doi.org/10.1177/13548565221127094)

journals.sagepub.com/home/con



Abstract

Social media research software has come to play increasingly important roles in processes of knowledge production. While epistemological, logistical, legal, and ethical concerns put the spotlight on the software tools researchers are relying on, little attention is paid to the role of the ‘toolmaker’ beyond a vague idea of the ‘power’ wielded by those who design, develop, and maintain these technical artifacts. This paper seeks to address this role, both conceptually and with attention to practical concerns, as a form of hybrid and relational authorship. We thereby shift the focus from tool to tool-making, from artifact to practice, in an attempt to produce a different kind of ‘un-blackboxing’ of tools than the somewhat overused tropes of open source code or open data. Our contribution proceeds in three steps. We first address tools and tool-making from a theoretical perspective, suggesting that their epistemological orientation reaches more deeply into the networks of research practice than words like ‘bias’ admit and proposing to consider the specific kind of hybrid authorship that emerges in this context. Calling on our own experiences as toolmakers, we then reflect on a cluster of issues where this authorial function becomes particularly visible. Here, we examine how motivations and commitments orient what a piece of software does and how it does it and discuss tool-making from the perspectives of co-development, maintenance and care, and ethics by design. We conclude by arguing that the most pressing concerns for tool-making lie in institutional arrangements that are crucial for the life of research software.

Corresponding author:

Bernhard Rieder, University of Amsterdam, Turfdraagsterpad 9, Amsterdam 1012 XT, Netherlands.

Email: b.rieder@uva.nl

Keywords

social media, research software, tool, authorship, tool-making, critical technical practice, ethics of care, ethics by design

Introduction

The role of social media platforms as central infrastructures and organizing agents in contemporary societies has come to be broadly recognized, as the frantic debates about misinformation, algorithmic governance, and monopolization attest. How we generate knowledge about both platform companies' conduct and the many different phenomena that emerge around and within their designed environments remains a difficult question, however. Secrecy, technical complexity, and sheer size constitute obstacles that are often hard to overcome for researchers hoping to understand 'platform life'. While computers are playing increasingly central roles in many academic disciplines (Jay et al., 2021), internet researchers are particularly dependent on software to deal with the specific difficulties encountered when trying to make sense of what goes on in the techno-corporate structures that have 'sucked up' important parts of human interaction.

'Software', here, can refer to data extractors interfacing with developer APIs or data scrapers that parse end user interfaces, but also to a broad set of old and new programs that serve at different stages of logistical, analytical, and interpretative processes, ranging from data handling and transformation to methods seeking to process *meaning*, such as topic modeling or sentiment analysis. The fact that many digital 'tools' are not merely transpositions of established calculative procedures, such as the correlation coefficient, into convenient interfaces but involving a wide variety of infrastructural logistics and analytical techniques – with potentially far-reaching epistemic effects – has in turn prompted debates around opacity, reliability, and methodological streamlining (e.g. Rieder and Röhle, 2012). Demands for 'tool criticism' (Traub and Van Ossenbruggen, 2015; van Es et al., 2018) and similar forms of academic reflexivity indeed start off from the idea that research software is not 'neutral' but a mediator that introduces its own 'biases' into the equation. The goal is to raise awareness about the different ways tools contribute to and orient research, but also to affect how these artifacts are designed and communicated in the first place.

While these broader considerations are not limited to the study of online platforms, a set of concrete issues have shaken up data-driven social media research more specifically. The recent 'APIcalypse' (Bruns, 2019), a substantial curtailing of data access in the wake of the Cambridge Analytica scandal, has made it more difficult for researchers to repurpose developer APIs. At the same time, initiatives such as Social Science One, which brokers access to Facebook data and funding for projects studying 'the effect of social media on democracy and elections'¹, and maker-for-research data interfaces, such as Facebook's Ad Library² or Twitter's new provisions for academic research³, promise more robust and transparent research conditions. Legal provisions, most importantly the EU's General Data Protection Regulation (GDPR) and upcoming Digital Services Act (DSA), introduce new limitations and requirements for data collection but also point to new rights for academic researchers vis-à-vis platform companies. Broader ethical concerns about indiscriminate mass observation of users have prompted the release of guidelines (franzke et al., 2020) by bodies such as the Association of Internet Researchers (AoIR), creating new demands on the design and use of research software.

While these epistemological, logistical, legal, and ethical concerns put the spotlight on the software tools researchers are relying on, there is little attention paid to the role of the 'toolmaker' – whether they are individuals, teams, or larger organizations – beyond a vague idea of the 'power'

wielded by those who design, develop, and maintain these technical artifacts. This paper therefore seeks to address this role, both conceptually and with attention to practical concerns, as a form of hybrid and relational *authorship* that involves orientation, intentionality, and ‘politics’. We do so from the vantage point of the embroiled practitioner: all three of us have been substantially engaged in developing social media focused research software over the last decade, working on tools such as Netvizz (Rieder, 2013), DMI-TCAT (Borra and Rieder, 2014), the YouTube Data Tools (Rieder, 2015), and 4CAT (Peeters and Hagen, 2022), which have been used by thousands of researchers to study social media platforms. This entanglement comes with its own blindspots but also affords unique and hopefully valuable insights into the different ideas, practices, and normative commitments that lurk behind the interfaces presented to end users.

Our goal, here, is to shift the focus from *tool* to *tool-making*, from artifact to practice, in an attempt to produce a form of ‘unblackboxing’ that is different from code inspections and technical forms of explainability. While these approaches have their own utility, there are other ways to tackle the substance, character, and epistemological ‘spin’ research software introduces into methodological chains. Following Philip Agre’s inspiring vision for a critical technical practice (CTP) that articulates ‘the craft work of design’ with ‘the reflexive work of critique’ (Agre, 1997: 155) in productive ways, we take this opportunity to examine our own contributions and situatedness in ways that are more fundamental and open-ended than the desire to increase the functional legibility of a specific technical artifact. In this sense, we reflect on the ‘social life of methods’ (Savage, 2013), but from the involved perspective of the technical practitioner. While not a manifesto of any sort, ours are ‘views from somewhere’ (Haraway, 1988: 590) and much like Agre’s personal and introspective narration of his changing relationship with AI, we draw heavily on lived experience; not to install some master discourse or interpretative privilege, but to mirror the realities of tool-making, where decisions have to be made at every step of the way – decisions that will affect those who use these tools and, by extension, the knowledge that is produced with their help. Although much of what follows applies to research software more broadly, we focus on social media research to stay close to the center of our expertise, to develop an argument that is high in detail, and to foreground a domain where the expanding role of tool-makers is particularly salient.

Our contribution proceeds in three steps. We first address tools and tool-making from a theoretical perspective, suggesting that their epistemological orientation reaches more deeply into the networks of research practice than words like ‘bias’ admit and proposing to consider the specific kind of hybrid authorship that emerges in this context. Calling on our own experiences as tool-makers, we then reflect on a cluster of issues where this authorial function becomes particularly visible. Here, we examine how motivations and commitments orient what a piece of software does and how it does it and discuss tool-making from the perspectives of co-development, maintenance and care, and ethics by design. We conclude by arguing that the most pressing concerns for tool-making lie in institutional arrangements that are crucial for the life of research software.

