

Tool Studies 2.0 – Zum Potenzial von Transformern für die Erkennung und Klassifikation von Software-Tools in DH- Publikationen

Burghardt, Manuel

burghardt@informatik.uni-leipzig.de
Computational Humanities, Universität Leipzig

Ruth, Nicolas

nicolas.ruth@web.de
Computational Humanities, Universität Leipzig

Niekler, Andreas

aniekler@informatik.uni-leipzig.de
Computational Humanities, Universität Leipzig

Einleitung: Zur Rolle von Tools in den Digital Humanities

Software-Tools spielen in den Digital Humanities eine zentrale Rolle, ja, sind geradezu genre-prägend für diese Disziplin. Diese wichtige, praktische Rolle ist u.a. belegt durch die Existenz diverser Tutorials und vor allem auch von Tool-Katalogen (vgl. Tab. 1) sowie Versuchen der einheitlichen Kategorisierung von Tools, etwa der TaDi-RAH-Taxonomie (Borek et al., 2016).

Tabelle 1: Übersicht zu verschiedenen Tutorials und Katalogen für Tools in den Digital Humanities.

Name	Typ	URL
Programming Historian	Tutorials	https://programminghistorian.org/
forTEXT	Tutorials	https://fortext.net/
CLARIAH-DE Tutorial Finder	Tutorials	https://teaching.clariah.de/search/
TAPoR	Katalog	https://tapor.ca/home
Digital Methods Initiative	Katalog	https://wiki.digitalmethods.net/Dmi/ToolDatabase
Alan Liu's DH Toychest	Katalog	http://dhresourcesforproject-building.pbworks.com
SSH Open Marketplace	Katalog	https://marketplace.sshopencloud.eu/
NFDI4Culture	Katalog	https://rijournal.com/article/57036/instance/5947376/ ; Tabelle 9, Appendix

Neben diesen stärker praxeologischen Aspekten finden sich auch diverse Diskurse um die Rolle und Implikationen von Tools in den Digital Humanities. Ein Thema

ist dabei etwa „Tool Criticism“, also der kritische Umgang mit Tools, insbesondere deren spezifischer Funktionsweise und den Effekten, die diese auf die letzten Ergebnisse haben (Koolen et al., 2019; Traub & van Ossenbruggen, 2015; van Es et al., 2018). Ein weiteres Themenfeld findet sich im Bereich der Mensch-Maschine-Interaktion, also den speziellen Anforderungen, die geisteswissenschaftliche Forschende an die Funktionalität und Usability von Software-Tools mitbringen (Burghardt & Wolff, 2014; Wolff, 2015).

Besonders stark ausgeprägt ist die Diskurslinie, die sich mit der Frage um die epistemologischen Effekte – also die unmittelbare Auswirkung von digitalen Tools auf den geisteswissenschaftlichen Erkenntnisprozess – beschäftigt (Burghardt et al., 2022; Dalbello, 2011; Kaden, 2016; Ramsay & Rockwell, 2012).

Dabei erstrecken sich die genannten Diskussionen vornehmlich auf Einzelbeispiele, größere empirische Untersuchungen gibt es bislang nur wenige. Dazu zählt u.a. eine Studie aus dem Umfeld des Research Software Engineering, bei der die Art und Häufigkeit von Softwarezitationen in DH-Abstracts analysiert wurde (Henny-Krahmer & Jettka, 2022). Weiterhin sind hier die zahlreichen Experimente von Frank Fischer und Kollegen, die u.a. DH-Abstracts und Tutorials des Programming Historian auf Tool-Vorkommen hin untersucht haben (Barbot et al., 2019; Fischer & Moranville, 2020b, 2020a; Zarei, Alireza et al., 2022) sowie auch erste eigene Experimente (Burghardt et al., 2022) zu nennen. Ein erster Austausch zwischen den unterschiedlichen Akteur*innen im deutschsprachigen Raum fand weiterhin im Rahmen einer gemeinsamen Veranstaltung mit dem Titel „Die Werkbänke der Digital Humanities: Zur Rolle von Tools und Software für die Forschungsarbeit“ bei der vDHd (2021) statt.