From tool to tool-making

When discussing research software, a first conceptual question concerns the objects being referred to. Definitions such as ‘software that is used to generate, process or analyse results that you intend to appear in a publication’ (Hetrick et al., 2014) productively demarcate the domain with an eye on practice and finality rather than specific technical properties. When looking at the landscape of objects that qualify, however, we find artifacts that differ substantially in terms of size, complexity, and level of integration (scripts, programs, full-fledged research infrastructures, etc.), but also with regard to scope and functional specificity (narrowly focused on a particular task in a chain or

covering multiple steps, targeting a single platform or several, etc.), production context (designed by a company, a research group, an individual researcher, etc.), cost to users (free to use, paid purchase, subscription, etc.), and many other aspects, including more ‘technical’ matters such as up-to-dateness or maintainability. Our goal, here, is not to provide an overview or classification of software artifacts used in social media research and elsewhere, but to ponder how making such artifacts constitutes a situated and relational practice that is never merely technical. It may for certain intellectual purposes make sense to distinguish between, for example, desktop tools and research infrastructures, but what we are looking for is to carve out a discursive position that builds inductively from individual cases and experiences, a *praxeology* rather than a *taxonomy*. One caveat, however: while our discussion is, at least in part, relevant to a broad set of software artifacts used in academic research, we are particularly interested in those that are not merely ad hoc solutions to analytical problems coming up in individual research projects, but *tools* that address an audience beyond an immediate group of collaborators and acquire a certain stability and unity. Many research projects do, in fact, promise to create software that will be useful to others, even if few survive for a significant time after the project ends. But this aspiration to create something that is, indeed, ‘useful to others’ is what we think of when we talk about tool-making.

The term ‘tool’ does, however, raise conceptual questions that need to be addressed. Used too casually, it may suggest mere utility, ignoring or minimizing software’s active role within processes of knowledge generation (van Es et al., 2021). While replacing it entirely with a term like ‘research software’ would overshoot the mark, it is crucial to think of tools as *mediators* that ‘modify the meaning or the elements they are supposed to carry’ rather than *intermediaries* that transport ‘without transformation’ (Latour, 2005: 39). Reducing software’s contribution to practical utility also implies that there could be something like tool-less research, which not only ignores hundreds of years of laboratory science, but also the recognition dear to media studies that pen, paper, and language itself are already *media* involved in shaping the way we think (Goody, 1977). From this perspective, one could locate the use of software in research practice within larger trajectories of change affecting method and methodology (e.g. Shapin and Schaffer, 1985).

Debates about the scholarly status of digital tools in areas like the digital humanities (e.g. Ramsay and Rockwell, 2012) have already shown that these issues are not limited to STEM fields. Software has become *constitutive* of research practice almost everywhere in academia, penetrating deeply into epistemic fabrics, most strikingly when complex methodological operations and decisions disappear behind point and click interfaces. *In extremis*, a tool may fully automate a methodological chain. But even if most research software is more open-ended, the specific ways data are collected and formatted, submitted to analytical techniques, and turned into numerical or graphical outputs is highly relevant, including the question how these interpretative gestures are made amenable to user input. Even a simple CSV file containing tweets is the product of design decisions on what data to include, which transformations to add (e.g. extracting top-level domains from URLs), whether to connect back into the platform (e.g. providing links to the original source), and so forth. Unreflective use of the term ‘tool’ risks obscuring the complicated epistemological constellations emerging in software-driven scholarly work.

Existing takes on the status of research software have already highlighted many of these issues, but there are good reasons to single out internet research in general and social media research in particular. On the one hand, social media platforms have emerged as novel and problematic research objects that pose methodological and logistical challenges due to their broad adoption, enormous scale, and peculiar character as complex techno-bureaucratic constructs. New kinds of datasets promise new kinds of insights, but a variety of technical, legal, and ethical concerns raise questions related to tool-making that are different from those commonly faced in STEM fields or the digital

humanities, for example, when it comes to brokering access to data that are enmeshed with platforms' commercial objectives. The great interest in social media has, on the other hand, led to the emergence of a large and lively field of research and software-making that spans many different disciplines. While our own tools primarily target researchers from the humanities and social sciences, who often lack the skills or time to code themselves, the involvement of disciplines like computer and information science makes for a space that calls for the reflective posture CTP encourages and demands. Unlike [Agre's \(1997\)](#) attempts to 'reform AI' in the direction of more openness and epistemic diversity, however, the challenge in the field of social media research is to navigate shifting grounds and methodological plenitude, where research software may serve as a 'stabilizer', taking on responsibilities that come with their own predicaments.

These arguments lead to another set of doubts, this time concerning the term 'bias', which is often used to address the apparently problematic non-neutrality of research software. On the one hand, overuse of the term conflates the meaning of bias as prejudice or partiality with the more circumscribed notion of statistical bias, which materializes when sampling or measurement errors lead to a statistic not representing the underlying population. There are good reasons to distinguish the former from the latter, even if the latter can have effects similar to the former in specific settings. On the other hand, when using 'bias' to describe all kinds of limitations, commitments, and idiosyncrasies of research software, one implies that there could be something like a piece of software without limitations, commitments, and idiosyncrasies, which is hard to imagine. And how would one define a normative baseline that represents 'neutrality' without full methodological consensus? We therefore prefer to reserve bias to its statistical meaning and use terms like 'orientation', 'substance', or simply 'functionality' to refer to research software's epistemic specificity and directionality. This rhetorical nuance also serves to highlight that tools – not unlike traditional academic outputs such as articles and monographs – are artificial objects that have been designed with intent and purpose; *everything* about them is caught up in some kind of orientation and preference structure, including the very fact that they exist in the first place (cf. [Endres, 2017](#)).

Moving from a perspective that focuses on distortions to one that takes the whole artifact as both constitutive and conditioning of research practice allows us to consider its full substance, that is, what it brings to bear in concrete situations. This includes more involved analytical gestures as well as basic – yet important – logistical dimensions. A tool may calculate the correlation coefficient the exact same way Pearson did in 1895, but the fact that it is much faster and can take into account much more data than a human computer has consequences for scientific practice ([Jay et al., 2021](#)). When thinking about social media research in logistical terms, a first consideration is that some platforms are served much better than others when it comes to tool availability, and the resulting difference in quantity, quality, and convenience is far from innocent. These differences depend in part on the level of data access provided, but the existence of a large number of data retrieval tools for Instagram, despite a quasi-universal loss of sanctioned API access, shows that much is possible given sufficient motivation.⁴ Questions like 'for which platform do we have working data extractors?', 'how much data can we get?', and 'how fast can we get it?' are hardly the first thing that comes to mind when we think about 'bias', but they are fundamental for orienting what kind of research can be undertaken.

In 1958, Simondon wrote that '[c]ulture is unbalanced because it recognizes certain objects, like the aesthetic object, granting them citizenship in the world of significations, while it banishes other objects (in particular technical objects) into a structureless world of things that have no signification but only a use, a utility function' ([Simondon, 2017](#): 11). When we *do* recognize the capacity of technical objects to mediate and signify, however, we not only arrive at questions of how this happens in detail and with what effects, but also of origination, expression, and *authorship*. In the

context of research software, any developer will notice that their tools are often named or linked but not academically referenced, even if there is an accompanying publication mentioned prominently in the interface. Netvizz, for example, is mentioned 2020 times on Google Scholar at the time of writing, but the relevant paper (Rieder, 2013) only 710 times. For Gephi, a prominent software for visual network analysis, this is 28,500 times vs. 7930 times for the paper (Bastian et al., 2009). It may seem innocuous or even petty to highlight this disparity, but it is indicative of a broader tendency to treat research software as a mere ‘utility function’ rather than a full-fledged scholarly contribution.⁵ This is particularly problematic if we consider tool-making, in light of the arguments presented above, to be a *prescriptive* gesture, a statement about what research there *should* be and how that research *should* look like. None of this happens in isolation, of course, and the ‘death of the author’ (Barthes, 1984) in the crossfire of intertextuality and interpretation applies no less to technical than textual objects. It is thus not our goal to install tool-makers as singular and sovereign originators, but to understand and locate tool-making as a hybrid form of authorship that crystallizes large networks of relations in technical artifacts that become themselves nodes in others’ research practice.

Within academia, the centrality of authorship is indeed not primarily a statement about the metaphysics of ideas but rather a form of conversational tracing, a means to establish accountability and responsibility for knowledge creation, in the sense of assuring opportunity for ‘account-giving’ or ‘response-giving’. Although the debates about scholarly credit for research software raise important questions (e.g. Ramsey and Rockwell, 2012), not least about the relationship between acknowledgement and software quality (Jay et al., 2021), our main objective with the term ‘authorship’ is to foreground the directionality, relationality, contingency, embeddedness, and overall ‘baggage’ research software comes with. There is more to say about software than what is found in source code and, heeding Agre’s call for CTP, if initiatives like ‘tool criticism’ are to succeed, toolmakers and tool-making ‘situations’ have to be brought into focus.⁶

We will dive into the complex set of roles toolmakers have come to play in the next section, but it already bears mentioning that the challenges we alluded to in the introduction – changes in data access, legal context, and ethical considerations – have led to a further ‘widening’ and ‘thickening’ of these roles. As social media research tools are caught up in increasingly complex settings and concerns, the ‘decision-load’ for toolmakers grows, further expanding their central position as designers, brokers, and communicators of research practice.

Authorship in tool-making

While these conceptual remarks establish a broader context for our argument, this section is our attempt to address some of the more specific issues that arise around tool-making and, indeed, to engage in a form of ‘account-giving’ with regard to our own contributions and choices. The goal is to critically examine some of the artifacts we have produced, and to reflect on our commitments, practices, and various forms of ‘situatedness’.⁷ The following five dimensions move from the initial motivations behind a specific project to much broader institutional concerns, paying attention to the many questions and decisions that come into play at each step of the way. We mainly argue on the basis of our own experiences here, but also draw on long-standing conversations with other groups and projects.

A distinction that runs through all of these points is that between endogenous and exogenous aspects, that is, between more directly technical questions of making and design, and more contextual issues that arise from the various ways tools connect to the world. While the two cannot be fully held apart, it makes sense to distinguish between the work that goes into creating and

maintaining a technical object and the various practices that contribute to the ‘life’ of research software, such as communication, documentation, and training, as well as the question of how tool-making is structured in organizational and institutional terms. In the former, toolmakers appear first and foremost as technologists, while the latter highlights the many ways they contribute beyond writing code and designing interfaces. Both roles, which conflate in practice, imply certain kinds of influence and power over users and use practices, and therefore demand normative reflection. Interrogating the way functionalities enable and constrain leads to *design ethics* and similar strands of thinking. Pondering the various relationships between tools and social context, however, suggests a broader *ethics of care* (Tronto, 1998) as a possible framework. While our goal in this paper is not to engage with moral philosophy substantively, these references constitute guideposts for connecting the practice of tool-making to established normative debates.

Motivations and commitments

When reflecting on tool-making, one can start by asking why tools are made in the first place. Answers to this question will heavily depend on context. For the specific case at hand, we begin by noting that social media platforms have come to play central roles in contemporary societies and are therefore generating strong scholarly interest in a variety of disciplines. Their broad impact means that governments and civil society also have a stake in knowledge generation. There is public interest in assuring the ‘observability’ (Rieder and Hofmann, 2020) of both platform companies’ conduct, for example with regards to algorithmic ordering, and the social phenomena that emerge within their designed infrastructures. The immense scale and technical character of online platforms poses problems, however: while methods that do not rely on automated data extraction exist and have value, many questions we want to ask depend on our capability to study larger – although not necessarily representative – samples of users or content items. Unlike more technical disciplines, where coding skills are widely distributed, researchers in the humanities and social sciences often lack the capabilities to assemble and analyze such datasets, making tools that are comparatively easy to use valuable as they incorporate logistical capacities and methodological gestures that may otherwise not be accessible to a majority of interested scholars.

While these factors create a particularly fecund environment for tool-making, not least in terms of available funding, this does not mean that the existence and feature set of research software is sufficiently explained by these broad incentives. In concrete circumstances, a variety of motivations and contingent reasons may come into play and many of the most heavily used tools in social media research are not the result of a straight line going from research proposal to funded project to working technical object that is widely used. In our own practice, many tools grew out of concrete scholarly interests that required writing some programming code to capture or analyze data. Others were experiments meant to explore research possibilities or specific analytical directions. In other words, tools often start as ‘buggy prototypes [and] tentative solutions’ (Van Es et al., 2021) and – importantly – they may not even start with the intention of becoming a tool in the sense of a full-fledged program that can be easily used by others. Many of these scripts or prototypes are discarded after the research project at hand finishes. This ‘pre-tool’ phase is crucial because it underlines something that is usually left unsaid: that tools are often part of a non-teleological discovery process. Even when a more stable artifact emerges from this initial ‘screwing around’ (Ramsay, 2014), it may not fully shed its experimental origins.

Our focus here is on those tentative solutions that *do* go on to become software used not only by their developers but also by other researchers.⁸ But even here, motivations are diverse and the goal to ‘serve users’ may only be one of many. Jacomy (2020) goes as far as arguing that ‘science tools

are not made for their users', listing a whole range of other reasons why someone may build a piece of software, including gaining visibility for oneself or one's research philosophy, meeting a funder's needs, earning income, making something free-to-use for students, or fostering collaboration with peers. The goal may not even be to *have* a tool but to learn or derive stimulation from the process of making it. Motivations may overlap, shift over time, and come to include a more user-oriented perspective as a tool gathers popularity; but it would be a mistake to assume that research software is necessarily created to help others do research. The following examples from our own work can illustrate the variety and idiosyncratic nature of tool-making.

4CAT, to start with a recent example, began as a set of scripts with a clearly demarcated purpose: to collect data from 4chan, a forum from which posts are deleted shortly after posting. Later, to make it easier for collaborators less versed in Python, a web interface was added to the scripts to allow them to create such datasets as well. Further enhancements, such as the addition of other platforms from which data could be collected, were also driven by concrete research needs of the authors or their direct peers. Eventually, as more people started using 4CAT and its reach grew beyond the initial circle of collaborators, the central concern became to make a tool that not only allowed users to capture and analyze data, but to do so *in an informed way*. This was partially a response to the oft-articulated criticism that research software 'black-boxes' research, providing users with an output without making it clear how that output was generated. Building 4CAT was thus an attempt to react to this criticism in practice and the methodological strategy was to experimentally create a tool that, amongst other things, provides clear traces of used parameters and direct access to the code fragments behind specific analytical procedures. In this, we of course seek to enable 'better' research for 4CAT's users and their feedback is invaluable for iterative improvements, but simply 'serving users' was never the central concern.

DMI-TCAT, a much-used platform for collecting and analyzing Twitter data, followed a somewhat different mode of reasoning. Here, our initial interest was to test Twitter's Streaming API and in particular the 1% sample. When we noticed the great potential for research and the lack of existing tools to exploit it, we transformed an initial stand-alone script into a more robust toolkit that would allow for both collection and analysis of Twitter data. The goal became to a) lower the barriers of entry in logistical terms by making data collection easier to use and less expensive, particularly to compete with in-house research; b) promote 'epistemic plurality' by allowing for different sampling strategies and analytical techniques; and c) foster robust and reproducible research by paying attention to issues like sampling errors and data import/export. Instead of pursuing a single methodological direction, we hoped that this would broaden *who* can study *what* on the platform, enabling a multi-angled form of platform accountability.

Netvizz, a now defunct extractor for Facebook data, again tells a different story. It initially started in 2008 as a purely technical demonstrator for a course on web programming, then morphed into a heavily used teaching device for network analysis. When Facebook removed the capacity to generate friend networks in 2015, it re-focused on Pages and Groups, finding new uptake as a means to study content rather than connectivity. It finally shut down in 2019 after Facebook removed most third-party access to its developer APIs. There is again not a single 'motivation' at work here, but a series of initially vague intuitions and purposes that evolved over time. Despite the lack of funding and many difficulties dealing with Facebook, the very large user base (at certain points over 10,000 unique users per month) created a sense of commitment that kept Netvizz alive. But more importantly, the long-term engagement with the platform became itself a research methodology – or at least a source of inspiration – within a larger platform- and software-studies perspective. The project provided an ongoing outlet for hands-on engagement with different aspects of tool development

(coding, interface experimentation, maintenance, etc.) and served as a means to study a platform's technical, social, and legal interactions with third-party developers.