Als grundlegende Methoden zur Erkennung von Tools finden sich in den genannten Studien vor allem zwei Ansätze: (1) lexikonbasierte Ansätze, bei denen mithilfe bestehender Tool-Listen, wie sie bspw. in den in Tab. 1 genannten Ressourcen verfügbar sind, ein einfacher look-up in einem Zielkorpus von DH-Publikationen erfolgt. Dieser Ansatz lässt sich sehr schnell umsetzen, leidet aber unter den üblichen Einschränkungen lexikonbasierter Verfahren, bspw. der Unvollständigkeit von statischen Wortlisten und gleichzeitig die Ambiguität einzelner Einträge (R, Python, Gate, etc.). (2) Der zweite Ansatz versucht diese Einschränkungen zu überwinden, indem auf verschiedenartige Verfahren des maschinellen Lernens mit dem Ziel eines Klassifikationstasks gesetzt wird. Hier gibt es einerseits vortrainierte Modelle aus dem Bereich der Software Entity Recognition (Patrice & Romary, 2015; Schindler et al., 2022), die aber zumeist aus dem naturwissenschaftlichen Bereich kommen und deshalb nur mäßig für den Einsatz in Digital Humanities-Publikationen geeignet sind (vgl. Henny-Krahmer & Jettka, 2022). In einer aktuellen Publikation experimentieren Zarei et al. (2022) mit einem auf Prodigy und dem spaCy-Framework basierenden NER-Ansatz, welcher für ein sehr klar umrissenes Anwendungsszenario mit vorhergehendem Training gute Ergebnisse bringt. Für einen Ansatz, der in der Lage ist auch Tools zu erkennen, die nicht vortrainiert wurden, schlagen die Autoren weiterführende Ansätze mit aktuellen Transformer-Modellen wie bspw. BERT (Devlin et al., 2019) vor. Erste Experimente mit

BERT für eine binäre Klassifikation von Sätzen mit / ohne Tools wurden von uns bereits im Rahmen einer Vorstudie (Burghardt et al., 2022) erfolgreich durchgeführt und zeigten sich nach einer ersten qualitativen Evaluation als sehr vielversprechend. Das vorliegende Paper knüpft hier nahtlos an und präsentiert Erkenntnisse aus aktuellen Experimenten mit dem RoBERTa-Modell (Liu et al., 2019), einer optimierten Variante des bekannten BERT-Modells. Der von uns verfolgte Ansatz ist neben einer grundlegenden Identifikation von Tools über deren Embedding-Vektoren auch in der Lage die Tools größeren Kategorien, wie bspw. *Textanalysetool* oder *Visualisierungstool*, zuzuordnen. Wir glauben, dass wir mit einem solchen Ansatz einen wichtigen Beitrag zu den bestehenden Tool-Diskursen in den DH leisten können und hoffen damit weitere empirische „Tool Studies“ zu befördern.

Tool-Identifizierung und -Klassifikation mit RoBERTa Tool-Embeddings

Der Task der Tool-Identifizierung und -Klassifikation lässt sich methodisch als Problem der Text-Klassifikation einordnen. Beim gewählten Ansatz handelt es sich um ein Verfahren des überwachten Maschinellen Lernens, welches konkret aus einer binären und einer mehrklassigen Klassifikationsaufgabe besteht. Die grundlegende Idee beim nachfolgenden Vorgehen ist, dass die Erwähnung eines Tools in einem Paper in einem spezifischen sprachlichen Kontext steht, der sich in Form eines Sequenz-Embeddings mithilfe eines transformer-basierten Sprachmodells abbilden und klassifizieren lässt.