These three examples are by no means exhaustive, but they widen and complicate the tool criticism debate by highlighting the complex and diverse motivations that drive software-making for research, moving away from an overly simplistic understanding of tools as little more than providers of functionality. The different epistemic roles these artifacts can play for their creators inform their specific choices and orientations and, by extension, the specific kind of scholarly contribution they constitute.

Co-development

While we have, so far, used the term toolmaker in a way that may evoke a single individual working on their own, this is clearly not the only – or even the most common – situation. Many tools in social media research are made by teams that bring together different specializations such as programming, interface design, methodological expertise, and subject matter familiarity. Authorship and responsibility for the resulting tools are shared between group members in these cases. The differences between production settings are clearly relevant and there would be much utility in a sociological comparison between projects using different methodologies for organizing interdisciplinary group work (cf. [Barry et al., 2008](#)). In this section, however, we do not venture much further into these differences and instead emphasize three series of embeddings that render any kind of tool-making a form of co-development, even when produced by a team of one. The concept of tool authorship we have in mind does not suggest a sovereign or unrestrained creator, but one that is constantly bound to a network of actors and materialities.

On the first level, we find two broad sets of entanglements. As already mentioned, online media and the various phenomena they connect with have become objects of interest and concern in academia and beyond, which means that research software enters into a conversation with existing knowledge and methodology. The thought processes that go into a specification document or into the more open-ended exploration that is typical of our own work happens within an ambient sphere of theoretical concepts, formulated research interests and directions, previously established findings, methodological strategies, and so forth. Academic inquiry is a collective endeavor, even in its most solitary moments, and many research tools are, in fact, little more than the application of well-known principles to new contexts. A second source of entanglement derives from the specific capacity of software to package all kinds of techniques into easily reused functional units ([Rieder, 2020](#)). Programming languages, operating systems, toolkits, libraries, and other 'containers' of functionality facilitate and orient development, and this holds true for academic software production. Libraries like the Natural Language ToolKit (NLTK) ([Bird et al., 2009](#)) implement a wide variety of text-processing functionalities, Matplotlib ([Hunter, 2007](#)) facilitates the creation of statistical charts, D3.js ([Bostock, 2012](#)) is often used to create interactive graphics, and so forth. Research software is heavily dependent on the labor and dedication that goes into creating the thousands of building blocks that inform development every step of the way. In some cases, a tool may simply be a (graphical) interface to a library that could not be used by non-programmers otherwise, which means that its contribution is not a new method but a widening of access to existing work. This can still be very relevant in practical terms, even if the contribution can be seen as merely adding 'convenience'. Taken together, these environments of academic and technical knowledge and technique constitute a first way of understanding tool-making as a form of co-development.

A second level is more specific to the case at hand. With social media research, creating a tool generally involves some level of discovery or probing of the ‘research affordances’ (Weltevrede, 2016) platforms provide. In certain understandings (Rogers, 2013), the construction of ‘digital methods’ indeed starts from an examination of ‘online devices’ to gauge how their ‘natively digital objects’ can be ‘repurposed’ for social and cultural research. This process of discovering possibilities – but also limitations and pitfalls – can be thought of as ‘technical fieldwork’ (Rieder et al., 2015) in the sense that research prospects depend on an in-depth understanding of matters such as data structures, API specifications, connection modalities, rate limits, bot protection measures, and so forth. The social media service under scrutiny is thus, in a sense, a co-developer of research software, each platform defining an idiosyncratic and potentially hostile environment that needs to be understood, tamed, or even tricked to be made amenable to research (Sandvig et al., 2014). Initial feature sets may well be driven by an immediate research interest, but this desire is necessarily mediated by encountering the materialities defined by platform owners. Following the logic of CTP, where learning from limitations is central, we can understand the process of diving into specifications, of bumping into technical restrictions, and of discovering ‘tricks’ and loopholes as epistemically productive, in the sense that it may lead to new pathways that turn into specific features or methods further down the line. It also sharpens our awareness of critical limitations, inconsistencies, and practical problems that researchers have to account for when interpreting findings.

On a third level, end users come into play. In some cases, a tool’s only users are their makers and their interests and needs are what drives development. When moving beyond this limited circle, user studies, feedback, feature requests, published work, and code- or data-sprints (Berry et al., 2016) are some of the means to bring in outside perspectives. If we think of tool-making for social media research as a discovery process, encounters with users are at least as important as the wrangling with platforms’ research affordances. This does not mean that every whim or desire can or should be taken into account, but that design decisions can be made with a clearer understanding of what is at stake, what kind of ‘will to know’ (Foucault, 2013) the tool can serve. Co-development along these lines can ideally result in a ‘distributed laboratory’ where the tool serves as a deployable environment for methodological innovation. However, toolmakers should be mindful that software can ‘formalize’ and ‘stabilize’ particular methodological approaches, unavoidably presenting a ‘preferred reading’ (cf. Hutchby, 2001: 445) of research objects through the selection of features and analytical pathways that are embedded in (or left out of) the tool. This is not a problem in itself, particularly if alternatives are available, but it places a responsibility on the toolmaker that cannot be avoided with a claim to co-development. From one perspective, co-development means that user and author both contribute to the evolution of a research tool; but one could also say that the tool *develops the user*, by presenting and framing particular approaches to research they may not yet be familiar with. Toolmakers can calibrate this relationship to a degree: they may present the tool as a *statement* – ‘this is how we think research should be done’ – or as part of an exchange that feeds directly back into the tool itself.⁹ But even in the second case, co-development hardly means equality.

Taken together, these three entanglements create a space of action for toolmakers that is bounded and ‘populated’ by other actors. Tool authorship can thus be seen as a process of *negotiation* that assembles various kinds of relationships that result in a functioning artifact.

Maintenance and care

Software rots over time. Not because the code itself deteriorates, but because the environment it runs in inevitably changes. When development stacks, operating systems, or browser engines are

updated, often for security reasons, software may need to be adapted to keep functioning correctly. These are well-known problems that require attention, but tools for social media research face other challenges as well. Often hosted online, they are particularly exposed to security vulnerabilities and may involve monthly payments; technical changes in platforms' APIs or interface code regularly break functionalities; rising user numbers may collide with insufficient compute resources and API rate limits; data access may require regular bureaucratic interaction (more on this later). This means that maintenance and resource planning are crucial and often entail more work than the actual creation process, work that remains invisible as it merely ensures continued functioning.

Maintenance is also 'choppy' and requires intermittent and difficult to anticipate intervention, often with a high level of urgency. What this really means in practice is often underestimated, as desire for novelty generally primes over concerns for sustainability. Vinsel and Russel have argued that contemporary societies – and academia is no exception – are marked by 'innovation-speak', which 'actively devalues the work of most humans, especially those who do the dirty work that keeps our technological civilization running [...] it fails to capture the essence of human life with technology – where maintenance and reliability are far more valuable than innovation and disruption' (2020: 13). Researchers indeed work in (funding) environments that are geared toward novelty and turnover, where one short-term project follows the next. When the project that hosted initial tool development ends, hand-wringing inevitably ensues about how to pay ongoing web hosting fees or how to address incoming bug reports and feature requests. Tools are too often left to fester and funding for ongoing maintenance and support is still hard to come by.

'Open-sourcing' software is sometimes suggested as a solution, from the assumption that others will contribute or continue the work as long as the project's source code and documentation are available. This passes over the substantial knowledge and experience gathered during the work on a tool, which feeds directly into further development. It also misses that well-functioning open source projects generally have strong and proactive leadership or community management. And, as discussed, research software can be seen as an argument for a particular methodological perspective, and this perspective is only implicitly embedded in source code. Making a tool's code available under a free license can still serve as a baseline for longer-term sustainability, even if much work is needed to make software truly open. This has been a major development direction for 4CAT, where a modular architecture and a well-documented Python API make it possible to contribute code without having to understand the tool's inner workings. Metadata is added to all outputs to help find the code module responsible for that output, making it easier for contributors to find their way around the codebase. Modularity also makes it easier to make non-code contributions, such as documentation and worksheets.

One area where tool-making for social media research specifically requires continuous effort is the brokering of data access, which can be seen as a form of 'bureaucratic maintenance'. While the APICALypse in the wake of the Cambridge Analytica scandal pushed the question into the spotlight when Facebook closed (most of) its APIs to (most of) the research community, the relationship with platform companies has always been precarious and reliant on largely invisible labor that cannot be easily standardized or outsourced. Gaining access to developer APIs was relatively easy until the mid-2010s, but complex and intransparent application processes, tool audits, and other bureaucratic work have become a constant companion. The already mentioned Netvizz ran through several vetting cycles over the years, including an investigation into potential abuses after Cambridge Analytica, which included an extensive questionnaire and conversations with Facebook's lawyers. Data access was ultimately cut anyway.

The YouTube Data Tools were also audited on the basis of design documents and source code, but continue to profit from generous access rights at the time of writing. In fact, one could argue that

their real value lies not in their rather trivial code and interface, but in these access rights. While newly registered projects receive 10k quota ‘units’ per day and individual projects can get bumped up to 100k after review, our token provides a staggering 50M, making it possible to provide large and complex data structures, such as video networks¹⁰, to hundreds of daily users. Possibilities for research are indeed heavily dependent on platforms’ access regimes, but also on the capacities of toolmakers to navigate them successfully. We should not underestimate the skills required for dealing with exogenous demands.

The role of ‘access broker’ can take other forms as well. While the already mentioned examples are largely invisible to users, data access regimes like the new academic research track in Twitter’s API v2 require ‘per-project’ rather than ‘per-tool’ application. Here, toolmakers can become active as process experts, sharing experiences from successful applications or even integrating the application protocol directly into the tool, through links or form fields. These increasingly complex bureaucratic processes affect what and how research is done, not unlike the co-development entanglements mentioned in the last section, and they can lead to direct repercussions for the material reality of the tool in question. For example, we negotiated increased access to Tumblr’s API for our instance of 4CAT but had to agree that any collected data would be deleted after 3 days. This required changes to the tool to allow for automatic deletion after a given amount of time, which then raised the question whether deleting data after a pre-configured interval would not be prudent in general, given the ever-present possibility that collected data may contain personal or otherwise sensitive information. In this way, seemingly abstract discussions about data access can lead to concrete changes and even inspire methodological and ethical reflection concerning the specific approach to research a tool should foreground.

These examples show how endogenous and exogenous aspects intermingle and suggest that we need to consider toolmakers not just as creators of technical artifacts but in much more expanded roles. The technical and bureaucratic issues mentioned in this section can be seen as part of a larger domain of *care* that is crucial for the life of a tool and the research it enables. We have not talked in great depth about elements like documentation, teaching materials, explanations of tool behavior, and other things that toolmakers often provide or are asked to provide. For example, we cannot answer the common question whether tool outputs are identical to what users see in the browser interface without constantly comparing these outputs ourselves. The utility of this information is clear, but providing it means investing time, including the time to answer an email or forum post. Connecting to an ethics of care, which ‘refers both to a mental disposition of concern and to actual practices that we engage in as a result of these concerns’ (Tronto, 1998: 16), can highlight how attending to these exogenous or ‘relational’ matters can be crucial for the actual practices researchers are able to develop around a specific tool; but it also serves as a critical reminder that caring for tools and users involves, much like in other domains, considerable amounts of labor – including emotional labor – that hardly find their way into budget considerations and funding applications.

Ethics by design

As a tool’s scope increases beyond its initial purpose, the question of how it can enable not just research, but research that conforms to certain legal and ethical standards becomes more pressing. Whereas research software can be built to conform to specific requirements when used in a single project, the situation gets more complicated when a tool is used for many different projects and deployed in many different legal and ethical contexts. In Europe, for example, the GDPR imposes certain legal requirements and California has its own Consumer Privacy Act (CCPA). Journalists follow different ethical directives than researchers, who may in turn be working within constraints

that vary between fields, institutions, and countries. Researchers in our field often follow the guidelines published by the Association of Internet Researchers ([franzke et al., 2020](#)), but there are many other standards published by research associations or interest groups, precluding one-size-fits-all solutions.

Whether unified standards would be desirable to begin with is debatable. We have in our own work generally sought to strike a balance between enabling a variety of ethical approaches, trusting that researchers are able to consider the ethical implications when using tools to capture, analyze, and potentially publish data, while simultaneously not promoting unethical uses. In general, this requires a light touch. In 4CAT, for example, we guide user behavior not through stringent constraints but through ‘nudges’ in the software’s interface, for example, by leaving the decision of whether to pseudonymize data to the user, but enabling it by default. Here, toolmakers should be aware that they are the chief designers of the ‘choice architectures’ ([Thaler et al., 2013](#)) users are being confronted with. With this role comes the onus to *offer the relevant choices* – the final responsibility for ethical conduct lies with the researcher, but by making it easy to, for example, pseudonymize, anonymize, or delete data, we can gently recommend a particular approach and contribute to a research ethos sensitive to potential privacy risks.

But there are limits to this perspective, particularly when considering that a tool may be used outside of the research community, by actors not working within institutional contexts that require ethical review and practice. Particularly with software that enables capture and analysis of large quantities of data, a potential concern is that those with less-than-benign objectives may also use them. This is well illustrated by Netvizz, where we decided to impose irreversible hashing on user names after we were able to retrieve data for over 1.9 M accounts from the ‘We are all Khaled Said’ Facebook Page for our own research ([Rieder et al., 2015](#)). It became clear that such information could be harmful to activists and dissidents if it were to be collected by, for example, government agencies. This underlines the difficulty of finding a generalized approach to embedding ethical choices within code and interfaces, and emphasizes the responsibility of the toolmaker to develop their software in dialogue with its use ‘in the field’.

These examples raise the question whether and how far an ‘ethics by design’ approach to research software is feasible and desirable. This approach was formulated particularly in the context of artificial intelligence research, as a call for engineers to ‘do no harm’ to human participants in experiments and to design autonomous artificial agents from the ground up to work within certain ethical frameworks of constraints (e.g. [Dignum et al., 2018](#)). When applied to research software, it can be seen as an appeal for toolmakers to seriously think about potential consequences and harms. Privacy, accessibility, security, and other values come into play here. Who should be able to use the tool? Is work being done to keep out potentially malicious actors? Are collected datasets secure? These can be difficult questions to answer and doing so may require specific expertise – software security is a craft that many toolmakers may not be proficient in and where tool work is actually funded, it is rarely an item on the budget. It is thus necessary to raise awareness of these issues and find ways to help academic toolmakers in confronting them.

Another problem within this decision space concerns the question of how strictly research software should follow the wishes of the platforms it collects data from. All platforms have some sort of end-user license agreement (EULA) that often prohibits automated data collection for anything other than a very limited set of purposes. This is both an ethical and legal issue. Platforms may resist efforts to collect data for legitimate research and toolmakers may be tempted to help circumvent these restrictions. And although EULAs, depending on jurisdiction, may not hold up in court, the very possibility of legal conflict may have chilling effects (cf. [Sandvig, 2017](#)). How one balances competing interests, here, is also a pragmatic question, contingent on how one estimates

the risk of repercussions from the company data is being collected from; a ‘famous’ (or notorious!) tool may have to implement a more limited data collection component than one that is able to ‘fly under the radar’. Suffice to say that toolmakers working in the social media space often struggle with these questions.