Korpus, Trainings- und Testdaten

Am Anfang steht die Erstellung spezieller, gelabelter Trainings- und Testdatensätze. Der Ausgangsdatsatz, der ebenfalls als Untersuchungsgegenstand für die folgende Analysen dient, besteht aus 3.737 englischsprachigen Zeitschriftenpublikationen aus dem Bereich der Digital Humanities, die zwischen 1966 und 2020 veröffentlicht wurden (Luhmann & Burghardt, 2021). Eine manuelle Extraktion von Sequenzen in Papers, die die Nennung eines Tools beinhalten, ist in Anbetracht der vorliegenden Datenmengen nicht durchführbar. Stattdessen wurden automatisch bekannte Tools im Korpus gesucht, um entsprechende Sequenzen ausfindig machen zu können. Um qualitativ hochwertige Trainingsdaten zu erhalten, haben wir zunächst nach besonders populären Tools gesucht. Dazu haben wir insgesamt acht Listen und Tutorials mit Toolnennungen (vgl. Tab. 1) durchsucht und nur diejenigen Tools ausgewählt, die in mindestens zwei unterschiedlichen Listen genannt wurden. Aus diesen Tools haben wir dann zur weiteren Qualitätssteigerung diejenigen entfernt, die ein hohes Maß an Ambiguität aufwiesen. Dieses reduzierte Lexikon, mit insgesamt 246 Tools, dient als Ausgangspunkt für die Erstellung der Trainingsdaten. Im nächsten Schritt wurden sodann alle Papers des Korpus tokenisiert, mit dem Tool-Lexikon durchsucht und bei einem Treffer als Textaus-

schnitt mit 15 Tokens vor und 15 Tokens nach dem Treffer als Sequenz-Label-Paar in den Trainingsdatensatz übernommen. Anschließend wurde die Menge der gefundenen Sequenzen durch zufällig gewählte und gleichmäßig über alle Papers verteilte, weitere Sequenzen ergänzt, die als Negativbeispiele für Toolnennung dienen.

Als Label wird hier für den binären Trainingsdatensatz „Tool“ und „Kein Tool“ verwendet und für den mehrklassigen Trainingsdatensatz die Zugehörigkeit des Tools zur bestehenden Taxonomie von Alan Lius „DH Toychest“, da diese fast alle Tools in unseren Trainingsdaten bereits enthält und aus unserer Sicht auch gut nachvollziehbar ist. Langfristiges Ziel ist hier die Übernahme einer standardisierten Taxonomie wie TaDiRAH. Anschließend wurden die Datensätze für jeden der beiden Klassifikationstasks nach dem Prinzip einer *5-fold cross-validation* in fünf Trainings- und Testsätze geteilt. Final besteht der Datensatz aus 3.780 Sequenz-Label-Paaren in jedem der fünf *folds*. Für jede Runde der Kreuzvalidierung wurden vier *folds* für das Training und der fünfte *fold* als Testdatensatz für die Validierung benutzt.

Modell-Evaluation

Als transformer-basiertes Sprachmodell wurde eine RoBERTa Implementierung von Hugging Face gewählt.¹ Aus Effizienzgründen wurde konkret „distilroberta-base“ als „case-sensitive“- und „knowledge distilled“-Variante verwendet. Die Ergebnisse dieser Modell-Evaluation sind sehr vielversprechend. So wird für den Klassifikationstask der binären Tool-Detektion für beide Klassen bei den 5 *folds* im Schnitt ein F1-Score von 0,99 erreicht. Für den Task der Tool-Kategorisierung wurden ebenfalls gute F1-Scores im Bereich 0,94 - 0,99 erzielt, die sich von Kategorie zu Kategorie geringfügig unterscheiden (vgl. Tab. 2).