The embedding into different ethical debates can take more formal or standardized routes as well, in particular around deontological or principled movements that seek to further certain norms and practices in the wider research software space. For example, an aspect that has become increasingly central over the past few years is the ‘FAIRness’ of research data and software, where FAIR stands for ‘findability, accessibility, interoperability, and reusability’ (Lamprecht et al., 2020; Wilkinson et al., 2016). This is a set of principles formulated in response to a perceived lack of ‘good data management’ in data-driven research that leads to underlying data being unavailable or difficult to find and interpret. As institutions more and more often demand prudent protection of research data but simultaneously encourage open access publication of results, the question is how research tools might facilitate this.

Interestingly, FAIR principles are argued to be ‘distinct from peer initiatives that focus on the human scholar’, instead putting ‘specific emphasis on enhancing the ability of machines to automatically find and use the data, in addition to supporting its reuse by individuals’ (Wilkinson et al., 2016: 1). From the perspective of research software, as discussed here, this emphasis could arguably expand from *finding and using* data to *producing* data in a FAIR way. This is especially useful as making data findable, accessible, and so forth can be quite complicated in a landscape filled with a multitude of data publishing standards and platforms, from general-purpose repositories such as Zenodo and Figshare to university-specific homegrown initiatives and anything in-between.

Taken together, these examples show how heavily legal and ethical questions have come to weigh on tool-making, particularly within the social media space. While it may be a stretch to refer to the explicit and implicit answers given during development as a form of authorship, there are clear repercussions for potential uses and users. Dealing with these increasingly complex issues puts the spotlight on the institutional embedding tool-making unfolds in.

Institutions

Many of the arguments we find in critiques of the ‘power’ of research software resemble broader takes on software as ‘regulator’ (Grimmelmann, 2005) or ‘institution’ (Katzenbach, 2012) in the sense that it governs or even prescribes the individual and collective behavior of its users. But software-making is itself embedded in institutional environments, such as research groups, university administrations, funding schemes, and academic conventions. These elements structure the conditions, pressures, and incentives developers are facing, and they manifest in such basic matters as how much development time is available and, in more complex situations, questions concerning decision structures and division of labor in development teams or publication opportunities, career trajectories, and institutional recognition.

One of the largest concerns is the organization of funding around relatively short term (1-3 years) projects, which too often results in the abandonment of software after a project ends. This is not only a problem in the immediate sense that some piece of functionality is no longer available, but also because the *experience* accumulated within the specific domain, including the discovery process we have referred to as ‘technical fieldwork’, may dissipate and get lost. And attempts to make research processes more transparent and collaborative, such as the FAIR principles mentioned above, are seriously hampered by volatility. From this perspective, the question of how to preserve and stabilize not only tools, but favorable tool-making environments comes to the front, particularly if

we consider institutional arrangements as fundamental for the knowledge accumulation critical and efficient tool-making depends on.

In our own work, we have often favored a ‘researcher-programmer model’, where software-making is tied to individuals that are not hired on a project basis but academic staff creating research tools next to their normal research and teaching responsibilities, often in their spare time. While this model has many advantages, including a tight coupling between research interests and software work, it can easily lead to overwork when other duties are not adjusted accordingly. Another approach has been to rely on research software engineers (RSE), a new professional category within academia, who ‘combine a professional attitude to the exercise of software engineering with a deep understanding of research topics’ (Baxter et al., 2012: 1). While the research-programmer ideal type is a single-person production unit, working with RSE implies at least some division of labor, with a researcher being part of a collaborative setup. Universities have been experimenting with different kinds of organizational arrangements for these collaborations, and accounts like Smithies’ and Ciula’s (2020) reflection on King’s Digital Lab as ‘department and laboratory’ are highly useful for raising the awareness of possible strategies. Given the specific challenges mentioned in the previous sections, the tight integration of heterogeneous skills and areas of expertise is particularly crucial for the social media research field, adding to the difficulty of finding the ‘right’ arrangement.

Further up the chain, we find even more elaborate institutional structures, such as software sustainability institutes¹¹, which ‘develop, maintain, and disseminate best practices for research software sustainability, and build community around them’ (Katz et al., 2021). Foundations and companies are other institutional forms seeking to tackle the considerable problems we find in short term project-funded software development. Which of these approaches fits best within a given context depends on the specific kind of research tool envisioned, but it is clear that there is a dire need to think more thoroughly about institutional arrangements and how they can render tool-making more sustainable. How to design appropriate funding instruments¹² is a key part of this reflection, but institutional forms of collaboration, incentive structures, and valorization models (e.g. through publishing opportunities) are equally important components. Fleshing out concrete solutions is beyond the scope of this paper, but awareness of the centrality of these issues is a crucial first step.

In the end, the concerns voiced by tool criticism may actually have more to do with organizational practice and institutional embedding than with the technical objects themselves. As Agre (1997: 155) notes, realizing CTP ‘will require a praxis of daily work’ that challenges the cultural and institutional patterns of academia. Opacity, for example, is hardly a matter of access to code when most research tools are open source anyway. What may rather be the issue underlying common critiques is an insufficiently developed relationality between endogenous and exogenous elements, between research tool and research practice, harking back to some of the issues addressed above in reference to an ethics of care.

What is lacking, then, is not so much transparency, but meaningful understanding, which can only be realized through explanation and communication taking into account the ‘will to know’ and level of expertise of the user. This again means that documentation, training materials, and so forth need to be taken as part of the toolmaker’s role as mediator and budgeted accordingly; but more substantially, it points to the question how tool-making can occur in integrated environments where true interdisciplinarity can thrive over longer durations. While we are not saying that these environments do not exist, we do believe that they are often implicit and accidental constructs rather than planned with purpose. The question, then, is how to recognize working models, stabilize them, and make them stronger.

Outlook

In this paper, we have argued that recognizing tool-making as a contribution to academic knowledge production is an important part of dealing with the increasingly important role of software in research methodology, practice, and – often overlooked – education. While terms like ‘bias’ still suggest that epistemic neutrality is a possibility or at least a desideratum, we suggest that it is both inevitable and productive that commitments, idiosyncrasies, and contextual embeddings shape tool-making and, consequently, tools themselves. Inevitable because any design activity involves a plethora of choices and productive because tool-making is a site of expression and negotiation that can stimulate collective processes of methodological debate and sense-making.

Building software is a process of discovery and articulation, a way of probing a complex socio-technical world through the lens of operationalization. This concerns direct methodological questions, such as what data is extracted, how it is assembled into units, and which analytical pathways are provided. Here, research software amounts to a methodological position paper, to an idea or proposition that outlines a particular approach and, hopefully, tries to make a case for it. But it also concerns broader or more contextual questions, such as how design processes are organized, who can and does contribute, and what kind of funding model there is. Here, research software plays the role of a laboratory, in the sense of a setting or frame that organizes epistemic work in other ways than simply implementing the steps of a method.

As tools evolve over time, they become – ideally – sites and manifestations of individual and collective learning, their immutability softening as toolmakers, users, and studied platform environments evolve and enter into conversation, leading to modifications, enhancements, or completely new technical artifacts. All of these aspects – what a tool proposes in terms of methodological ideas, how it is organized as a process and project, and which values and motivations drive its design – are crucial for its eventual shape and possible effects. On all levels, normative commitments ensue and our reference to ethics by design and ethics of care tried to make these commitments explicit and connect them to conceptual reference points.