Tabelle 2: Ergebnisse des mehrklassigen Klassifikationsansatzes. Kategorien mit zu geringem Vorkommen für aussagekräftigen Klassifikationswert wurden entfernt (Testsamples > 100). *Mittelwert über alle Runden der Kreuzvalidierung

Class	F1-Score*
Authoring / Annotation / Editing / Publishing Platforms & Tools (including collaborative platforms)	0,97
Exhibition/Collection/Edition Platforms & Tools	0,96
Platforms and Communication	0,99
Programming Languages/Packages	0,98
Text Analysis Tools	0,97
Visualization Tools	0,94
No Tool	0,99

Analysen

Die sehr guten Ergebnisse aus der Evaluation sollen im Folgenden im Rahmen einer beispielhaften Analyse weiter diskutiert werden. Dazu wurde der Ursprungsdatensatz mit den 3.737 DH-Papers mit einem „Sliding Window“-Ansatz und einer Fenstergröße von 31 Tokens (das entspricht der Größe unserer Trainingssequenzen, siehe oben) und einer Überlappung von fünf Tokens in insgesamt 753.210 Sequenzen geteilt. Diese wurden anschlie-

ßend klassifiziert und in eine neue Datenbank geschrieben.

Ergebnisse der binären Toolidentifikation

Für den binären Klassifizierungstask der Tool-Detektion werden 14.561 Sequenzen mit potenziellen Toolnennungen vorhergesagt. Obwohl die Trainingssätze auf Basis von nur 246 unterschiedlichen Tools erstellt wurden, ist der Anteil an gefundenen Sequenzen, die genau eines dieser Tools enthalten mit 45,5% überschaubar – mehr als die Hälfte der Sequenzen enthalten andere Tools.

An dieser Stelle soll weiterhin ein Vergleich zu TAPoR, dem mit über 1.600 Einträgen größten Tool-Katalog, angestellt werden. 54,7% der von uns identifizierten Tool-Sequenzen enthalten Tools, die auch in TAPoR gelistet sind und die auch schon von vorhergehenden Lexikonansätzen gefunden wurden (vgl. Barbot et al., 2019; Fischer & Moranville, 2020b, 2020a; Burghardt et al., 2022). In den restlichen 45,3% der Sequenzen finden sich allerdings neue Tools, also solche, die nicht schon in der sehr umfangreichen TAPoR-Liste dokumentiert sind. Mithilfe eines POS-Taggers wurden aus diesen Sequenzen Eigennamen gefiltert, um einen Überblick über die *neuen* Tools zu erhalten.

Es sind dies einerseits Programmiersprachen (*Java*, *SNOBOL4*, *Swift*, *Pascal*, *NetLogo*), Datenbanken (*SQL*) und Markupssprachen (*TEI*, *SGML*), aber auch diverse Belege aus dem Online-Bereich, etwa Social Media (*YouTube*, *Wikipedia*, *Facebook*, *Instagram*, ...) und Web Browser (*Netscape*, *Mozilla*) sowie auch diverse Beispiele aus dem Bereich des Desktop Publishing (*WordPerfect*, *Microsoft Word*, *WordStar*, *PageMaker*). Daneben finden sich zahlreiche weitere, teils antiquierte Tools, die nicht ohne Weiteres zu größeren Gruppen zusammengefasst werden können, etwa: *TreeForm*, *Storyspace*, *MtScript*, *Seshat*, *FarsiTag*, *PlotVis*, *LexStat*, *Galgo*, *Neurolingo*, *AustLit*, *XyWrite*, *StoryTrek*, u.v.m.

Es zeigt sich also, dass unser Ansatz das Tool-Inventar bestehender Kataloge wie TAPoR deutlich erweitern kann, und über die Embeddings auch weitere, neue Tools gefunden werden.

Betrachtet man den jüngsten Aufruf (April 2022)² von TAPoR, bei dem Mitglieder der DH-Community gebeten werden, weitere Tools in TAPoR zu ergänzen, dann könnte unser Ansatz hier im großen Stil automatisiert weitere Vorschläge unterbreiten, indem systematisch verschiedene DH-Publikationen klassifiziert werden.

Ergebnisse der multi-class Toolkategorisierung

In einer zweiten Analyse wurden die Ergebnisse der automatischen Toolkategorisierung diachron ausgewertet (vgl. Abb. 1). Die Charts sind zur Paperanzahl pro Jahr im Datensatz normalisiert und zeigen einen zeitlichen Verlauf des Vorkommens der jeweiligen Kategorie.