If CTP, as introduced by Agre, is first of all a critical self-examination of academic practice, we need to recognize that reflecting on research tools and their uses remains incomplete if design and development processes are left to the side. Although not the focus of this paper, this also holds for the research practices of end users, who are themselves becoming method-makers in the sense that they combine various gestures and artifacts into methodological chains. What are the decisions that they are making and what kind of hybrid and relational authorship are they engaged in when they construct their own research situations? And how can they account for the complex forms of mediation that the tools they patch together bring with them?

Our call for broader recognition of tool authorship is as much an attempt to raise the visibility of these processes as it is an effort to interrogate, conceptualize, and transform them within actual project proposals and research practices. As social media research becomes more complex on all levels, the mediating role of research software is ever more central, demanding critical consideration that goes beyond scrutiny of the tools themselves. While we spent considerable time discussing immediate concerns and design choices, it is on the level of institution-building that the most consequential interventions can happen. Moving tool-making into the spotlight in this way leads beyond necessary tool criticism to a wider reflection on how academic inquiry should be organized more broadly to respond not only to an increasing ‘technification’ of research practice, but to the ongoing incursion of computing into the world itself.

Acknowledgements

The authors would like to thank Mathieu Jacomy, the organizers and participants of the preparatory workshop for this special issue, and three anonymous reviewers for their helpful comments.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work has been supported by the Platform Digitale Infrastructuur Social Science and Humanities (PDI-SSH).

ORCID iD

Bernhard Rieder  <https://orcid.org/0000-0002-2404-9277>

Stijn Peeters  <https://orcid.org/0000-0002-0161-8019>

Erik Borra  <https://orcid.org/0000-0003-2677-3864>

Notes

1. <https://socialscience.one/our-facebook-partnership>, last retrieved 7 March 2022.
2. <https://www.facebook.com/ads/library/>, last retrieved 7 March 2022.
3. <https://developer.twitter.com/en/solutions/academic-research>, last retrieved 7 March 2022.
4. On the other hand, one of the reasons why there are so many Instagram data retrieval apps is because they often break, and in such a way that one basically needs to start over programming them. Yet toolmakers still keep trying due to the perceived importance of the platform.
5. As a reaction, there have been attempts to make tools more easily citable through DOIs and some projects even include ‘citation licenses’ that obligate users to reference them correctly. For a discussion in the hard sciences see [Katz and Stall \(2021\)](#). Additionally, making software FAIR may help here ([Lamprecht et al., 2020](#)).
6. We should note that there is an emerging genre of reflection on tool-making situations, for example [Smithies and Ciula’s \(2020\)](#) discussion of the rationale behind King’s College’s Digital Lab.
7. In his doctoral dissertation, Mathieu [Jacomy \(2021\)](#) draws on Donna Haraway’s notion of ‘situated knowledge’ to discuss and account for visual network analysis as situated practice, arguing that ‘dependencies and attachments are present everywhere, revealing the entanglement of any situation’ (27). Instead of appealing to some detached and decontextualized notion of truth, an understanding of all knowledge as situated asks for the kind of ‘account-giving’ we attempt in this section.
8. Although one could also include research software made for an imagined user that never finds an actual audience (cf. [Bradley, 2019](#)).
9. Mathieu Jacomy and Anders Munk further develop this argument in another article in this issue.
10. Video networks are based on YouTube’s ‘related videos’ API endpoint, which makes part of the platform’s recommendation system amenable to research. The generation of a single video network with a crawl depth of two can easily require tens of thousands of quota units.
11. <https://www.software.ac.uk/>, last accessed 7 March 2022. Related initiatives are developed in Research Software Engineer groups, such as in Germany (<https://de-rse.org/en/aims>, last accessed 7 March 2022).

12. Since 2020, our work is supported by the Dutch Platform Digitale Infrastructuur Social Sciences and Humanities (PDI-SSH), a forward-looking example for a funding body that invests in sustainable research infrastructure.

References

- Agre PE (1997) Toward a Critical Technical Practice: Lessons Learned Trying to Reform AI. In: Bowker GC, Star SL, and Turner W (eds), *Social Science, Technical Systems, and Cooperative Work: Beyond the Great Divide*. New York, London: Psychology Press, pp. 131–157.
- Barry A, Born G and Weszkalnys G (2008) Logics of interdisciplinarity. *Economy and Society* 37(1): 20–49. DOI: [10.1080/03085140701760841](https://doi.org/10.1080/03085140701760841).
- Barthes R (1984) La mort de l’auteur *Le Bruissement de La Langue*. Paris, pp. 61–67.
- Bastian M, Heymann S and Jacomy M (2009) Gephi: An Open Source Software for Exploring and Manipulating Networks *Third International AAAI Conference on Weblogs and Social Media*, pp. 361–362.
- Baxter R, Hong NC, Gorissen D, et al. (2012) The Research Software Engineer. Oxford, p. 4. https://www.pure.ed.ac.uk/ws/files/65195747/DR2012_12_1_.pdf
- Berry DM, Borra E, Helmond A, et al. (2016) The Data Sprint Approach: Exploring the field of Digital Humanities through Amazon’s Application Programming Interface. *Digital Humanities Quarterly* 9(4). <http://www.digitalhumanities.org/dhq/vol/9/3/000222/000222.html>
- Bird S, Klein E and Loper E (2009) *Natural Language Processing with Python*. Beijing; Cambridge. <http://www.dawsonera.com/depp/reader/protected/external/AbstractView/S9780596550967>
- Borra E and Rieder B (2014) Programmed method: developing a toolset for capturing and analyzing tweets. *Aslib Journal of Information Management*. DOI: [10.1108/AJIM-09-2013-0094](https://doi.org/10.1108/AJIM-09-2013-0094).
- Bostock M (2012) *D3. Js - Data-Driven Documents*.
- Bradley J (2019) Digital tools in the humanities: Some fundamental provocations? *Digital Scholarship in the Humanities* 34(1): 13–20. DOI: [10.1093/lc/fqy033](https://doi.org/10.1093/lc/fqy033).
- Bruns A (2019) After the ‘APocalypse’: social media platforms and their fight against critical scholarly research. *Information, Communication & Society* 22(11): 1544–1566. DOI: [10.1080/1369118X.2019.1637447](https://doi.org/10.1080/1369118X.2019.1637447).
- Dignum V, Baldoni M, Baroglio C, et al. (2018) Ethics by Design: Necessity or Curse? New York, NY, USA: Ethics, and Society Association for Computing Machinery, pp. 60–66. AIES ’18. DOI: [10.1145/3278721.3278745](https://doi.org/10.1145/3278721.3278745).
- Endres B (2017) *A Literacy of Building: Making in the Digital Humanities*. Making Things and Drawing Boundaries. Experiments in the Digital Humanities. JSTOR: 44–54.
- Foucault M (2013) *Lectures on the Will to Know*. Springer.
- franzke AS, Bechmann A, Zimmer M, et al. (2020) Internet Research: Ethical Guidelines 3.0. Available at: <https://aoir.org/reports/ethics3.pdf>.
- Goody J (1977) *The Domestication of the Savage Mind*. Cambridge: Cambridge University Press.
- Grimmelmann J (2005) *Regulation by Software*. Rochester, NY: Social Science Research Network. <https://papers.ssrn.com/abstract=2358693>
- Haraway D (1988) Situated Knowledges: The Science Question in Feminism and the Privilege of Partial Perspective. *Feminist Studies, Inc*, 14, pp. 575–599. DOI: [10.2307/3178066](https://doi.org/10.2307/3178066).
- Hettrick S, Antonioletti M, Carr L, et al. (2014) *Uk Research Software Survey 2014*. DOI: [10.5281/ZENODO.14809](https://doi.org/10.5281/ZENODO.14809).
- Hunter JD (2007) *Matplotlib: A 2D graphics environment*. *Computing in science & engineering*. IEEE Computer Society, pp. 90–95.