Dabei fällt bspw. auf, dass (a) Tools für „Authoring / Annotation / Editing / Publishing“ ebenso wie (b) Tools für „Exhibition / Collection / Edition Platforms“ und (c) Kommunikationsplattformen alle erst Anfang der 2010er an Fahrt aufgenommen haben. Gleichzeitig zeigt sich im

Falle der Programmiersprachen, dass diese in den Anfängen der DH, also den 1960er - 1980er Jahren, besonders populär waren, danach nimmt deren Nennung in Publikationen allerdings deutlich ab.

Tools für die (e) Textanalyse hatten ihren ersten Peak in der Mitte der 1970er, für etwa 10 Jahre, und dann nochmals besonders extrem in den frühen 2000ern und später wieder von 2010-2020. Für eine weitergehende Analyse dieser Konjunkturen ist ein close reading der entsprechenden Publikationen vonnöten, um so zu untersuchen, ob bspw. besonders populäre Einzeltools wie *Voyant* (Release 2003) für diese Spitzen verantwortlich sind. Spannend – und ggf. als Ursache für den Rückgang der Programmiersprachen zu interpretieren – ist weiterhin die steile Karriere von (f) Visualisierungstools, ab Anfang 2010.

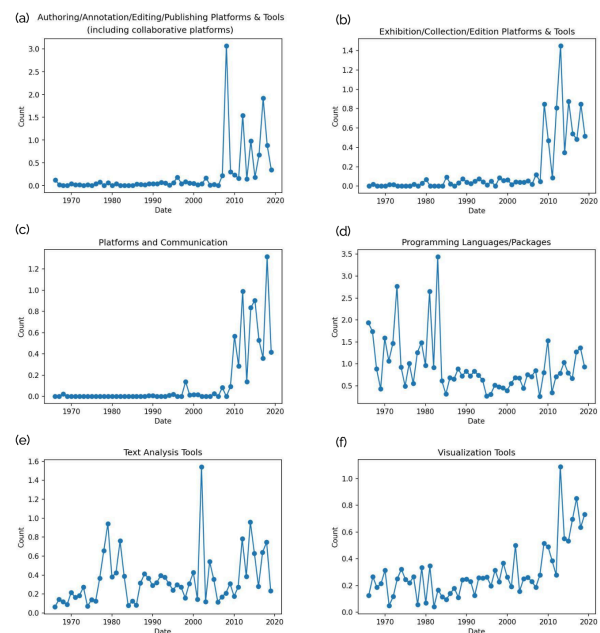


Abbildung 1. Diachrone Ergebnisse der multi-class Toolkategorisierung

Ausblick: Tool Studies 2.0

In diesem Beitrag haben wir dargestellt, dass es vielfältige Diskurse zur Rolle und Funktion von Tools in den DH gibt. Wir sehen großes Potenzial bei systematischen, empirischen Analysen von Tools, um die bestehenden Diskussionen zu ergänzen. Die bisherige, erste Welle der Tool Studies setzt primär auf lexikonbasierte Verfahren (vgl. Barbot et al., 2019; Burghardt et al., 2022; Fischer & Moranville, 2020b, 2020a). Vereinzelt wurden auch einfache Ansätze aus dem Bereich des maschinellen Lernens (vgl. Burghardt et al., 2022; Henny-Krahmer & Jettka, 2022; Zarei, Alireza et al., 2022) erprobt. Mit ersten Experimenten, die das große Potenzial von Transformer-Architekturen in Kombination mit Kontext-Embeddings aufzeigen, hoffen wir bestehende Tool Studies einen Schritt weiter zu bringen. Wir planen weitere Experimente, bei denen wir zum einen das Korpus erweitern wollen, aber

auch weitere Modelloptimierungen vornehmen wollen. U.a. soll künftig ein Klassifikationsschema nach dem Vorbild von TaDiRAH trainiert werden. Neben einer diachronen Vermessung von Tooltrends in den Digital Humanities, soll in letzter Instanz mit einem ausreichend generalisierten Klassifikator ein großes interdisziplinäres Korpus nach dem Vorbild von Luhmann & Burghardt (2021) bezüglich der dort vorkommenden Tools analysiert werden. Ziel wird es sein aufzuzeigen, welche Tools ggf. von anderen Disziplinen in die DH importiert wurden, welche Tools aus den DH erfolgreich exportiert wurden, und welche Tools ggf. DH-spezifisch sind, und in keiner anderen Disziplin vorkommen.

Fußnoten

1. Epochen: 1.0; weitere Konfiguration: „train_batch_size“: 16, „eval_batch_size“: 64, „warmup_steps“: 500, „weight_decay“: 0.01.

2. Tweet: <https://twitter.com/tapordotca/status/1517564033519345664?s=21&t=Mj6Hk76pigAxGt-K8iaUuKA>

Bibliographie

- Barbot, L., F. Fischer, Y. Moranville & I. Pozdniakov.** 2019. "Which DH Tools Are Actually Used in Research?" In *Weltliteratur.Net – A Black Market for the Digital Humanities* (blog). <https://weltliteratur.net/dh-tools-used-in-research/>.
- Borek, Luise, Quinn Dombrowski, Jody Perkins & Christof Schöch.** 2016. "TaDiRAH: A Case Study in Pragmatic Classification." *Digital Humanities Quarterly* 10, No. 1.
- Bradley, John.** 2019. "Digital Tools in the Humanities: Some Fundamental Provocations?" *Digital Scholarship in the Humanities* 34, No. 1: 13–20. https://academic.oup.com/HTTPhandlers/Sigma/LoginHandler.ashx?error=login_required&state=395478f1-3ea2-4078-8eb8-b5c3bf51d898redirecturl%3Dhttpsazjzacademicwoupzwcomzjdshzjarticlezj34zj1zj13zj5063425.
- Burghardt, Manuel, Jan Luhmann & Andreas Niekler.** 2022. "Tools as Epistemologies in DH? A Corpus-Based Exploration." Book of Abstracts of the ADHO Digital Humanities Conference. Tokyo.
- Burghardt, Manuel & Christian Wolff.** 2014. "Humanist-Computer Interaction: Herausforderungen Für Die Digital Humanities Aus Perspektive Der Medieninformatik." Universität Regensburg. <https://doi.org/10.5283/EPUB.35716>.
- Bush, Vannevar.** 1945. "As We May Think." *The Atlantic*.
- Dalbello, Marija.** 2011. "A Genealogy of Digital Humanities." *Journal of Documentation* 67, No. 3: 480–506. <https://doi.org/10.1108/00220411111124550>.
- Devlin, Jacob, Ming-Wei Chang, Kenton Lee & Kristina Toutanova.** 2019. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4171–86. Minneapolis, Minnesota: Association for Computational Linguistics. <https://doi.org/10.18653/v1/N19-1423>.
- Fischer, F. & Y. Moranville.** 2020a. "DH Tools Mentioned in 'The Programming Historian'?" In *Weltliteratur.Net – A Black Market for the Digital Humanities* (blog). <https://weltliteratur.net/dh-tools-programming-historian/>.
- Fischer, F. & Y. Moranville.** 2020b. "Tools Mentioned in DH2020 Abstracts." In *Weltliteratur.Net – A Black Market for the Digital Humanities* (blog). <https://weltliteratur.net/tools-mentioned-in-dh2020-abstracts/>.
- Henny-Krahmer, Ulrike & Daniel Jettka.** 2022. "Softwarezitation Als Technik Wissenskulturs: Vom Umgang Mit in Den Digital." In *DHd2022: Kulturen Des Digitalen Gedächtnisses. Konferenzabstracts*, 203–6. Potsdam.
- Kaden, Ben.** 2016. "Zur Epistemologie Digitaler Methoden in Den Geisteswissenschaften", In *Berliner Beiträge zu Digital Humanities*. <https://doi.org/10.5281/ZENODO.50623>.
- Kittler, Friedrich.** 1993. "Es Gibt Keine Software." In *Draculas Vermächtnis: Technische Schriften*, 1st ed., 225–42. Leipzig.
- Koolen, Marijn, Jasmijn van Gorp & Jacco van Ossenbruggen.** 2019. "Toward a Model for Digital Tool Criticism: Reflection as Integrative Practice." *Digital Scholarship in the Humanities* 34, No. 2: 368–85. <https://doi.org/10.1093/lc/fqy048>.
- Licklider, J. C. R.** 1960. "Man-Computer Symbiosis." *IRE Transactions on Human Factors in Electronics* HFE-1, No. 1: 4–11. <https://doi.org/10.1109/THFE2.1960.4503259>.
- Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer & Veselin Stoyanov.** 2019. "RoBERTa: A Robustly Optimized BERT Pretraining Approach". <https://doi.org/10.48550/ARXIV.1907.11692>.
- Lopez, Patrice & Laurent Romary.** 2015. "GROBID - Information Extraction from Scientific Publications." *ER-CIM News* 100, no. 100.
- Luhmann, Jan & Manuel Burghardt.** 2022. "Digital Humanities—A Discipline in Its Own Right? An Analysis of the Role and Position of Digital Humanities in the Academic Landscape." *Journal of the Association for Information Science and Technology* 73, No. 2: 148–71. <https://doi.org/10.1002/asi.24533>.
- Ramsay, Stephen & Geoffrey Rockwell.** 2012. "Developing Things: Notes toward an Epistemology of Building in the Digital Humanities." In *Debates in the Digital Humanities*. Minneapolis; London: University of Minnesota Press.
- Schindler, David, Felix Bensmann, Stefan Dietze & Frank Krüger.** 2022. "The Role of Software in Science: A Knowledge Graph-Based Analysis of Software Mentions in PubMed Central." *PeerJ Computer Science* 14, No. 8. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8771769/>.
- Traub, Myriam C. & Jacco van Ossenbruggen.** 2015. "Workshop on Tool Criticism in the Digital Humanities." CWI Techreport.
- Unsworth, John.** 2003. "Tool-Time, or 'Haven't We Been Here Already?': Ten Years in Humanities Computing".
- Van Es, Karin, Maranke Wieringa & Mirko Tobias Schäfer.** 2018. "Tool Criticism: From Digital Methods to Digital Methodology." In *Proceedings of the 2nd International Conference on Web Studies*, 24–27. New York,