- Hutchby I (2001) *Technologies, Texts and Affordances*. Cambridge University Press, pp. 441–456. DOI: [10.1017/S0038038501000219](https://doi.org/10.1017/S0038038501000219).
- Jacomy M (2020) Science tools are not made for their users. *Reticular*. <https://reticular.hypotheses.org/1387>
- Jacomy M (2021) *Situating Visual Network Analysis*. Copenhagen: Aalborg University.
- Jay C, Haines R and Katz DS (2021) Software Must be Recognised as an Important Output of Scholarly Research. *International Journal of Digital Curation* 16(1): 6. DOI: [10.2218/ijdc.v16i1.745](https://doi.org/10.2218/ijdc.v16i1.745).
- Katz DS, Carver JC, Hong NPC, et al. (2021) Addressing Research Software Sustainability via Institutes. <http://arxiv.org/abs/2103.03690>
- Katzenbach C (2012) *Technologies as Institutions: Rethinking the Role of Technology in Media Governance Constellations*. Rochester, NY: Social Science Research Network. <https://papers.ssrn.com/abstract=2392711>
- Lamprecht A-L, Garcia L, Kuzak M, et al. (2020) Towards FAIR principles for research software. *Data Science Groth P, Groth P, and Dumontier M (eds) 3(1): 37–59*. DOI: [10.3233/DS-190026](https://doi.org/10.3233/DS-190026).
- Latour B (2005) *Reassembling the Social: An Introduction to Actor-Network-Theory Clarendon lectures in management studies*. Oxford; New York: Oxford University Press.
- Peeters S and Hagen S (2022) The 4CAT Capture and Analysis Toolkit: A Modular Tool for Transparent and Traceable Social Media Research. *Computational Communications Research* 4(2): 571–589.
- Ramsay S (2014) *The Hermeneutics of Screwing Around The Hermeneutics of Screwing Around*. University of Michigan Press. DOI: [10.2307/j.ctv65swr0](https://doi.org/10.2307/j.ctv65swr0).
- Ramsey S, Rockwell G and Gold MK (2012) *Developing Things: Notes toward an Epistemology of Building in the Digital Humanities*. Minneapolis:USA: University of Minnesota Press, pp. 75–84. <https://dhdebates.gc.cuny.edu/read/untitled-88c11800-9446-469b-a3be-3fdb36bfd1e/section/c733786e-5787-454e-8f12-e1b7a85cac72>
- Rieder B (2013) Studying Facebook via data extraction: the Netvizz application. In: *Proceedings of the 5th Annual ACM Web Science Conference, New York, NY, USA, 2 May 2013*, pp. 346–355. WebSci '13. Association for Computing Machinery. DOI: [10.1145/2464464.2464475](https://doi.org/10.1145/2464464.2464475).
- Rieder B (2015) YouTube Data Tools. Software. Available at: <https://tools.digitalmethods.net/netvizz/youtube/>.
- Rieder B (2020) *Engines of Order: A Mechanology of Algorithmic Techniques*. Amsterdam: Amsterdam University Press. DOI: [10.5117/9789462986190](https://doi.org/10.5117/9789462986190).
- Rieder B, Abdulla R, Poell T, Woltering R, and Zack L (2015) Data critique and analytical opportunities for very large Facebook Pages: Lessons learned from exploring “We are all Khaled Said”. *Big Data & Society*. SAGE Publications Ltd, 2, 2053951715614980.
- Rieder B and Hofmann J (2020) Towards platform observability. *Internet Policy Review* 9(4). DOI: [10.14763/2020.4.1535](https://doi.org/10.14763/2020.4.1535).
- Rieder B and Röhle T (2012) Digital Methods: Five Challenges. In: Berry DM (ed), *Understanding Digital Humanities*. London: Palgrave Macmillan UK, pp. 67–84. DOI: [10.1057/9780230371934_4](https://doi.org/10.1057/9780230371934_4)
- Rogers R (2013) *Digital Methods*. Cambridge: MIT Press.
- Sandvig C (2017) Heading to the Courthouse for Sandvig v Sessions. *Social Media Collective*. <https://socialmediacollective.org/2017/10/19/heading-to-the-courthouse-for-sandvig-v-sessions/>
- Sandvig C, Hamilton K, Karahalios K, et al. (2014) Auditing Algorithms: Research Methods for Detecting Discrimination on Internet Platforms. Seattle WA, p. 23.
- Savage M (2013) The ‘Social Life of Methods’: A Critical Introduction. *Theory, Culture & Society* 30(4): 3–21. DOI: [10.1177/0263276413486160](https://doi.org/10.1177/0263276413486160).
- Shapin S and Schaffer S (1985) *Leviathan and the Air-Pump: Hobbes, Boyle, and the Experimental Life*. Princeton, N.J.: Princeton University Press.
- Simondon G (2017) *On the Mode of Existence of Technical Objects*. Minneapolis: Univocal Publishing.

- Smithies J and Ciula A (2020) Humans in the Loop: Epistemology & Method in King's Digital Lab. *Routledge International Handbook of Research Methods in Digital Humanities*. Routledge: 155–172. DOI: [10.4324/9780429777028-13](https://doi.org/10.4324/9780429777028-13).
- Thaler RH, Sunstein CR and Balz JP (2013) Choice Architecture. In: Shafir E (ed), *The Behavioral Foundations of Public Policy*. Princeton University Press, pp. 428–439.
- Traub MC and van Ossenbruggen J (2015) *Workshop on Tool Criticism in the Digital Humanities*. CWI Techreport.
- Tronto JC (1998) An Ethic of Care. *Generations: Journal of the American Society on Aging* 22(3): 15–20.
- van Es K, Wieringa M, and Schäfer MT (2018) Tool Criticism: From Digital Methods to Digital Methodology *Proceedings of the 2nd International Conference on Web Studies - WS.2 2018*. Paris, France: ACM Press, pp. 24–27. DOI: [10.1145/3240431.3240436](https://doi.org/10.1145/3240431.3240436).
- van Es K, Schäfer MT and Wieringa M (2021) Tool Criticism and the Computational Turn. A “Methodological Moment” in Media and Communication Studies. *Medien & Kommunikationswissenschaft* 69(1): 46–64. DOI: [10.5771/1615-634X-2021-1-46](https://doi.org/10.5771/1615-634X-2021-1-46).
- Vinsel L and Russell AL (2020) *The Innovation Delusion*.
- Weltevrede E (2016) *Repurposing Digital Methods: The Research Affordances of Platforms and Engines*. Amsterdam: University of Amsterdam. https://pure.uva.nl/ws/files/2569586/168511_NIET_IDENTIEK_Weltevrede_RepurposingDigitalMethods.pdf
- Wilkinson MD, Dumontier M, Aalbersberg IJJ, et al. (2016) The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data* 3: 160018. DOI: [10.1038/sdata.2016.18](https://doi.org/10.1038/sdata.2016.18).

Author Biographies

Bernhard Rieder is Associate Professor of New Media and Digital Culture at the University of Amsterdam and a collaborator with the Digital Methods Initiative. His research focuses on the history, theory, and politics of software and in particular on the role algorithms play in the production of knowledge and culture. This work includes the development, application, and analysis of computational research methods and the investigation of political and economic challenges posed by large online platforms.

Stijn Peeters is an Assistant Professor at the University of Amsterdam's Department of Media Studies and the Amsterdam School for Cultural Analysis (ASCA). He is additionally the Technical Director of the Digital Methods Initiative and the lead developer of the 4CAT Capture & Analysis Toolkit. His research interests include digital methods, fringe platforms, and historical platform studies.

Erik Borra is Assistant Professor in Journalism and New Media at the University of Amsterdam and a collaborator with the Digital Methods Initiative. His research focuses on digital methods, controversy mapping, and the development of new digital research instruments.