NY, USA: Association for Computing Machinery, 2018. <https://doi.org/10.1145/3240431.3240436>.

Wolff, Christian. 2015. "The Case for Teaching 'Tool Science': Taking Software Engineering and Software Engineering Education beyond the Confinements of Traditional Software Development Contexts." In *2015 IEEE Global Engineering Education Conference (EDUCON)*, 932–38. Tallinn, Estonia: IEEE. <https://doi.org/10.1109/EDUCON.2015.7096085>.

Zarei, Alireza, Yim Seung-Bin, Frank Fischer, Ma-tej Ďurčo & Philipp Wieder. 2022. "Measuring the Use of Tools and Software in the Digital Humanities: A Machine-Learning Approach for Extracting Software Mentions from Scholarly Articles." In *Book of Abstracts, ADHO DH Conference*. Tokyo.

Zarei, Alireza, Yim Seung-Bin, Matej Ďurčo, Klaus Illmayer, Laure Barbot, Frank Fischer & Edward Gray. 2022. "Der SSH Open Marketplace: Kontextualisiertes Praxiswissen Für Die Digital Humanities." In *DHd2022: Kulturen Des Digitalen Gedächtnisses. Konferenzabstracts*. Potsdam. <https://doi.org/10.5281/ZENODO.6327975>.